Computing Quality of Information for Wireless Sensor Networks

Jiajun Jiang[1], Hyeonin Shin[1], Chung-Hao Chen[1,*], Lijia Chen[2] and Wen-Chao Yang[3]

[1]Electrical and Computer Engineering, Old Dominion University, USA

[2]School of Physics and Electronics, Henan University, China

[3]Department of Forensic Science, National Central Police University, Taiwan

Abstract: Wireless sensor networking research is a structural and computer related design that mainly focused on internal wireless sensor network issues such as MAC and routing protocols, energy saving, hard ware design and to some extent on the architecture of gateways that connect a wireless sensor network with the rest of the world. They offer a low-cost solution that provides a high data density. Information obtained from such systems are imprecise in nature but is used for important decision making tasks. This precipitates the need to dynamically compute the quality of information (QoI) based on sensor observations. However, the sensors deployed in an environment do not have the same belief level due to their differences in capabilities and imprecision in sensing and processing. The belief in a sensor represents the level of accuracy in accomplishing a task that can be computed either by comparing the current observation with a reference data set or by performing a physical investigation. It is essential to understand how the sensors are performing with respect to the objective tasks. In this paper we propose a modified Information-driven sensor query (IDSQ) algorithm using reward-and-punishment mechanism to dynamically compute the belief in sensors by leveraging the differences of the individual sensor's opinion. In this structural, results show the suitability of utilizing the dynamically computed belief as an alternative to the accuracy of the sensors. The structural can then be used to distribute the model processing into the wireless sensor network.

Keywords: Quality of information, Multi-sensor systems, QoI structural, Accuracy, Reward-and-punishment mechanism, Structural and computer related design, Sensor belief.

1. INTRODUCTION

In today's environment the evolving of technology leads to the growth of different kind of low cost sense technologies that are utilized in various sectors. The technology that supports is wireless sensing technology it is done with the help of wireless sensor networks (WSN). WSN consists of a huge number of devices, often called motes or sensor nodes. For such a scenario to be feasible, the sensor nodes use cheap, low-quality components and often are battery-powered. A typical WSN node is equipped with various sensing devices, has a small amount of computing power, and is capable of relaying messages from other sensor nodes [1].

Combining WSN with the latest computer technology can be used in very extensive fields, such as industrial control, smart home, environmental monitoring, precision agriculture, military applications, space exploration, intelligent transportation, logistics management, and health monitoring. It has the advantage of reducing equipment complexity and maintenance costs, while also reducing labor costs and returning information in time.

*Address correspondence to this author at the Electrical and Computer Engineering, Old Dominion University, USA; Tel: 757-683-3475;
Email: cxchen@odu.edu

By their nature wireless sensor nodes are prone to errors and failures. Typically, wireless transmissions are unreliable, node hardware can fail, sensors are not very accurate, and nodes can run out of battery power. In such an unstable environment, it is very essential to monitor and assess the quality of information (QoI) of the data provided by the WSN. QoI has been defined as a collection of attributes including timeliness, accuracy, reliability, throughput, and cost. The modeling of sensor network data is an important means to understand and measure QoI requirements in relation to fundamental characteristics of the sensor network, such as for example sampling rate, amount of data to be transmitted, and measurement accuracy. Defining an accurate model of sensor network data is not an easy task as a sensor network monitors the state of a physical system that often is unknown or very difficult to characterize. Due to the imprecision of sensor observation it is therefore very important to measure the QoI based on sensor-driven information, which is a very challenging task. The challenge lies in addressing the following issues [2]:

- Modeling quality of information in terms of relevant parameters and dynamically assessing these parameters in a multi-sensor setting
- Developing quality-aware application and demonstrating the impact of QoI in various application scenarios

E-ISSN: 2409-9821/18

- Addressing the flexibility of addition, removal and replacement of sensors, data processing units and other components in a dynamic sensor environment

The first and second issues have been of interest of many researchers, who studied quality aspects QoI [3] or quality of context (QoC) [4]. On a wider note, both QoI and QoC have the similar objective of measuring the quality value of sensor-driven information, such as the information about what happened in the environment. The listed current works give a better understanding of QoI and its use in several application scenarios. However, they provide little information towards the development of a QoI framework for flexibly handling the addition, removal and replacement of several components, which is related to the third issue above.

The belief in a sensor represents the level of accuracy in accomplishing an observation task [5]. The accuracy of an observation can be calculated by comparing the current observation with the reference data set and/or by performing physical investigation. However, performing a physical investigation or having a reference data set is not practical in an automated monitoring scenario. On the other hand, in a dynamic environment, new sensors may be added with the existing sensors, while nonfunctioning and/or unreliable sensors may be replaced on a demand basis.

The present study in this field either compute the belief of the sensors based on predetermined belief levels [6] or are suitable in a particular application context [5]. In addition, they hardly provide any indication of how the measured belief may resemble the accuracy in a dynamic sensor environment. Furthermore, although very few of the works have used the group observation in determining belief [7], the proposed method fully controls such observations in determining the belief of the participatory sensors.

In this paper, we propose a mechanism to dynamically compute and evolve the belief in sensors over a period of time not only for the WSN but also for the various multi sensors. Our approach determines the difference of opinions obtained by fusing the observations of the sensors and, based on a reward-and-punishment mechanism [8], either increases or decreases the belief of the sensors. In the remainder of this paper, we briefly introduce a related work in Section II, show the general formulation of the problem in Section III, our proposed models are provided in Section IV, provide the implementation results in Section V, followed by the conclusion and future work plan in Section VI.

2. RELATED WORK

Sensor-network data modeling has been addressed in many research projects. Guestrin *et al.* [9] have proposed a model based on linear regression that exploits spatio-temporal data correlation. Their method uses this model in conjunction with Gaussian elimination to reduce the amount of data sent over the sensor network. Deshpande *et al.* [10] present a model based on time varying multivariate Gaussian random variables. Their approach, dubbed BBQ (Barbie-Q: A Tiny-Model Query System), treats sensors as multivariate Gaussian random variables. If the statistics of the Gaussian random variables (mean and covariance matrix) are known, then knowing the outcome for some of the variables in a particular experiment also increases the knowledge about the likely outcome of the unobserved variables.

