RC24323 (W0707-159) July 31, 2007
Computer Science

IBM Research Report

A Framework for QoI-Inspired Analysis for Sensor Network Deployment Planning

Sadaf Zahedi
University of California, Los Angeles
Electrical Engineering Department
Los Angeles, CA 90095-1594

Chatschik Bisdikian
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 704
Yorktown Heights, NY 10598

A framework for QoI-inspired analysis for sensor network deployment planning*

Sadaf Zahedi†
Univ. of California, Los Angeles
Electrical Engineering Department
Los Angeles, CA 90095-1594, USA
szahedi@ee.ucla.edu

Chatschik Bisdikian
IBM Corporation
Thomas J. Watson Research Center
Hawthorne, NY 10532, USA
bisdik@us.ibm.com

ABSTRACT
The quality of information (QoI) that sensor networks provide to the applications they support is an important design objective for their deployment and use. In this paper, we introduce a layered framework for QoI-centered evaluation of sensor network deployment. Considering the detection of transient events class of applications, the layered framework allows separation of the deployment evaluation in three steps: input pre-processing, core analysis, and result post-processing. The layering allows the creation of a rich, modular toolkit system for QoI-centered analysis that can accommodate both existing and new system modeling and analysis techniques. We demonstrate the utility of the framework by comparing the QoI performance of finite-sized sensor networks with general deployment topology, while, at the same time, deriving some new analysis results for the class of applications considered herein.

1. INTRODUCTION
With advances in computing and communication technologies, low(er) cost, intelligent networked sensor systems find their way in a multitude of application environments in areas as diverse as the military intelligence gathering, habitant monitoring, forest monitoring, utility grid monitoring, environmental control, machinery control, and so on.

For the past several years, considerable amount of research work has been contacted regarding the internal operation of a sensor network including ad hoc deployment and operation of sensor networks, energy-aware architectures and protocols, coverage and localization, efficient query dissemination, etc. Given the fact that sensor networks are deployed to serve a purpose (an application), while the value and necessity of the study of the internal operation of these networks is unquestionable, we find it appropriate that additional studies are also needed that bridge the behavior of the sensor networks with the application(s) they support.

We have elected to use the concept of quality of (sensor) information (QoI) as a means to capture an application's information needs from the sensors. Traditional research on QoI, also referred to as IQ (for *information quality*) as well, has its roots in the study of structured information residing in database systems with respect to data consistency, completeness, currency, etc. [5]. Within the context of sensor networks, a comprehensive definition of QoI is still an inconclusive endeavor, but for the purpose of this paper, we have adopted the following definition: [1]

DEFINITION 1. *QoI is the collective effect of the (accessible) knowledge that is available regarding the information derived from the sensors that determine the degree of accuracy and confidence by which the aspects of the real world that are of interest to the user of the information can be represented.*

It follows from the above definition that QoI is a multi-dimensional concept that is applicable (at least some aspects of it) to a user, i.e., an application. The multi-dimensionality of QoI pertains to attributes that "codify" and possibly quantify the knowledge about the information.For example, knowledge about the *timeliness* of the available information, its *reliability*, its (arithmetic) *precision*, and so on, may be used to describe how accurately we can represent the real world using the sensor-derived information [1, 2, 3, 5].

Which information attributes are more important and/or what is the range of acceptable values for them, will depend on application(s) and their needs as specified by an application planner. Given the broad application space for sensor networks, we have elected to use for our research work a general class of applications that are part of many deci-

*Research was sponsored in part by the U.S. Army Research Laboratory and the U.K. Ministry of Defence and was accomplished under Agreement Number W911NF-06-3-0001. The views and conclusions contained in this document are those of the author(s) and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

†Ms. Zahedi performed this worked during her summer 2007 internship at IBM Research.

[1]We have actually based our wording of the definition on a paraphrase of ITU's Rec. E.800 definition of quality of service provided by a telecommunications services provider.

sion making operations. This class of applications relates to event detection that find applications in surveillance and intelligence gathering operations like detecting presence of enemy weaponry, hostile activities (e.g., gunfire, explosions), monitoring remote territories, and so on.

Upon detection of an event of interest, an operative can take an action to mitigate the effects of the event. The effectiveness and the severity of the action, (or the inaction, if the event is not detected) will depend on the QoI that is provided to the operative. Thus, for the class of event detection applications, we elect as QoI attributes of importance the *detection probability* P_d of correctly detecting the occurrence of the event and the *false alarm rate* P_f, i.e., the probability of declaring the occurrence when it did not occur.

With the above as the motivating background, in this paper, we introduce a QoI analysis framework instantiated through a toolkit system. The toolkit (and the analysis framework it represents) serves as a computational aid for a sensor systems designer to evaluate the performance of his/her design based on deployment and QoI constraints provided by the application planner. Considerable amount of work has been done for modeling and analyzing the detection and false alarm performance of detection systems considering various system parameters as signal-to-noise ratio (SNR), channel fading, spatial correlations, and so on. [13, 4, 11, 6, 7]. Niu and Varshney in [11] consider a homogenous system that results to identical SNR at the sensors of the network, while Jayaweera in [6] has considers non-similar SNRs as a result of the spatial spatial distribution of the nodes and channel fading. Katenka et al. in [7] have also studied the decision fusion algorithm, they have derived an approximation for system's decision threshold that provides performance guarantees for the system. These research considers fixed and even persistent events. Detection systems have also studied with relation to energy requirements as well. For example, the work in [10] and [9] proposes a hybrid (neither centralized, nor distributed) energy-driven detection scheme, based on the binary observations of the sensors, which provides designers the flexibility to balance the detection accuracy and energy consumption. In this paper we have not considered energy efficiency as a metric of design or comparison.

