

Robust Target Localization from Binary Decisions in Wireless Sensor Networks

Natallia Katenka, Elizaveta Levina, and George Michailidis

Department of Statistics, The University of Michigan,

1085 S. University, Ann Arbor, MI 48109

Email: {nkatenna, elevina, gmichail}@umich.edu

December 1, 2007

Abstract

Wireless sensor networks (WSN) are becoming an important tool in a variety of tasks, including monitoring and tracking of spatially occurring phenomena. These networks offer the capability of densely covering a large area, but at the same time are constrained by the limiting sensing, processing and power capabilities of their sensors. In order to complete the task at hand, the information collected by the sensor nodes needs to be appropriately fused. In this paper, we study the problems of estimating the location of a target and estimating its signal intensity. The proposed algorithms are based on the local vote decision fusion (LVDF) mechanism, where sensors first correct their original decisions using decisions of neighboring sensors. These corrected decisions are more accurate, robust, and improve detection; however, they are correlated, which renders maximum likelihood estimation intractable. We adopt a pseudo-likelihood formulation and examine several variants of localization and signal estimation algorithms based on original and corrected decisions using direct optimization methods as well as an EM approach. Uncertainty assessments about the parameters of interest are provided using a parametric bootstrap technique. An extensive simulation study of the developed algorithms along with several benchmarks establishes the overall superior performance of the LVDF based algorithms, especially in low signal-to-noise ratio environments. Extensions to tracking moving targets and localizing multiple targets are also considered.

Keywords: wireless sensor network, target localization, decision fusion, target tracking, maximum likelihood

1 Introduction

1.1 Background on Wireless Sensor Networks (WSN)

Detection, identification and tracking of spatial phenomena are important tasks in various environmental and infrastructure applications. Advances in wireless sensor technologies that enable flexible deployments and enhanced sensing and computing capabilities have rendered sensor networks an important tool in this area. Typically, the sensors measure the phenomenon under consideration at discrete points in space and time and the task of the network is to integrate (fuse) the available data in order to estimate and track the parameters of interest; for example, in areas and facilities surveillance monitoring (Estrin, 2006), the task is to detect an intrusion and follow its path; in habitat monitoring (Mainwaring et al., 2002), to identify and track herds; in environmental monitoring (Padhy et al., 2005), to estimate soil moisture levels (Cardell-Oliver et al., 2005), dispersion of pollutants (Kim et al., 2006), etc. Other recent applications of WSNs include identification of chemical, biological, radiological, nuclear and explosive phenomena¹, and infrastructure monitoring (Xu et al., 2004), to name a few.

In order to carry out these tasks, the physical characteristics of the sensors and the constraints imposed by the technology must be taken into consideration. Typically, sensors are autonomously powered devices capable of collecting measurements of one or more types (e.g. acoustic, infrared, etc.) with limited communication, computing and storage capabilities. In a WSN, the nodes are linked by a wireless medium – radio, infrared or optical. The advantage of the latter two media is that they are robust to interferences from electrical devices and hence less prone to false transmissions; however, their main disadvantage is that they require a line of sight for both sender and receiver. On the other hand, radio transmissions can occur at longer distances but their fidelity tends to be lower. The flow of transmissions between sensors is controlled by the medium access control (MAC) protocol. Since the goal is to save power, the favored protocols are of the broadcast type (see SMACS and EAR, Sohrabi et al. (2000)), as opposed to point-to-point protocols used in cellular networks. The advantage of broadcasting is that, to a large extent, it eliminates the necessity of transmission acknowledgments and retransmissions of dropped packets, due to the presence of many neighboring nodes. The transmitted data need to be routed to a 'sink' node, also known as data fusion center. The network layer controls the routing protocol between the sensor and the sink nodes. Again, power efficiency is of paramount importance, and specially designed protocols have been developed for

¹Sensornet: Nationwide detection and assessment of chemical, biological, radiological, nuclear and explosive threats, "http://www.sensornet.gov"

WSNs (more details can be found in Akyildiz et al. (2002)). Finally, their processing and storage capabilities range from devices capable of carrying out computational tasks to simple sensing devices; the former are able to obtain measurements, process them and even store some 'sufficient' statistics for a limited time period, while the latter are constrained to getting data and immediately transmitting them. Finally, in many cases the location of the sensors is unknown and needs to be estimated by the network (Ji and Zha, 2004).