Researchers in multi-sensors have developed a number of algorithms for quantifying and estimating uncertainties in sensing and estimation applications. Information-driven sensor query (IDSQ) uses a set of information criteria to *select* which sensors to get data from and then incrementally combine the data. The belief update in IDSQ could use either an information filter or a more general non-Gaussian technique such as the sequential Monte Carlo method. IDSQ models the uncertainty in a geometric sense so that active sensor selection can be guided using the spatial configuration of the sensors.

Wirelessly networked sensors often deploy a multi hop RF communication strategy to conserve energy. Pottie's *et al* [11] recent work in data diffusion routing and more generally energy-aware communication attempts to minimize power consumption by using network parameters such as topologies or node power levels.

Measuring the sensor belief has been an interesting research topic due to the uncertainty and imprecision involved in the sensor-based information gathering. To determine the sensor belief, the works in [12] have highlighted on finding out the rate of change in successive measurements from the sensor and argued that the greater the rate of change, the lower the belief. The rate of change is obtained based on the past data, the writers have defined some fuzzy rule sets to determine the self-belief of the sensors. Hughes *et al.* [7] compared the performance of one sensor with another and derived a model for calculating the belief of the sensors. The performance of a sensor is determined based on the current detection outcome that supports an activity. The evidence from multiple sensors that support an activity from an abstract level is used to derive the belief value. Atrey *et al.* [5] propose a dynamic belief calculation approach in the framework of a multimedia surveillance system. Using this mechanism, the belief of a set of non-trusted sensory streams evolves based on their association with other trusted streams. However, it is apparent that the determination of trusted streams would require certain pre-computation, which might cause some overhead in obtaining the overall belief of the sensors and the information of interest.

3. PROBLEM FORMULATION

3.1. Sensing Model and Measure of Uncertainty

Estimation problem is clarified using standard estimation theory. The time-dependent measurement, $\mathbf{z}_i(t)$ of sensor *i* with characteristics, $\lambda_i(t)$ is related to the parameters, $\mathbf{x}(t)$ that we wish to estimate through the following observation model [13]

$$\mathbf{z}_i(t) = \mathbf{h}\big(\mathbf{x}(t), \lambda_i(t)\big), \tag{1}$$

where $\mathbf{h}$ is a (possibly non-linear) function depending on $\mathbf{x}(t)$ and parameterized by $\lambda_i(t)$ which represents the (possibly time dependent) knowledge about sensor *i*. Typical characteristics, $\lambda_i(t)$ about sensor *i* include sensing modality, which refers to what kind of sensor *i* is, sensor position $\mathbf{x}_i$ and other parameters, such as the noise model of sensor *i* and node power reserve.

In (1), we consider a general form of the observation model that accounts for possibly non-linear relations between the sensor type, sensor position, noise model, and the parameters we wish to estimate. A special case of (1) would be

$$\mathbf{h}\big(\mathbf{x}(t), \lambda_i(t)\big) = \mathbf{f}_i(\mathbf{x}(t)) + \mathbf{w}_i(t),$$

Where $\mathbf{f}_i$ is an observation function, and $\mathbf{w}_i$ is additive, zero mean noise with known covariance.

In case $\mathbf{f}_i$ is a linear function on the parameters, (1) reduces to the linear equation

$$\mathbf{h}\big(\mathbf{x}(t), \lambda_i(t)\big) = \mathbf{H}_i(t)\boldsymbol{x}(t) + \mathbf{w}_i(t), \tag{2}$$

In order to illustrate our technique, we will later consider the problem of stationary target localization with stationary sensor characteristics. Here, we assume that all sensors are acoustic sensors measuring only the amplitude of the sound signal so that the parameter vector $\mathbf{x} = [x, y]^T$ is the unknown target position, and

$$\lambda_i = [\mathbf{x}_i, \sigma_i^2]^T, \tag{3}$$

where $\mathbf{x_i}$ is the known sensor position, and σ_i^2 is the known additive noise variance. Note there is no longer a time dependence for $\mathbf{x}$ and λ_i Assuming that acoustic signals propagate isotropically, the parameters are related to the measurements by

$$\mathbf{z}_i = \frac{a}{\|\mathbf{x}_i - \mathbf{x}\|^{\frac{\alpha}{2}}} + \mathbf{w}_i, \tag{4}$$

where a is a given random variable representing the amplitude of the target, α is a known attenuation coefficient, and $\|\mathbf{x}_i - \mathbf{x}\|$ is the Euclidean norm. $\mathbf{w}_i$ is a zero mean Gaussian randomvariable with variance σ_i^2.

In the remainder of this paper, we define the *belief* as a representation of the current *a posteriori* distribution of x given measurements $\mathbf{z_1}, \dots, \mathbf{z}_N$:

$$p(\mathbf{x} | \mathbf{z_1}, \dots, \mathbf{z}_N).$$

Typically, the expectation value of this distribution

$$\bar{\mathbf{x}} = \int \mathbf{x}\, p(\mathbf{x} | \mathbf{z_1}, \dots, \mathbf{z}_N) d\mathbf{x}$$

is considered the estimate (*i.e.*, the minimum mean square estimate), and we approximate the residual uncertainty by the covariance:

$$\Sigma = \int (\mathbf{x} - \bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})^T p(\mathbf{x} | \mathbf{z_1}, \dots, \mathbf{z}_N) d\mathbf{x}.$$

In order to calculate the belief based on measurements from several sensors, we must pay a cost for communicating that information. Thus, maintaining what information each sensor node has about other sensor nodes is an important decision. This is why the sensor characteristics are clearly represented because it is important to know what information is available for various information processing tasks. Since combining measurements into the belief are now assigned costs, the problem is to intelligently choose a subset of sensor measurements which provide “good” information for constructing a belief state as well as minimizing the cost of having to communicate sensor measurements to a single node. In order to choose sensors to provide “good” updates to the belief state, it is essential to understand a measure of the information.

3.2. Sensor Selection

Given the current belief state, we wish to incrementally update the belief by incorporating measurements of other nearby sensors. Among all available sensors in the network, however, not all provide useful information that improves the estimate. Furthermore, some information might be useful, but redundant. The task is to select an optimal subset and to decide on an optimal order of how to incorporate these measurements into our belief update. Due to the distributed nature of the sensor network, this selection has to be done without explicit knowledge of the measurement residing at each individual sensor to avoid communicating less useful information. Hence, the decision has to be made solely based upon the sensor characteristics such as the sensor position or sensing modality and the predicted contribution of these sensors.