We have recognized that past research in the area comprises of point approaches to the detection evaluation. Thus, in this paper, building upon our previous work [1, 2], we take a system-level approach to detection evaluation considering QoI analysis techniques for a class of centralized, distributed (as well as hybrid) detection architectures for events whose signatures have finite support, i.e., *transient events*. Studying the transient event, brings a new dimension to the work due to the time delays between the samples of the spatially distributed sensors. Furthermore, considering the transient events, different sampling policies will result in different errors in performance of detection architectures.

The organization of the paper is as follows: In section 2, we introduce the reference model for the system under consideration and the general toolkit framework. In section 3, we present the core analysis approach based on hypothesis testing. In section 4, we include example uses of the framework while, at the same time, deriving some new results for the

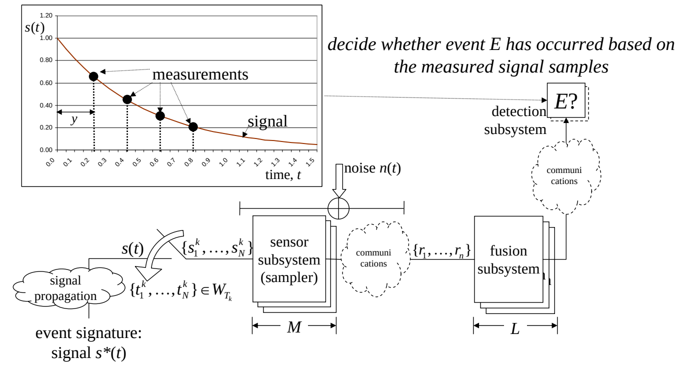

Figure 1: Functional architecture of the reference detection system.

analysis of networks with finite number of sensor and transient events. We conclude in section 5 with some concluding remarks.

2. THE REFERENCE DETECTION MODEL AND QOI ANALYSIS FRAMEWORK

We start this section introducing a reference architecture of our sensor-enabled, detection system and then we introduce the QoI analysis toolkit that is built around the reference model.

2.1 The reference detection system

Figure 1 shows the reference architecture of our sensing system. It comprises three functional subsystems: (a) the *sensor subsystem* or *sampler*; (b) the *fusion subsystem*; and (c) the *detection subsystem*. The sensor subsystem comprises M sensors that sample the physical world; they provide these samples to the fusion subsystem. The fusion subsystem, comprising a collection of L fusion centers, operates on the samples it receives (which could be corrupted by noise) to produce a "summary" description of the samples. The summary, in turn, is used by the detection subsystem to decide whether an event of interest has occurred or not. These three subsystems may be collocated or separated as could be the case of a networked sensing system.

According to the reference model, the sensor subsystem takes samples $\{s_1, s_2, \ldots\}$ (if the event occurred) of the event signature as it is experienced locally by the sensors (the index k in the figure identifies "things" related to the k-th sensor). The sampled signature is distorted by noise and the combination of the two (or just the noise component, when the event did not occurred) produce recordable observations $\{r_1, r_2, \ldots\}$. These observations are then processed by the fusion subsystem to generate the sample summaries.

In case the event occurred, the sequence $\{s_1, s_2, \ldots\}$ can be thought as samples of "projections" of the original event signature $s^*(t)$ at the locations of the sensors. These projections accommodate the impact of the signal propagation, or, in other words, they are reflective of the physical geography of the system and the medium properties. Since, decision making is really made based on the recorded obser-

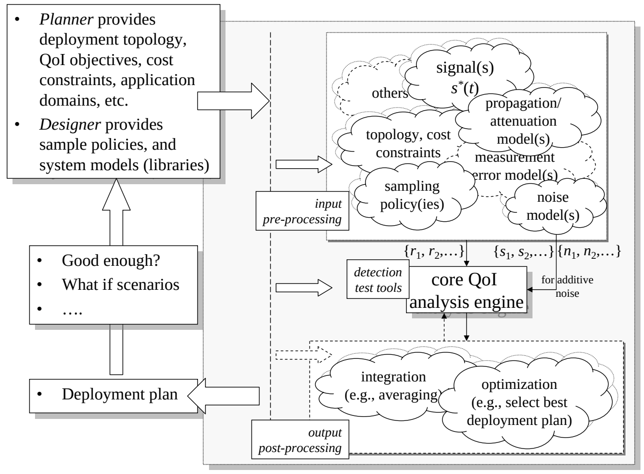

Figure 2: The QoI analysis framework and toolkit usage architecture

vations, if the above projections were known, one could have proceeded with the detection analysis unbeknownst to the physical geometry of the system. In other words, a core fusion and detection analysis engine can be developed that is independent of "external factors" by assuming knowledge of the signal projections at the sensor locations. Anchored on this core analysis engine, a system-level analysis framework can be developed that accommodates the remaining system parameters, like the deployment and observation topology, the application domain (and hence the collection of event signatures $s^*(t)$ to be encountered), the signal propagation models, the noise models, and so on.

Based on the above observation and the reference system model in figure 1, next we propose a toolkit architecture and a framework for a QoI analysis system.