As emphasized above, the overarching limiting factor in WSNs is power, which in most cases comes from batteries and for larger systems from small solar panels. The power constraints can be fairly stringent for many sensor systems, to the point that they spend most of the time in a 'sleep' mode, taking measurements and transmitting them at fixed time intervals on the order of minutes. Nevertheless, advances in micro-machining and micro-electronic technologies are likely to produce significantly more task capable sensors in the not too distant future.

The technological constraints imposed by the nature of WSN, together with their increased importance in a variety of monitoring and environmental settings have created a fertile ground for research. There is active involvement from diverse research communities, including electrical engineering and computer science, materials science and manufacturing, and more recently statistics.

1.2 An Overview of Target Detection and Localization Problems in WSN

The problem we address in this paper is that of estimating a target's location and signal magnitude by a WSN, when only binary transmissions from the sensors are allowed under the power constraints. In general, the problem is formulated as follows: consider N identical sensors deployed at locations s_i , $i = 1, 2, \dots, N$ over a two-dimensional monitoring region R . A target at location $v = (v_x, v_y)$ in the region R emits a signal that is captured by the deployed sensors. Specifically, let $E_i = S_i + \epsilon_i$, $i = 1, 2, \dots, N$ denote the energy measured by the i -th sensor, where $S_i \equiv S_i(v)$ is the signal of the target measured at location i , and E_i is S_i contaminated by i.i.d. random noise ϵ_i . Given the energy measurements, some basic tasks of a WSN are: (i) detect the presence of a target, (ii) identify its location, (iii) with information available over time, track its trajectory through the monitoring region R and (iv) estimate the strength of the signal which may characterize the type of target present.

The WSN accomplishes these objectives by collecting the available measurements from the sensors and processing them appropriately. The *information fusion* occurs at a central location (the fusion center). In the remainder of the paper, it is assumed that important communication and networking issues such as

lossless protocols, connectivity between sensors, synchronization of transmissions among sensors and with the fusion center, etc., have been resolved in advance; here we focus on the collaborative signal processing task at hand. The information transmitted to the fusion center can be either the energy measurements E_i themselves or binary decisions $Y_i = I(E_i \geq \tau_i)$ determined by a pre-specified threshold τ_i related to the individual sensors false alarm probability, with $I(\cdot)$ denoting the indicator function. The former approach is called value fusion, and the latter decision fusion. In general, value fusion is more accurate in terms of detection probability and localization, but decision fusion is more economical due to lower communication cost of one-bit transmissions and proves more robust in noisy environments (Clouqueur et al., 2001). Here, we focus on the situation where only binary transmissions are allowed, although we include value fusion as a benchmark for performance evaluation (Section 3).

1.3 Related Work

The canonical signal processing problems of target detection, localization and tracking over time have received an increasing degree of attention over the last few years. The literature on these problems goes back to radar systems (see Abdel-Samad and Tewfik, 1999), where localization was mostly performed via beam-forming methods. Existing localization algorithms for WSNs can be divided into two general classes: those based on energy readings E_i (Kaplan et al., 2001; Li et al., 2002; Sheng and Hu, 2003; Blatt and Hero, 2006) and those based on binary decisions Y_i (Niu and Varshney, 2004; Noel et al., 2006; Ermis and Saligrama, 2006). Li et al. (2002) used non-linear least squares to localize the target, assuming an isotropic exponentially decaying signal model. For acoustic energy measurements, a maximum likelihood (ML) estimation method based on the Expectation-Maximization (EM) algorithm and a projection solution for the problem of target localization was proposed by Sheng and Hu (2003). The EM algorithm was used to fit the mixture model for energies coming from multiple targets. These methods proved to be more accurate than nonlinear least squares estimates, but computationally more demanding. Compared to techniques that depend on such physical variables as direction of arrival (DOA) and time delay of arrival (TDOA) (Kaplan et al., 2001; Chen et al., 2002; Meesookho and Narayanan, 2005), energy based methods do not require a very accurate synchronization among the sensors and provide accurate estimation of target locations. However, energy-based techniques require transmission of real value data from all the sensors, which may not be feasible under communication constraints. Moreover, the methods of Sheng and Hu (2003) require transmission of the mean and variance of the background noise, which often are unknown and may have to be estimated together with the target location and signal amplitude. Options for reducing communications