Figure **1** shows the basic idea of optimal sensor selection. The image is based upon the assumption that estimation uncertainty can be effectively approximated by a Gaussian distribution, illustrated by uncertainty ellipsoids in the state space. In Figure **1**, the solid ellipsoid indicates the belief state at time *t* and the dashed ellipsoids are the incrementally updated belief after incorporating an additional measurement from a sensor, S1 or S2, at the next time step. Although in both cases,

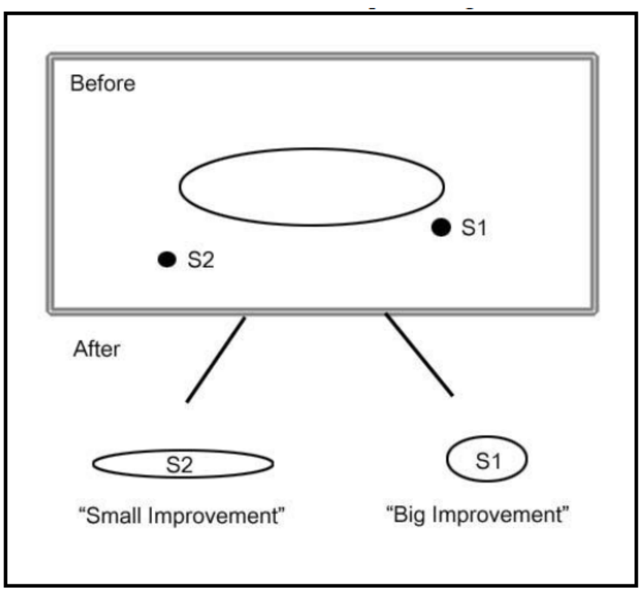

Figure 1: Sensor selection based on information gain of individual sensor contributions.

S1 and S2, the area of high uncertainty is reduced by the same amount, the residual uncertainty in the case of S2 maintains the longest principal axis of the distribution. Based on the underlying measurement task, we will choose case S1 over S2.

3.3. Information Utility Measures

It is essential to define a measure of information utility to quantify the information gain provided by a sensor measurement. We want to show that information content is inversely related to the "size" of the high probability uncertainty region. We first introduce an information-theoretic definition of the utility measure. There are many kinds of measuring methods (*Covariance-Based, Fischer Information Matrix, Entropy of Estimation Uncertainty, Volume of High Probability Region, Sensor Geometry Based Measures*) [13]. In this paper we only describe "*Expected Posterior Distribution measures*" [14] that prove to be practically useful.

Our objective is to predict the information utility of a piece of nonlocal sensor data before obtaining the data. In practice, the prediction must be based on the currently available information: the current belief state and the characteristics of the sensor of interest which includes information such as the sensor position and sensing modality that can be established beforehand.

We assume there are *N* sensors labeled from 1 to *N* and the corresponding measurements of the sensors are. Let $U \subset \{1,\ldots,N\}$ be the set of sensors whose measurements have been incorporated into the belief. That is, the current belief is $p(\mathbf{x}|\ )$. The sensor selection task is to choose a sensor whose data has not been incorporated into the belief yet and which provides the most information. To be specific, let us define an information utility function that assigns a value to each probability distribution. In this case, we ignore the cost term in the objective function. The best sensor, defined by the earlier objective function, is given by

$$\hat{j} = \operatorname{argmax}_{j \in V} \varphi_{\text{Utility}}\ (p(\mathbf{x}|\{\mathbf{z}_i\}_{i \in U} \cup \{\mathbf{z}_i\}))$$

where *V* is the set of sensors whose measurements are potentially useful.

Measures on Expected Posterior Distribution

The idea of using expected posterior distribution is to predict what the new belief state (posterior distribution) would look like if a simulated measurement of a sensor from the current belief state is incorporated. The utility of each sensor can then be quantified by the entropy or other measures on the new distribution from the simulated measurement. We use the tracking problem to derive an algorithm for evaluating the expected utility of a sensor. When a real new measurement is available, the new belief or posterior is evaluated using the familiar sequential Bayesian filtering [14] :

$$p(\mathbf{x}^{(t+1)}|\overline{\mathbf{z}^{(t+1)}}) = C\cdot p(\mathbf{z}_j^{(t+1)}|\mathbf{x}^{(t+1)}) \cdot \int p(\mathbf{x}^{(t+1)}|\mathbf{x}^{(t)}) \cdot p(\mathbf{x}^{(t)}|\overline{\mathbf{z}^{(t)}})\, d\mathbf{x}^{(t)} \quad (5)$$

where $p(\mathbf{x}^{(t)}|\overline{\mathbf{z}^{(t)}})$ is the current belief given a history of the measurement up to time t : $\overline{\mathbf{z}^{(t)}} = \{ \mathbf{z}^{(0)}, \dots, \mathbf{z}^{(t)} \}, p(\mathbf{x}^{(t+1)}|\mathbf{x}^{(t)})$ p(specifies the predefined dynamics model, and $p(\mathbf{z}_j^{(t+1)} \mid \mathbf{x}^{(t+1)})$ is the likelihood function from the measurement of sensor j and C is a normalization constant.

How do we compute the expected value of $p(\mathbf{x}^{(t+1)}|\overline{\mathbf{z}^{(t+1)}})$without having the data $\mathbf{z}_j^{(t+1)}$ in the first place? The idea is to guess the shape of likelihood function from the current belief and the sensor position.