2.2 The QoI analysis toolkit architecture

Figure 2 shows the component and "usage" architecture of the proposed toolkit system. It comprises three major functional blocks for: (a) input pre-processing; (b) detection (QoI) analysis (the core analysis engine); and (c) output post-processing. The input pre-processing deals with all those aspects of the toolkit that generally relate to the constraints imposed on the signal(s) to be detected. These constraints include the deployment and observation topologies (which determine where the sensors are located and where the events occur), the signal propagation and attenuation models (which determine how the original signal projects itself at the sensor locations), the sampling policies (which determine which sensors contribute which samples to the detection process), the noise models (which determine the distortion process of the signal), and so on. All of the above contribute to the creation of the observation sequence $\{r_1, r_2, \ldots\}$ that feeds into the core analysis engine which then calculates the QoI attributes for a very specific (well defined) set of system parameters. The figure also shows the special case of additive noise, where typically, noise samples $\{n_1, n_2, \ldots\}$ are simply added to the signal samples $\{s_1, s_2, \ldots\}$; this case will be studied further in the next section. Given the application requirements, post-processing of the QoI analysis results may be necessary, for example, to calculate averages over an observation region, or calculate optimal position of sensors, and so on. During post-processing, the services of the core analysis engine may be requested again for the QoI analysis of the system for a different set of system parameters.

In addition to the toolkit itself, the figure shows the relation of the application planner and the system designer to the toolkit. The planner provides the problem definition and constraints, e.g., in the form of the observation region, possible cost constraints, the application domain, and so on. The designer provides system level model libraries, e.g., propagation models, noise models, communication and interference models, and so on. When a deployment plan with associated QoI attributes has been produced, the planner may take a decision as to whether the expected performance of the system will be satisfactory enough, or may request additional analysis and trade-off studies for "what if" situations.

The layered analysis framework suggested by the toolkit allows us to place (research) emphasis on different aspects of the system, while still be able to relate these aspects back to the overall system design objectives. For example, in the next section, we focus on the core analysis engine independently of the specifics of the sensor deployment. Special cases of the latter then are considered during the derivation of numerical results in section 4.

We close this section by providing an example of input preprocessing; this prep-processing case will use in the presentation of numerical results later in the paper as well. The example relates to the physical topology (geometry) of the sensor network.

Let the physical topology of the system be represented by the distance vector[2] $\mathbf{d} = [d_1, \ldots, d_M]^T$, where d_k is the distance of the path k that a signal takes from the event location to sensor k. Over path k, let $a_k(t)$ be the *attenuation* the signal experiences, and let v be the propagation speed for the signal. Assuming that an event occurs at time $t = 0$ and possesses the (transient) event signature $s^*(t)$, the signal signature seen by sensor k (the k-th event signature projection), $1 \leq k \leq M$, would be (excluding any noise components):

$$s_k(t) = a_k(t) s^*(t - \tau_k) u(t - \tau_k), \qquad (1)$$

where $u(t)$ is the unit step function. The *time shift* τ_k is due to the propagation delay to sensor k and equals $\tau_k = d_k/v$. As discussed earlier, if $s_k(t)$ were known, a QoI analysis methodology can be employed independently of the original signal $s^*(t)$.

3. THE CORE QOI ANALYSIS ENGINE: HYPOTHESIS TESTING

Hypothesis testing is one of the primary tools used for the performance analysis of detection systems [12, 8]. We will base our core analysis engine on hypothesis testing as well. In this section, we will quickly review the key results from

[2]Bold letters represent the (column) vector version of a corresponding collection of parameters. Unless otherwise stated, T represents the matrix transposition operation.

hypothesis testing and we will (re)interpret them within the context of our toolkit for sensor networks. In the course of doing so, since our focus is on transient (or, in general, non-constant) signals, some new results for detection in sensor networks will also be derived.

In hypothesis testing a number of hypothesis are made, e.g., no event occurred (the *null* event), an event occurred, an event of type E occurred, and so on. Then, based on observations made, sampled data in the context of sensors, one of these hypotheses is declared to hold true. The selection of a hypothesis is done to satisfy certain performance objectives. In Bayesian hypothesis testing, which we will also adopt in this paper, the objective is to minimize the average cost of making a decision.

For our toolkit, we start considering binary hypotheses, i.e., event occurrence, (hypothesis H_1) vs. the null event (hypothesis H_0). The general QoI influenced hypothesis testing formulation for a single sensor where presented in [1, 2], and were based on the following traditional formulation [12, 8]:

$$\begin{array}{llll} \text{hypothesis } H_1: r_i = & s_i + n_i, & i = 1, \ldots, N, \\ \text{hypothesis } H_0: r_i = & n_i, & i = 1, \ldots, N. \end{array} \tag{2}$$

where, under hypothesis H_1, s_i represents the value of the signal at the i-th sampling instance, while, under both hypotheses, n_i represents an additive noise component to the i-th sample, and r_i represents the i-th measurement that the fusion subsystem sees. Hypotheses testing is based using a likelihood ratio test (LRT):[3]

$$\Lambda(\mathbf{r}_N) = \frac{f_{\mathbf{R}_N|H_1}(\mathbf{r}_N)}{f_{\mathbf{R}_N|H_0}(\mathbf{r}_N)} \underset{H_0}{\overset{H_1}{\gtrless}} \eta, \tag{3}$$

where $f_{\mathbf{R}_N|H_i}(\cdot)$ represents the probability density function for the N observations conditioned on hypothesis H_i, $i \in \{0, 1\}$; for notational brevity, in the sequel, we will skip the size index N unless necessary. The threshold η, is calculated from the a priori probabilities for the two hypotheses, P_0 and P_1, and the cost (of one unit) is incurred only when a wrong decision is made, $\eta = P_0/P_1$, and the Bayesian test minimizes the risk of making a wrong decision. When the noise is described by a zero mean additive stationary Gaussian process with covariance matrix $\mathbf{C} = E\{\mathbf{n}^T\mathbf{n}\}$, where $\mathbf{n} = [n_1, \ldots, n_N]$, then the test in (3) reduces to the following:

$$l \triangleq \mathbf{r}^T\mathbf{C}^{-1}\mathbf{s} \underset{H_0}{\overset{H_1}{\gtrless}} \eta^* \triangleq \ln(\eta) + \frac{1}{2}\mathbf{s}^T\mathbf{C}^{-1}\mathbf{s}. \tag{4}$$