Without loss of generality, the current belief is represented by a discrete set of samples on a grid of the state space. This nonparametric representation of the belief state allows to represent highly non-Gaussian distribution and nonlinear dynamics. Figure **2** shows an example of the grid-based state representation. The gray squares represent the likely position of the target as specified by the current belief. The brighter the square, the more likely the target is there. For a sensor i, given the observation model $\mathbf{z}_j^{(t+1)} = h(\mathbf{x}^{(t+1)}, \mathbf{w}_i^{(t)})$,, where $\mathbf{w}_i^{(t)}$ is the sensor noise, we can estimate the measurement $\mathbf{z}_j^{(t+1)}$from the predicted belief and compute the expected likelihood function:

$$\hat{p}(\mathbf{z}_j^{(t+1)}|\mathbf{x}^{(t+1)}) = \sum_{v_k\in S(\mathbf{x}^{(t+1)})} L_{ki}\left(\mathbf{x}^{(t+1)}, v_k\right) \times \left[p(\mathbf{x}^{(t+1)}|\overline{\mathbf{z}^{(t)}})|_{\mathbf{x}^{(t+1)}=v_k}\right] \quad (6)$$

Where the marginal likelihood is defined as

$$L_{ki}\left(\mathbf{x}^{(t+1)}, v_k\right) \triangleq \hat{p}(\mathbf{z}_j^{(t+1)}(\mathbf{x}^{(t+1)} = v_k)|\mathbf{x}^{(t+1)})$$

And the prediction as

$$p(\mathbf{x}^{(t+1)}|\overline{\mathbf{z}^{(t)}}) \triangleq \sum_{u_k\in S(\mathbf{x}^{(t)})} \left[p(\mathbf{x}^{(t+1)}|\mathbf{x}^{(t)})|_{\mathbf{x}^{(t)}=u_k}\right] \times \left[p(\mathbf{x}^{(t)}|\overline{\mathbf{z}^{(t)}})|_{\mathbf{x}^{(t)}=u_k}\right] \quad (7)$$

Using the estimated likelihood function $\hat{p}(\mathbf{z}_j^{(t+1)}|\mathbf{x}^{(t+1)})$ from sensor i, the expected posterior belief can be obtained as follows:

$$\hat{p}(\mathbf{x}^{(t+1)}|\overline{\mathbf{z}^{(t+1)}}) = C\cdot \hat{p}(\mathbf{z}_j^{(t+1)}|\mathbf{x}^{(t+1)}) \cdot p(\mathbf{x}^{(t+1)}|\overline{\mathbf{z}^{(t)}}). \quad (8)$$

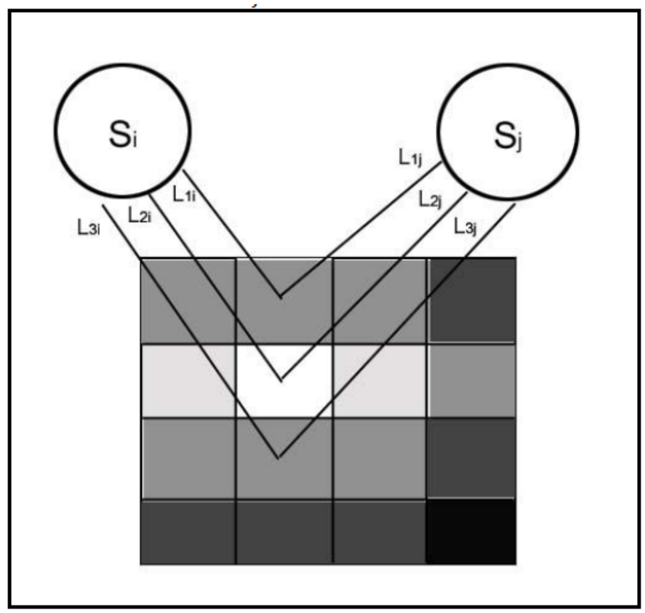

Figure 2: The expected likelihood function for each sensor(i or j) is a weighted sum of the marginal likelihood function conditioned at each grid in the predicted belief distribution.

We can then apply measures such as the entropy to the expected belief $\hat{p}(\mathbf{x}^{(t+1)}|\overline{\mathbf{z}^{(t+1)}})$, as an approximation to the true belief $p(\mathbf{x}^{(t+1)}|\overline{\mathbf{z}^{(t+1)}})$. This approach can apply to non-Gaussian belief since the discrete approximation of the belief state assumes a general form. To compute the expected belief, however, we have conditioned the expected likelihood function on the predicted belief state.

4. PROPOSED MODEL

We describe the information-driven sensor query (IDSQ) algorithms based on the fixed belief carrier protocol in which a designated node such as a cluster leader holds the belief state. The querying node selects optimal sensors to request data from using the information utility measures. To compute the belief state, we will use the reward-and punishment mechanism.

4.1. IDSQ Algorithm

We formulated the problem of distributed tracking as a sequential Bayesian estimation problem. This section outlines a sensor selection algorithm based on the cluster leader type of distributed processing protocol. Although the algorithm is presented using the cluster leader protocol, the ideas of guiding sensor selection using an information-driven criterion can be equally well supported by other methods such as the directed diffusion routing. The description of the algorithm below will clearly discuss what information each sensor node has, even though this part of the algorithm is auxiliary to the sensor selection aspect of the algorithm. Figure **3** shows the flowchart of this algorithm which is identical for every sensor in the cluster.

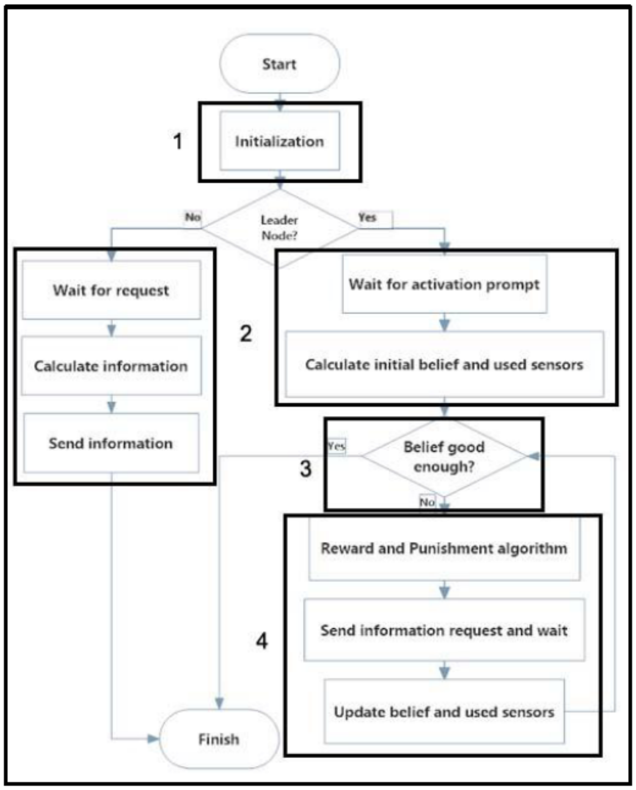

Figure 3: Flowchart of the information-driven sensor querying algorithm.

The measurement model for each sensor *i* is

$$z_i = \frac{a}{\|\mathbf{x}_i - \mathbf{x}\|^{\frac{\alpha}{2}}} + w_i \tag{9}$$

$a \in \mathcal{R}$ is the amplitude of the target uniformly distributed in the interval $[a_{low}, a_{high}]$,

$\mathbf{x} \in \mathcal{R}^2$ is the unknown target position,

$\mathbf{x}_i \in \mathcal{R}^2$ is the known sensor position,

$\alpha \in \mathcal{R}$ is the known attenuation coefficient and

w_i is white, zero-mean Gaussian noise with variance σ_i^2.

The representation θ of the belief will be the history of the collected measurements from the sensors. Thus, the true belief $p(\mathbf{x} \mid \theta)$ and any statistics thereof can be calculated.

4.1.1. Initialization

Assuming all sensors are synchronized so that they are running the initialization routine simultaneously, the first calculation is to pick a leader from the cluster of sensors. Depending on how the leader is determined, the sensors will have to communicate information about their position. Assume also that the leader node has knowledge of certain characteristics $\{\lambda_i\}_{i=1}^N$ of the sensors in the network such as the positions of the sensor nodes.

The relevant characteristics of each sensor *i* are

$$\lambda_i = \begin{bmatrix} \mathbf{x}_i \\ \sigma_i^2 \end{bmatrix} \tag{10}$$

where $\mathbf{x}_i$ is the position and σ_i^2 is the variance of the additive noise term. To be simple, the leader is chosen to be the one whose position $\mathbf{x}_i$ is nearest to the center of the sensors, that is,

$$l = \arg_{j=1,\dots,N}\min \left\| \mathbf{x}_j - \frac{1}{N}\sum_{i=1}^{N} \mathbf{x}_i \right\|. \tag{11}$$

To find the leader, all sensors communicate their characteristics λ_i to each other.

4.1.2. Follower Nodes and Initial Sensor Reading

If the sensor node is not the leader, then the algorithm follows the left branch in Figure **3**. These nodes will wait for the leader node to query them, and if they are queried, they will process their measurements and transmit the queried information back to the leader.

If the sensor node is the leader, then the algorithm follows the right branch in Figure **3**. When a target is present in the range of the sensor cluster, the cluster leader will become activated. The leader node *l* was activated when its amplitude reading satisfied $z_i > \gamma$.

This basically means the leader node becomes activated when the target is less than some given distance away from the leader node, assuming there is no other sound source present. The leader will then store its amplitude value $\theta = z_l$ which is its representation of the belief, and keep track of which sensors' measurements have been incorporated into the belief state *U* = {*l*}.

4.1.3. Belief Quality Test

If the belief is good enough, based on some measure of goodness, the leader node is finished processing. Otherwise, it will continue with sensor selection.

4.1.4. Sensor Selection

According to the belief state, $p(\mathbf{x} \mid \{\mathbf{z}_i\}_{i\in U})$, and sensor characteristics $\{\lambda_i\}_{i=1}^{N}$ pick a sensor node from $\{1,\dots,N\} - U$ which satisfies some information criterion. Say that node is *j*. Then, the leader will send a request for sensor *j*'s measurement, and when the leader receives the requested information, it will update the belief state with $\mathbf{z}_i$ to get a representation of $p(\mathbf{x} \mid \{\mathbf{z}_i\}_{i\in U} \cup \mathbf{z}_i)$, and add *j* to the set of sensors whose measurements have already been incorporated $U := U \cup \{j\}$. We have four different criteria for choosing the next sensor $\hat{\jmath}$:

4.1.4.1. Nearest Neighbor Data Diffusion

$$\hat{\jmath} = \arg_{j\in\{1,\dots,N\}-U} \min \left\| \mathbf{x}_l - \mathbf{x}_j \right\| \tag{12}$$

4.1.4.2. Mahalanobis distance

First, calculate the mean and covariance of the belief state:

$$\mu = \int \mathbf{x}\, p(\mathbf{x} \mid \theta) d\mathbf{x}$$

$$\Sigma = \int (\mathbf{x} - \mu)(\mathbf{x} - \mu)^T p(\mathbf{x} \mid \theta) d\mathbf{x}$$

And choose by

$$\hat{\jmath} = \arg_{j\in\{1,\dots,N\}-U} \min (\mathbf{x}_j - \mu)^T \Sigma^{-1} (\mathbf{x}_j - \mu). \tag{13}$$

4.1.4.3. Maximum Likelihood

This is an ad hoc generalization of the Mahalanobis distance criterion for distributions that are multi-modal. For the special case when the true distribution is Gaussian, this criterion corresponds exactly with the Mahalanobis distance criterion.

$$\hat{\jmath} = \arg_{j\in\{1,\dots,N\}-U} \max p(\mathbf{x}_j \mid \theta). \tag{14}$$

4.1.4.4. Reward-and-Punishment Mechanism

We will show this algorithm in section *B* with details.

Finally, go back to step 3 until the belief state is good enough. At the end of this algorithm, the leader node contains all the information about the belief from the sensor nodes by intelligently querying a subset of the nodes which provide the majority of the information. This reduces unnecessary power consumption by transmitting only the most useful information to the leader node.

4.2. Reward and punishment mechanism

4.2.1. Determining Initial Belief Based on Past Accuracy

The static belief is computed from the past accuracy of each of the sensors. The past accuracy of the sensors is determined by comparing the observations provided by the sensor with the ground truth in the training session. The measure of the past accuracy of the sensors may be represented using four possible parameters including true positive (TP), false positive (FP), false negative (FN) and true negative (TN). For a sensor *i*, these parameters are used to compute the accuracy of the evidences for the occurrence of event *j* as [15],

$$Acc_i^j = \frac{TP_j + TN_j}{TP_j + FP_j + FN_j + TN_j},\ 1 \le i \le n. \tag{15}$$

Acc_i^j is an average value obtained through multiple training repetitions. At the early state of placement, the past accuracy of the sensor observation will be assigned as the value of initial belief, *i.e.* $f_i^j(0) = Acc_i^j,\ i = 1,2,\dots,n.$ To further clarify this, let us consider that a sensor correctly identifies an event eight times (TP = 8) and wrongly misses it two times (FN = 2) out of ten occurrences of that event. Therefore, the Acc_i^j will be equal to 0.80 (TN = 0 and FP = 0).