The parameter l is referred to as the *sufficient statistic* and represents the summary operation performed by the fusion subsystem on the measurements R_i. Note that the execution of the comparison between l and η^*, for deciding in favor of the one or the other hypothesis, is the responsibility of the detection subsystem in figure 1. Finally, in (4)

$$\psi^2 \triangleq \mathbf{s}^T\mathbf{C}^{-1}\mathbf{s}, \tag{5}$$

is reflective of the *signal-to-noise ratio* (SNR) for this setup.

[3] Reversing the notation in [12], the R_i's in (2) are random variables, while the r_i's in (3) are value instances of the corresponding random variables. For a particular observation instance, the r_i's are what is provided to the fusion subsystem and not the R_i's.

For the special case of a single sensor system and additive, white Gaussian (AWG) noise process with zero mean and variance σ^2 (a $\mathcal{N}(0, \sigma)$ random variable), the QoI performance metrics, i.e., the probability of detection P_d and false alarm rate P_f can be derived from (4), and it is given by [2, 8]:

$$P_d = \Pr(l \geq \eta^\star | H_1) = 1 - \Phi\left(\frac{\ln(\eta)}{\psi} - \frac{\psi}{2}\right), \text{ and} \tag{6a}$$

$$P_f = \Pr(l \geq \eta^\star | H_0) = 1 - \Phi\left(\frac{\ln(\eta)}{\psi} + \frac{\psi}{2}\right), \tag{6b}$$

respectively, where $\Phi(\cdot)$ is the cumulative distribution function for a $\mathcal{N}(0, 1)$ random variable. The square of the parameter ψ is reflective of the signal-to-noise ratio (SNR) and in particular:

$$\psi^2 = \frac{1}{\sigma^2}\sum_{i=1}^{N} s_i^2. \tag{7}$$

Within the context of sensor network, the covariance matrix $\mathbf{C}$ of the noise process in (4) may capture correlations across both the spatial (i.e., across the sensors) and temporal dimensions. We are currently investigating the implications of this fact. Here, we will consider a special case where the covariance matrix is diagonal with variance σ_k^2 for the measurements related to sensor k, $1 \leq k \leq M$.

It should be apparent that, due to the additive nature of terms in (4), the "contributions" from the various sensors to (4) may be (under certain conditions) separable and, hence, possibly easier to describe the system-wide QoI performance using simpler formulations as the ones in in (6) and (7). We will study the implications of this possibility in the next two subsections where we instantiate the above discussion for two different fusion architectures.

In a sensor network, a fusion architecture captures how sensors, and their samples, relate to the fusion and ultimately detection subsystems, see figure 1. In one case, we may have all sensors feeding their samples to a single fusion subsystem; the $M \to 1$ or centralized case. In another, more general case, we may have multiple fusion and detection subsystems in the system, with each sensor associated with only one of them; the $M \to L$ case, where $1 \leq L \leq M$. Just as we consider one or more fusion centers, we can also consider the case of having multiple decision subsystems, tied to a single system-level decision subsystem. Each fusion subsystem will be associated with only one of the decision subsystems. The possibilities for these architectures are numerous and choosing the right one will depend on some form of cost vs. performance trade-off analysis as well; the cost in this case could include the cost of deployment, cost of nodes of various types, cost of energy paid, communication cost and so on. Next we consider two extreme cases: (a) $L = 1$ and (b) $L = M$.

Before proceeding, we first introduce some notations. Let $\mathcal{S}_e^M$ represent the set of all sensors. Let also $\mathcal{R}^N$ and $\mathcal{S}_g^N$ represent the set of all the sensor observations and the set of all the corresponding signal projections, respectively, involved in a single instance of the detection process; in a sense, the sets $\mathcal{R}^N$ and $\mathcal{S}_g^N$ collect all the r_i's and s_i's in (2). The cardinality of the sets is $|\mathcal{S}_e^M| = M$ and $|\mathcal{R}^N| = |\mathcal{S}_g^N| = N$. Let also $\mathcal{S}_g^k$ represent the set of signal (projection) samples $s_1^k, \ldots, s_{N_k}^k$ due to sensor k, where $N_k = |\mathcal{S}_g^k| \geq 0$. Hence, $\mathcal{S}_g = \cup_{k=1}^M \mathcal{S}_g^k$ and $N = \sum_{k=1}^M N_k$. Note that the N_k's need not be equal to accommodate, for example, cases where the sampling rates of the various sensors are not equal. Finally, if $\mathbf{I}_N$ represent the $N \times N$ identity matrix, then the block diagonal covariance matrix considered hereafter will satisfy:

$$\mathbf{C} = \mathrm{diag}\big(\sigma_1^2 \mathbf{I}_{N_1}, \sigma_2^2 \mathbf{I}_{N_2}, \ldots, \sigma_M^2 \mathbf{I}_{N_M}\big). \tag{8}$$

3.1 L=1: Centralized detection

When $L = 1$, all N measurements are fed to a single fusion and detection subsystem. Hence, in this case, we can apply LRT to the entire set of observations $\mathcal{R}^N$ and signal projections $\mathcal{S}_g^N$ to derive the "system-wide" versions of the QoI performance metrics. It follows from (5) and (8) that the system-wide SNR is given by:

$$\psi_{sys}^2 = \sum_{k=1}^{M} \sum_{i=1}^{N_k} \frac{(s_i^k)^2}{\sigma_k^2} = \sum_{k=1}^{M} \psi_k^2, \tag{9}$$

where ψ_k^2 is the SNR due to the signal projection at the location of the k-th sensor. It can be easily shown now that test in (4) results in QoI performance metrics as in (6) with the per sensor ψ replaced by the system-wide ψ_{sys} in (9). Let us define the following:

Definition 2. *The equivalent sensor of a multi-sensor system, is a single-sensor sensing system, that achieves the same QoI level as the multi-sensor system using the same observation set.*

It follows from (9) that the system considered here possesses an equivalent detector.

3.2 L=M: Distributed detection

When $L = M$, detection occurs in two levels. Firstly, a detection decision is made by each sensor (or more accurately, by each detection subsystem that is associated with one-and-only-one sensor) based on its own measurment. Secondly, a detection is made at the system level based on the detection decisions made locally.[4]

Clearly, the QoI performance at the sensor level is given by (6) again, where we use ψ_k instead for ψ. We will write P_d^k and P_f^k for the probability of detection and false alarm for sensor k, respectively, in this case.

Next, to obtain the system level detection probabilities, we need to make use of the *detection policy* that the system uses. A detection policy describes how the sensor-level detection decisions combine to generate the system-wide decision. Such policies will typically be based on some form of a counting strategy, e.g., decide that the event has occurred if at least, say, Q out of the M sensors indicated that it did. Note that while commonly a majority rule is used where $Q > M/2$, we do not believe that this is necessarily the best policy for the general case and have studied it in more detail in section 4.1. The use of a majority rule tacitly assumes that each sensor detects the event with equal probability, which may not always be true for any sensor deployment geography. In accordance to our toolkit's usage architecture, see figure 2, the detection policy is passed to the core analysis engine by the system designer. Any fine tuning of the detection policy to increase performance is made via output post-processing, e.g., compare different decision thresholds.

[4]A properly modified version of these statements hold can be made for the more general case where $L > 1$ as well.

Assuming a "counting" detection policy, let $\mathcal{S}_q^M$ represent the collection of all subsets of sensors that contain q sensors. The the cardinality of $\mathcal{S}_q^M$ is $M!/(q!(M-q)!)$. To declare that the event has occurred, when it has really occurred, i.e., under hypothesis H_1, there should be at least one set of sensors $\mathbf{x}_q \in \mathcal{S}_q^M$ with $q \geq Q$ for which, *all* the sensors in $\mathbf{x}_q$ indicate that the event has occurred.

Each sensor takes a decision locally based on its own measurements. Thus, given these measurements, each sensor makes a detection decision independently of the other sensors. This conditionally independent decision making process, should not be confused with the fact that measurements made by the sensors may relate to each other. Therefore, the system-wide probability of detection $P_d(Q;M)$ for a given threshold Q is given by:

$$\begin{aligned} P_d(Q;M) &= Pr(q \geq Q | H_1) \\ &= \sum_{q=Q}^{M} \Big\{ \sum_{\mathbf{x}_q \in \mathcal{S}_q^M} \Big[\big(\prod_{\substack{\text{sensor} \\ k \in \mathbf{x}_q}} P_d^k \big) \big(\prod_{\substack{\text{sensor} \\ k \notin \mathbf{x}_q}} (1 - P_d^k) \big) \Big] \Big\}. \end{aligned} \tag{10}$$

A similar expression for $P_f(Q,M)$ can be obtain by substituting P_d^k with P_f^k in (10).

While highly unlikely, when the P_z^k's, $z \in \{d, f\}$, are equal to P_z^* independent of k, e.g., when the event -if it occurs- is equidistant from all the sensors, or the differences of the P_z^k's are sufficiently negligible, then the above expressions reduce to the tail distribution of a binomially distributed random variable with parameters M and P_z^* and we have the following:

$$P_z(Q;M) = \sum_{q=Q}^{M} \Big[\binom{M}{q} (P_z^*)^q (1 - P_z^*)^{(M-q)} \Big]. \tag{11}$$

A system with the above binomial case was consider in [11]. Note that for $Q - 1 \leq M$, (11) can be evaluated recursively:

$$P_z(Q;M) = (1 - P_z^*) P_z(Q;M-1) + P_z^* P_z(Q-1;M-1). \tag{12}$$

4. NUMERICAL STUDIES

In this section, we supplement the core analysis development of the previous section with specific deployment models to derive performance results and the corresponding systems. Specifically, we consider two case studies. In the first case, we compare the QoI performance of a totally distributed and the corresponding centralized sensor system that we discussed in section 3. In the second case, we study the behavior of a sensor system by deriving upper and lower bounds for the probability of detection P_d. The use of the framework allows us a systematic step-wise development of the performance results reusing analysis approaches as necessary.

design parameters	
Topology	$M = 4$ sensors
	$\mathbf{d} = [d_1, d_2, d_3, d_4]^T = [1, 2, 4, 8]^T$
Sampling policy	$[N_1, N_2, N_3, N_4] = [20, 20, 20, 20]$
	N_k samples in 1 time unit
Attenuation model	$a_k(t) = a_k = 1/(1 + d_k^2)$
Propagation model	$\tau_k = d_k/\nu$ $(\nu = 3 \times 10^8 m/s)$
Noise variance	$\sigma_1^2 = \sigma_2^2 = \sigma_3^2 = \sigma_4^2 = \sigma^2$