4.2.2. Analyzing the Current Observation of the Sensors

The current observation is acquired as a score in the scale of 0–1, which indicates the probability of the occurrence of an event. Such a probability score can be acquired by using Bayesian classification strategies to classify a particular observation task or event based on the sensory data.

In a multisensory environment, let there be $n \ge 2$ sensors denoted by a set $S_n = \{S_1, S_2, \dots, S_n\}$, which monitor the event $I_j,\ 1 \le i \le r,$ occurring in the environment. For a particular event I_j let n' number of correlated sensors provide observations about I_j at time instant *t*, which are referred as the "current" observations. Among these n' sensors, not all of them will always agree or disagree on the occurrence of an event. This is again due to the imprecision in the sensing, processing, and changing environment context. Therefore, there would usually be two groups of sensors ϕ_1 and ϕ_2, where ϕ_1 would be in support of the evidence, and ϕ_2 would not support the evidence.

Each time a group of any two sensors S_i and S_k are used together for sensing an event I_j, and their individual observation scores are in favor of the occurrence of the event, the group's overall observation score is computed by using a Bayesian formulation as

$$P_{i,k}^j = \frac{(P_i^j)^{f'_i} \cdot (P_k^j)^{f'_k} \cdot e^{r_{i,k}^j}}{N} \tag{16}$$

where the term N is a normalization factor to limit the probability value within [0, 1], which is expressed as

$$N = (P_i^j)^{f_i'} \cdot (P_k^j)^{f_k'} \cdot e^{r_{i,k}^j} + (1 - P_i^j)^{f_i'} \cdot (1 - P_k^j)^{f_k'} \cdot e^{-r_{i,k}^j} \quad (17)$$

P_i^j and P_k^j are the individual observation scores of the sensors S_i and S_k respectively. The terms $f_i' = f_i/(f_i + f_k)$ and $f_k' = f_k/(f_k + f_i)$ are the two factors computed from the past belief of sensors S_i and S_k at time $t - 1$, where $f_i' + f_k' = 1$. In addition, f_i' and f_k' represent the weights assigned to the sensor observations based on their changing belief.

The term $r_{i,k'}^j \in [-1, 1]$ refers to the agreement coefficient between the sensors S_i and S_k. This is used as a boosting factor when fusing the probability scores. The value −1 indicates a full disagreement, whereas 1 indicates a full agreement among the sensors with respect to observing an event.

$$r_{i,k}^j = \beta\left[1 - 2 \times \left|P_i^j(t) - P_k^j(t)\right|\right] + (1\text{-}\beta)\left[r_{i,k}^j(t-1)\right] \quad (18)$$

$r_{i,k}^j(t-1)$ the term represents the past agreement between the sensors S_i and S_k. The term $1 - 2 \times \left|P_i^j(t) - P_k^j(t)\right|$ represents their current agreement value. The weighting factors β and (1-β) are assigned to the current and past agreement coefficients, respectively.

Finally, the overall observation scores $P_{\phi_1}^j$ and $P_{\phi_2}^j$ of the two sensor groups ϕ_1 and ϕ_2, respectively, are computed by iteratively. Subsequently, if $P_{\phi_1}^j \geq P_{\phi_2}^j$, then the final winning decision from the sensors would be the one provided by the group ϕ_1, which is in favor of the occurrence of the event I_j. Otherwise, the system will assume that the event I_j did not occur, and the probability of nonoccurrence of that event is $P_{\phi_2}^j$. These overall observation scores are further used to determine the dynamic belief of the sensors by adjusting the reward-and-punishment values for the two groups of sensors.

4.2.3. Computing the Run-Time Belief Using the Reward-and –Punishment Mechanism

In this section, we describe how the belief of sensors dynamically grows based on their initial belief and overall observation scores $P_{\phi_1}^j$ and $P_{\phi_2}^j$ for the two groups of sensors. The main idea of our method is to reward and punish the sensors by increasing and decreasing their belief levels, respectively, based on a group decision. The degree of reward and punishment is determined according to the margin by which a sensor group wins over the other. Precisely, we model the dynamic belief in each of the sensors as

If $P_{\phi_1}^j \geq P_{\phi_2}^j$

$$\begin{cases} f_{kk\in\phi_1\neq 0}^j(t) = \dfrac{1}{Z} \cdot f_{kk}^j(t-1) \cdot e^{(\lambda\alpha)(t)} \\ f_{kk'\in\phi_2\neq 0}^j(t) = \dfrac{1}{Z} \cdot f_{kk'}^j(t-1) \cdot e^{-(\lambda\alpha)(t)} \end{cases}$$

else

$$\begin{cases} f_{kk'\in\phi_2\neq 0}^j(t) = \dfrac{1}{Z} \cdot f_{kk'}^j(t-1) \cdot e^{(\lambda\alpha)(t)} \\ f_{kk\in\phi_1\neq 0}^j(t) = \dfrac{1}{Z} \cdot f_{kk}^j(t-1) \cdot e^{-(\lambda\alpha)(t)} \end{cases} \quad (19)$$

where, the terms $f_{kk}^j(t-1)$ and $f_{kk'}^j(t-1)$ are the evolving confidence of each of the sensors in group ϕ_1 and ϕ_2, respectively, with respect to the event I_j based on the individual sensor stream. The exponential terms $e^{(\lambda\alpha)}$ and $e^{-(\lambda\alpha)}$ are the growth and decay factors, respectively, based on current observations. The growth factor is used to positively evolve the belief of the streams that are in support of the overall decision. The value of $\lambda = abs(P_{\phi_1}^j - P_{\phi_2}^j)$. Also, α is the number that is used to control the rate of growth or decay in belief and is experimentally determined. The term Z is the normalization factor to keep the value of confidence in [0, 1].

5. IMPLEMENTATION AND RESULT

In this section we will prove the sensor selection algorithm. In the IDSQ algorithm, we will only implement on the sensor selection model using Reward-and-Punishment mechanism. To show the effectiveness of the suggested mechanism, we conducted an experiment with random data set obtained using Matlab. Our objective was to dynamically determine the belief of the sensors and the high-level information we obtained based on the observation of the sensors. To this end, we provide the details of our experiment in the following sections.