Table 1: design parameters of the scenario

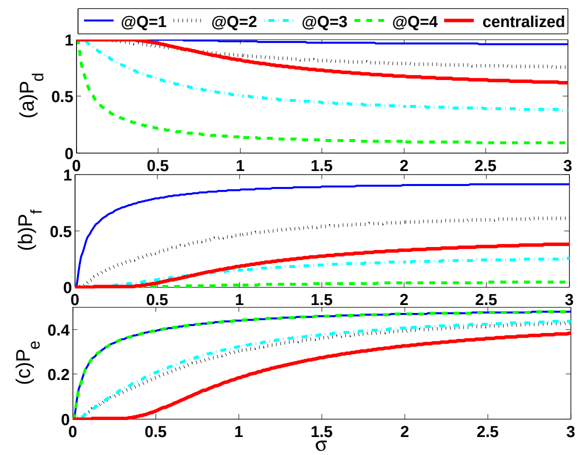

Figure 3: Centralized vs. distributed architectures detection ($M = 4$, $\eta = 1$ and $1 \leq Q \leq M$)

For both cases, we use a 4-sensor system described in table 1; note that we assume an electromagnetic signal, from the family of signals of (relatively) "slow" and "fast" decaying $s^*(t) = 1 - t^n$ and $s^*(t) = (1 - t)^n$, respectively, that last for one time unit. This family of signals, is a convenient one that, for an appropriate selection of the *decay parameter* n ($n \geq 0$), can span from a unit pulse to a linear and to a unit delta functions. The parameters populating table 1 plus the signal description is provided by the application planner and/or the system designer.

4.1 Centralized vs. distributed architectures

In this section, we study the QoI merits of the centralized ($L = 1$) and distributed ($L = M$) architectures –we acknowledge that there are other comparison metrics, e.g., energy consumption [10] and [9], transmission delay, etc., but we focus on the impact of these architectures on QoI.– We will also study their performance as our predisposition of whether the event has occurred (i.e., a priori probability for hypothesis H_1) changes. We consider $\eta = P_0/P_1$ (see discussion about η following (3)). We first consider $\eta = 1$, i.e., $P_0 = P_1$ which will serve as our reference point and then $\eta = 2$ (> 1) and $\eta = 0.5$ (< 1) as well. The signal signature is $s^*(t) = 1 - t^2$, and each sensor contributes 20 samples. We consider a "lucky" system that happens to start sampling just after the signal occurred. We will study the implications of this assumption in our case study analysis in the next subsection.

For the first case of $\eta = 1$, figure 3.a and 3.b show the P_d the

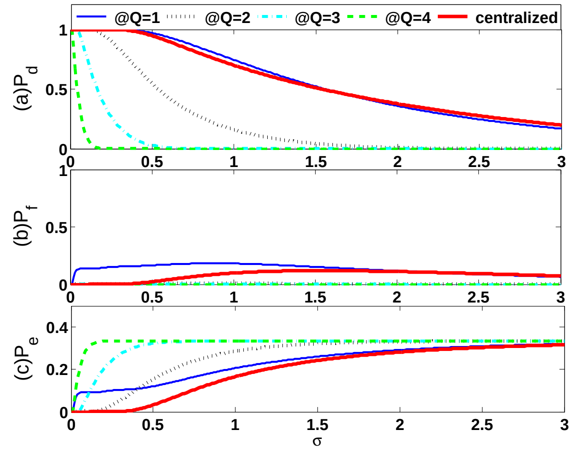

Figure 4: Centralized vs. distributed architectures detection ($M = 4$, $\eta = 2$ and $1 \leq Q \leq M$)

P_f as a function of σ (the standard deviation of the noise) for the centralized architecture and the distributed architecture with different decision thresholds Q. As expected, in the case of the distributed architecture, both probabilities decrease with increasing Q as the detection policy decides in favor of H_1 more liberally; this trend holds true for all cases of η studied. The corresponding QoI performance metrics for the centralized architecture lie above the "middle point," between $Q = 2$ and 3.

Naturally, one will expect that the higher the P_d, and the lower the P_f, the better the QoI will be. To achieve a more direct comparison of the two architectures, we have decided to use an average of the two metrics captured by the *probability of error* $P_e = P_0 P_f + P_1 (1 - P_d) = P_1 (\eta P_f + (1 - P_d))$, which is shown in figure 3.c. The centralized architecture achieves the best (i.e., smallest) P_e. That is not surprising as this architectures makes best use of the information available. However, an interesting byproduct of this analysis is that setting Q at the middle point ($Q = 2$) attains the least P_e for the distributed architecture; due to their poor performance with respect to P_d and P_f for $Q = M$ and $Q = 1$, respectively, these two extreme cases achieve the lowest P_e.

The case where $\eta = 2$, is shown in figure 4. First of all, with $\eta > 1$ (i.e., $P_0 > P_1$), the condition for deciding in favor of H_1 becomes harder to achieve when compared with the $\eta = 1$ case, see (4). Therefore, P_d and P_f decrease in magnitude when compared with these probabilities when $\eta = 1$ (for the same σ level) for both architectures. Comparing the two architectures, the P_d and P_f of the centralized one lie below the "middle point," between $Q = 2$ and 1. Comparing the P_e performance of the two architectures, we again see the centralized one achieving the best performance. With respect to the distributed architecture, the best distributed detection policy is attained when $Q = 2$ for smaller σ and when $Q = 1$ for larger σ. The reduction of Q for the best threshold in the case where $\eta > 1$ when compared with $\eta = 1$ follows from the fact that with decreasing P_1 it becomes harder for any sensor to declare in favor of H_1. Therefore, having even a small(er) number of sensors declaring in favor of H_1 is reason enough to declare in favor of H_1 system-wide.

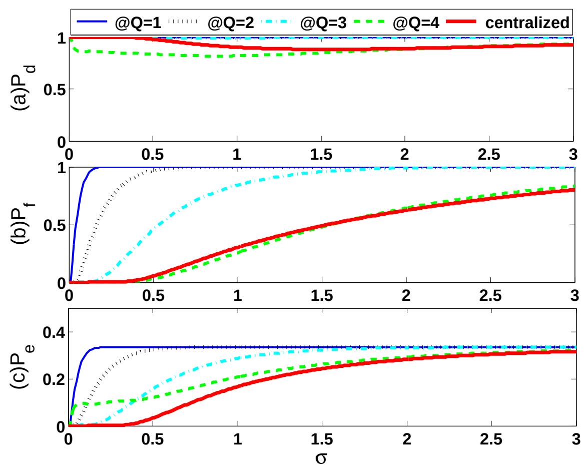

Figure 5: Centralized vs. distributed architectures detection ($M = 4$, $\eta = 0.5$ and $1 \leq Q \leq M$)

Finally, the case where $\eta = 0.5$ is shown in figure 5. Arguing as in the $\eta > 1$ case, the behavior of the QoI metrics in this case are in reverse order relative to the $\eta = 1$ case when compared with $\eta > 1$. With respect to P_e, while the centralized architecture still performs the best, the best distributed detection policy is attained when $Q = 3$ for smaller σ and when $Q = 4$ for larger σ.

Based on the behavior of the two systems when compared on the basis of P_e, we conjecture that in general the best detection policy for the distributed architectures is achieved at a threshold Q that decreases below $M/2$ (and possibly toward M) as η increases above 1. Contrary, the threshold Q increases above $M/2$ (and possibly toward 1) as η decreases below 1.

4.2 The lucky and unlucky sensor

In the previous subsection, we made reference to a lucky sensor k, that sampled the signal just after the signal projection first arrived at the sensor location (and then took additional $N_k - 1$ samples as well). For the decaying signal considered, this lucky sensor picks the strongest possible signal (or better event signature projection) measurements, and hence achieves the highest possibility of detecting the event, hence, $P^k_{d;l} \geq P^k_{d;r}$, where the indexes l and r stand for lucky and real (k-th sensor), respectively. Similarly, we may have an unlucky sensor that samples just before the projection arrives at the sensor and hence, when the signal samples are made, they are the weakest possible, which results in $P^k_{d;u} \leq P^k_{d;r}$.

In general, under hypothesis H_1, a sensor will take its first signal y_k time units after the signal projection arrives at the k-sensor, see figure 1, where $0 < y_k < T_k$; where T_k is the sampling period of the k-th sensor. The i-th sample contributed to the decision process by the k-th sensor is

$$s^k_i = a_k * s^*(t_i - \tau_k + y_k) + n_i, \quad 1 \leq i \leq N_k. \tag{13}$$

Since, y_k is in general unknown (could be modeled as a random variable), the convenience of using the lucky and unlucky sensors instead to bound the performance for the case of decaying signals becomes apparent. We will study these bounds next; this analysis can be generalized for non-decaying signals as well.

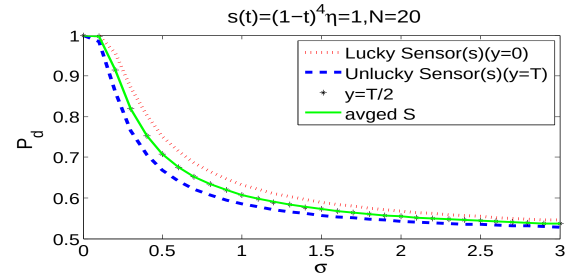

Figure 6: Performance region for $M = 4$ sensors, $s^*(t) = (1-t)^4$ and $\eta = 1$

We will consider the class of slow decaying signals $s^*(t) = (1-t)^n$, applied to a centralized system with parameters as in table 1, we also use $P_0 = P_1$, i.e., $\eta = 1$. Along with the upper and lower bounds derived from the lucky and unlucky sensors, we also consider two additional approximations: (a) one where we assume that $y_k = T_k/2$ (since all the sensor contribute we may drop the index k here); and (b) an average signal approximation given by

$$\hat{s}^k_i = \frac{a_k}{T_k} \int_0^{T_k} s^*(t_i - \tau_k + y)\, dy. \tag{14}$$

Figure 6 shows the P_d for the four cases considered as a function of σ. As expected, the lucky system (where all the sensors are lucky) achieves the highest probability of detection and the unlucky system the lowest for all values of σ. Interestingly, the case where $y = T/2$ and the average signal approximation in (14) result in very close results. In general, it is expected that the higher the sampling rate, the upper and lower bounds become tighter, and so will any approximation that restricts itself between the two bounds. We also notice that as σ increases, P_d for all cases (and hence for the real system too) decreases toward 0.5. That should be expected looking at (6a) that as σ increases (or ψ decreases) and $\eta = 1$, the probability of detection tends toward 0.5. Applying this observation to the equivalent sensor of our centralized system results in the behavior seen in figure 6. Finally, figure 7 shows a comparison of the performance behavior for fast and slow decaying signals as a function of n, for the same system as in figure 6. Specifically, we show the integral between the upper and lower performance bounds (noted as performance area in the figure) and the maximum difference (MAX_{diff}) between the bounds for each class of signals. It can be easily shown that for the fast decaying signals as n increases they tend to the unit delta function and the samples taken differ significantly from each other. Thus, the performance differences between the lucky and unlucky sensors of fast decaying signals increase. The reverse behavior is exhibited for the slow decaying signals that tend to the unit pulse with n and hence the differences between samples diminish. Note that for $n = 0$, we have the case of a persistent signal and there are no upper and lower bound