5.1. System Setup and Computing Initial Belief

In our experimental setting, we used random data sets which are training and test set. In order to calculate the initial belief status, training data set was used. We set the four data group in the zone. The zones refer to some logical partition based on the overall position. To evaluate our proposed method, we used 50 data observations from the four sensors. Each observation from a sensor was classified into a supporting group or not supporting group using a Bayesian classifier, which provided a probability score.

As mentioned before, the initial belief of the sensors may be based on their past accuracy, which can be calculated by using (15). However, the calculation of the past accuracy is often not practical due to the overhead it incurs and due to the varying environment context. Therefore, we intentionally ignore the accuracy computation for some sensors and set their initial accuracy as 0.50 on a scale between 0 and 1. However, for two of the sensors (*S*1 and *S*3), we set the initial accuracy as 0.90 and 0.70 each.

5.2. Analyzing the Current Observation

In our experiment, from the 50 sample observations, we apply our method to analyze the current observation of the sensors. We obtain the probability score from the sensors based on the empirical mean and variance value to determine whether the condition is support (C1) or not support (C2). For this, we made a threshold based on the probability of 0.5. Now, we compute the four probability scores as (S1, 0.52, C1), (S2, 0.46, C2), (S3, 0.44, C2), and (S4, 0.54, C1). The scores are grouped into two subsets: one is in support, and the other is not support. That means, given the example observation, the two groups would be ϕ_1 = {0.52, 0.54} and ϕ_2= {0.46, 0.44}, respectively. The fused scores for each of the groups are computed, for which we need to first determine the value of the agreement/disagreement factors among the participating sensors using (18). Given the two sensor observations, the agreement/disagreement among these sensors are computed as $r_{1,4}^{j}$= 0.9650 and $r_{2,3}^{j}$= 0.9737 where the initial agreement/disagreement is assumed to be zero. Now, using (16), we compute the fused scores for each of the groups, which are $P_{\phi_1}^{j}$= 0.8848 and $P_{\phi_2}^{j}$= 0.8507. In obtaining the fused scores, we assume that β is 0.4.

We now compare the fused score of the two groups and observe that the ϕ_1 group has the highest fused scores ($P_{\phi_1}^{j}$= 0.8848); therefore, it is nominated as the winning group, whereas the ϕ_2 group is nominated as the losing group. In the next section, we will show how the belief of the individual sensors is computed based on the observations of the two fused scores.

5.3. Obtaining the Run-Time Belief

The proposed belief calculation method either rewards or punishes a sensor based on whether its observation supports the winning decision. Therefore, for the example case considered in the previous section, the belief of the sensors in group ϕ_1 will increase, whereas the belief of the sensors in group ϕ_2 will decrease. Accordingly, the belief of the sensors *S*1, *S*2, S3 and *S*4 are computed using (19), resulting in their new belief values 0.9396, 0.3663, 0.5743 and 0.6337. In this computation, the value of α is considered as 0.1, which controls the rate of growth or decay factor for belief evolution. Figure **4** shows the dynamically evolved belief over the 50 sample instances. We make the following observations from the results in Figure **4**, which we summarize as follows.

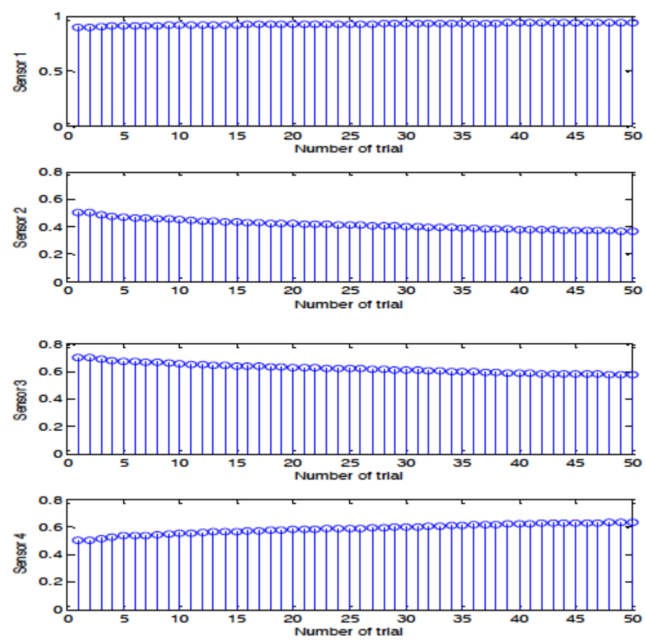

Figure 4: Run-time belief computed for all sensors using the proposed method.

We computed the initial accuracy of sensor *S*1 and S3 based on the past accuracy via training (assumed 0.9 and 0.7). Therefore, this value is likely to be more representative than if it was just set by the designer due to the unavailability of the training data. Consequently, we observe that the evolved belief was not increased much in the case of *S*1 and S3. The initial belief of sensors *S*2 and *S*4 were set as 0.50 without any training. Therefore, we observe that their belief evolves much bigger than S1 and S3. The effect of reward and punishment is also visible from the individual sensor belief. For example, in the case of sensors *S*1 and *S*4 which is winning group, we notice that the slop of the belief increase is higher than S2 and S3 which is losing group. It is also visible from the figure that, in some instances, the belief of the sensors remains the same as the past instance due to the fact that all the sensors either support or oppose the overall observation.

5.4. Computing Required Number of Experiment

Finally, we can calculate the required number of repeat for the sensor selection. Figure **5** shows required experiment number for sensor 4 which is in winning group based on the growth and decay factor and threshold level. For example, if initial belief is 0.5 and threshold level is 0.1, sensor selection belief status should be at least 0.6. In this figure, because sensor 4 is in the wining group, based on the required the number of experiment is increased.

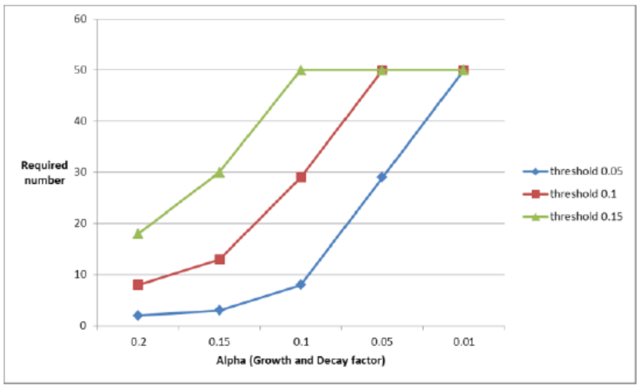

Figure 5: Required number of experiment for sensor 4 based on the threshold level.