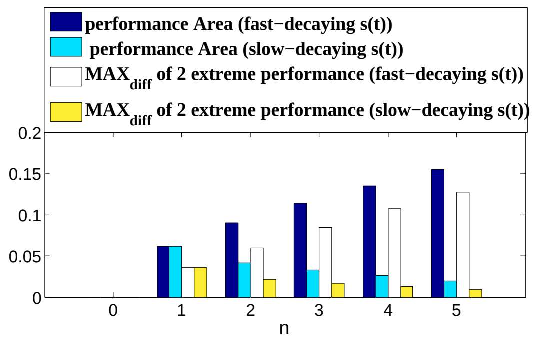

Figure 7: Performance comparison for fast-decaying signals $(1-t)^n$ and slow decaying signals $1-t^n$

differences in this case.

5. CONCLUDING REMARKS

In this paper, we have used QoI as the means to capture an application's information needs from an underlying sensor networks. We then introduce a QoI-based framework for the evaluation of sensor network deployments. The deployment evaluation aids application planners in evaluating the performance capabilities of the network, thus, allowing them to possibly develop contingency plans in case the expected performance is below desired QoI levels.

The framework is reflective of how application planners and system designers will use it, and input parameters to it. It considers a class of event detection applications and comprises of: (a) an input preprocessing step, where the physical constraints of the deployment are considered; (b) a core analysis step, where a common performance analysis approach is used based on hypothesis testing; and (c) a result post-processing step, where "point solutions" obtained through core analysis are combined and processed further to derive the desired end-results. We have applied the framework on analyzing a rather non-homogenous system comprising a finite-size sensor network with transient (non-constant) signals, arbitrary sensor deployment, and different noise levels at each sensor. Note that persistent events, typically considered in most of work on sensor networks, can be studied as limiting cases of transient events, but this is not necessarily true the other way around. Extensions to more general noise models including both spatial and temporal correlations are into considerations. We also, analyzed the performance of "lucky" vs. "unlucky" systems with respect to sampling times thus develop performance bounds of various sampling possibilities.

We have compared centralized vs. distributed detection architectures with respect to QoI performance. For distributed schemes we have also studied the selection of an optimal threshold for counting-based system-wide detection policies. In the latter case, we have demonstrated how a priori knowledge about event occurrence influences the selection of the best detection policy threshold. This analysis is also applicable for case where local fusion is performed but the detection is still centralized, e.g., when the sum on the right side of (4) is calculated in a distributed fashion, but the inequality comparison occurs at a central location.

The framework facilitates the decoupling of the three steps, and the mix-and-match of different analysis, and modeling approaches (both existing and new ones), e.g., use of a Neyman-Pearson instead of Bayesian detection testing for the core analysis engine. The decoupling permits the focus to be placed at separate aspects of the problem at different times, thus simplifying the analysis, and modeling approaches engaged at each step. This various approaches can then become components of a toolkit, that can be used for the analysis of a broad collection of sensing systems.

6. REFERENCES

[1] C. Bisdikian. On sensor sampling and quality of information: A starting point. In *Proc. IEEE PerSeNS Workshop*, White Plains,NY,USA, March 19, 2007.

[2] C. Bisdikian. Quality of information trade-offs in the detection of transient phenomena. In *SPIE Unattended Ground, Sea, and Air Sensor Technologies And Applications IX Conference*, Orlando, FL, USA, April 9-13 2007.

[3] E. Blasch and S. Plano. Dfig level 5 (user refinement) issues supporting situational assessment reasoning. In *8th International Conference on Information Fusion*, 25-28 July 2005.

[4] R. Blum, S. Kassamand, and H. Poor. Distributed detection with multiple sensors:part ii-advanced topics. In *Proceedings of the IEEE*, volume 85, 1997.

[5] S. Ehikioya. A characterization of information quality using fuzzy logic. In *18th Int'l Conf. of the North American Fuzzy Information Processing Society,(NAFIPS 1999)*, June 10-12 1999.

[6] S. K. Jayaweera and K. Al-Tarazi. Large system decision fusion performance in inhomogeneous sensor networks. *Proc. of the IEEE Int'l Conf. on Video and Signal Based Surveillance*, page 72, 2006.

[7] N. Katenka, E. Levina, and G. Michailidis. Local vote decision fusion for target detection in wireless sensor networks. In *Joint Research Conference on Statistics in Quality Industry and Technology*, Knoxville, TN, USA, June 7-9, 2006.

[8] S. M. Kay. *Fundamentals of Statistical Signal Processing: Detection theory.* Prentice Hall, 1998.

[9] A. E. L. Yu. *Detection, Energy, and Robustness in Wireless Sensor Networks.* John Wiley & Sons, 2006.

[10] L.Yu, L.Yuang, G. Qu, and A. Ephremides. Energy-deriven detection scheme with with gauranteed accuracy. In *in Proc. IPSN'06*, Nashville,TN, USA, April 9-21 2006.

[11] R. Niu and P. Varshney. Distributed detection and fusion in a large wireless sensor network of random size. In *EURASIP Journal on Wireless Communications and Networking*, volume 5, 2005.

[12] H. L. van Trees. *Detection, Estimation, and Modulation Theory, part I.* John Wiley & Sons, 1968.

[13] R. Viswanathan and P. Varshney. Distributed detection with multiple sensors:part i-fundamentals. In *Proceedings of the IEEE*, volume 85, 1997.