Table **1** shows the comparison between our proposed method and Frolik *et al*. [12] in terms of the number of iteration for selecting the best sensor. In Tale 1, the initial belief is set to 0.5 and the growth and decay factor is set to 0.15 for both methods and three threshold levels, 0.05, 0.1, and 0.15 are used for comparisons. We can see that our method uses less number of iterations to selecting the best sensor.

Table 1: The Comparison Between our Method and Frolik *et al*. [12]

	0.05	0.1	0.15
Our Method	5	14	30
Frolik *et al*. [12]	6	20	35

CONCLUSION

In this paper we showed the structure to process sensor data models. Using data models can help to combine readings from different sensors to assess the quality of information (QoI) or to increase energy conservation. We have formulated the problem of distributed tracking using wirelessly connected sensors as an information optimization problem and introduced practically feasible measures of information utility. Introducing an information utility measure allows to dynamically select the best subset of sensors among all possible sensors within the sensor network.

We presented experimental results on sensor selection model of IDSQ using Reward-and Punishment mechanism. The results show the potential of the proposed method and observe that the belief computed in this process may be used as a feasible alternative to the measure of accuracy in a dynamic sensor environment. In our experiment, we only used random data set but algorithm can be used to real time scenarios. We have presented a novel method to dynamically compute the belief of multiple sensors deployed in an observation environment. Our method learns the difference of opinions provided by a group of sensors and utilizes this difference to dynamically evolve their belief levels. In our model, we used an existing Bayesian fusion scheme to fuse the observations.

In the future, we need to prove this algorithm to IDSQ and the other multi sensor networks which are camera and temperature sensors. In this way, the sensor network can seek to make an informed decision about sensing and communication in an energy constrained environment, which is well utilized for monitoring and testing in some harsh environments around the world.

REFERENCES

[1] Hunkeler U, Scotton P, "A quality-of-information-aware framework for data models in wireless sensor networks," Mobile Ad Hoc and Sensor Systems, 2008. MASS 2008. 5th IEEE International Conference on, vol., no., pp.742-747, Sept. 29 2008-Oct. 2 2008
https://doi.org/10.1109/MAHSS.2008.4660118

[2] Hossain MA, Ahmed DT, Parra J, "A framework for computing quality of information in multi-sensor systems," Instrumentation and Measurement Technology Conference (I2MTC), 2012 IEEE International, vol., no., pp.1842-1846, 13-16 May 2012
https://doi.org/10.1109/I2MTC.2012.6229590

[3] Bisdikian C, "On Sensor Sampling and Quality of Information: A Starting Point," Pervasive Computing and Communications Workshops, 2007. PerCom Workshops '07. Fifth Annual IEEE International Conference on, vol., no., pp.279-284, 19-23 March 2007
https://doi.org/10.1109/PERCOMW.2007.88

[4] Abid Z, Chabridon S, "A fine-grain approach for evaluating the quality of context," Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on, 2011; 21(25): pp.444-449, https://doi.org/10.1109/PERCOMW.2011.5766930

[5] Atrey PK, El Saddik A, "Confidence Evolution in Multimedia Systems," Multimedia, IEEE Transactions on, 2008; 10(7): 1288-1298.
https://doi.org/10.1109/TMM.2008.2004907

[6] Rolik J, Abdelrahman M, Kandasamy P, "A confidence-based approach to the self-validation, fusion and reconstruction of quasi-redundant sensor data," Instrumentation and Measurement, IEEE Transactions on, vol.50, no.6, pp.1761-

1769, Dec 2001
https://doi.org/10.1109/19.982977

[7] Hughes K, Ranganathan N, "A model for determining sensor confidence," Robotics and Automation, 1993. Proceedings, 1993 IEEE International Conference on, 1993; 2, 2-6: 136-141.
https://doi.org/10.1109/ROBOT.1993.292137

[8] Hossain MA, Atrey PK, El Saddik A, "Learning Multisensor Confidence Using a Reward-and-Punishment Mechanism," Instrumentation and Measurement, IEEE Transactions on, 2009; 58(5): 1525-1534.
https://doi.org/10.1109/TIM.2009.2014507

[9] Guestrin C, Bodik P, Thibaux R, Paskin M, Madden S, "Distributed regression: an efficient framework for modeling sensor network data," Information Processing in Sensor Networks, 2004. IPSN 2004. Third International Symposium on, 2004; 1- 10: 26-27.
https://doi.org/10.1145/984622.984624

[10] Deshpande A, Guestrin C, Madden S, Hellerstein JM and Hong W. "Model-driven data acquisition in sensor networks," in Proceedings of the Thirtieth International Conference on Very Large Data Base(VLDB' 04), 2004; pp.588-599
https://doi.org/10.1016/B978-012088469-8.50053-X

[11] GJ. Pottie and WJ. Kaiser, "Wireless integrated network sensors." Comm. ACM, 2000; 43(5): 51-58.
https://doi.org/10.1145/332833.332838

[12] Frolik J, Abdelrahman, M, Kandasamy P, "A confidence-based approach to the self-validation, fusion and reconstruction of quasi-edundant sensor data," Instrumentation and Measurement, IEEE Transactions on , 2001; 50(6): 1761-1769.
https://doi.org/10.1109/19.982977

[13] M. Chu, Haussecker, and F. Zhao, "Scalable Information-driven sensor qurying and routing for ad hoc heterogeneous sensor networks," Int. J. H High-Performance Compu. Applicat, 2001

[14] Feng Zhao, Jaewon Shin, Reich J. "Information-driven dynamic sensor collaboration," Signal Processing Magazine, IEEE, 2002; 19(2): pp.61-72.
https://doi.org/10.1109/79.985685

[15] Hossain MA, Atrey PK, El Saddik A. "Learning Multi-Sensor Confidence using Difference of Opinions," Instrumentation and Measurement Technology Conference Proceedings, 2008. IMTC 2008. IEEE, 2008; 12-15: pp.809-813.
https://doi.org/10.1109/IMTC.2008.4547148

Received on 14-12-2018 Accepted on 30-12-2018 Published on 31-12-2018

DOI: http://dx.doi.org/10.15377/2409-9821.2018.05.5