cost include implementation of an optimization-based localization algorithm in a distributed manner (Blatt and Hero, 2006), or obtaining energy information only from cluster heads rather than all sensors (Zou and Chakrabarty, 2003). Specifically, Zou and Chakrabarty (2003) proposed a two-step communication protocol between the cluster head and the sensors within the cluster: sensors send binary decisions to the cluster head, which determines candidate target locations and sends requests for stored energy readings to selected sensors.

Binary decision transmission offers significant cost savings, since only positive one-bit detection notifications are sent to the fusion center. Niu and Varshney (2004) developed maximum likelihood target location estimation from binary and multi-bit discrete data, along with the corresponding Cramer-Rao bound. The MLE approach reduces the problem of localization to that of non-linear function optimization, which may suffer from existence of local maxima, slow convergence and high computational complexity. Noel et al. (2006) proposed an improved MLE approach using the same likelihood as Niu and Varshney (2004) maximized by particle swarm optimization techniques, which was shown to outperform deterministic quasi Newton-Raphson schemes. Another recent approach used distributed false discovery rate to select the most informative sensors to communicate with the fusion center, although it relies on multiple within network communications on each step of the localization (Ermis and Saligrama, 2006). Other related papers include target localization in relation to the coverage problem (Wang et al., 2005), and a Bayesian approach to target localization and sensor selection and placement (Wang et al., 2006).

1.4 New Localization Algorithms for Binary Decisions

In this paper, we develop a target localization technique based on binary decisions; however, instead of using the original decisions Y_i , we employ binary decisions that incorporate information from neighboring sensors through the Local Vote Decision Fusion (LVDF) algorithm recently developed by the authors (Katenka et al., 2006). Specifically, individual sensors first adjust their decisions using the majority vote among neighboring sensors and subsequently make a collective decision about a target's presence (more details on LVDF are given in 2.1). The LVDF algorithm is similar to median filtering used in image processing (Gallagher and Wise (1981); Fitch et al. (1984); Zeng (1994); Hwang and Haddad (1995) and others), originally proposed by Tukey (1977). However, the problems of WSNs – target detection with controlled system-wide false alarm rate, target localization and signal estimation – do not arise in image processing and need to be addressed separately. Detection procedures for LVDF were developed in Katenka et al. (2006),

where it was shown that the de-noising effect of LVDF leads to a robust procedure for target detection, particularly in noisy environments with low signal-to-noise ratio. In this paper, we develop a localization and signal estimation procedure for LVDF that enjoys the same robustness properties as the corresponding detection algorithm using a pseudo-likelihood approach (the corrected decisions are correlated, which makes likelihood computations analytically intractable). We also derive an EM algorithm for maximum likelihood estimation from binary decisions, both for the original decisions Y_i and the LVDF-corrected ones. An extensive simulation study establishes that the EM algorithms provide significantly more accurate estimates than direct optimization of the likelihood of binary decisions, but has higher computational complexity. We also provide a bootstrap procedure for uncertainty assessment. Finally, hybrid methods which use partial energy information (obtaining energy readings from sensors that made positive decisions) are shown to perform as well as methods based on the full energy measurements, but at a lower communication cost. We also show how the proposed algorithms can be applied to tracking a moving target over time. It should be noted that at present, appropriate data sets for detection and localization tasks are not publicly available, since target monitoring WSN data tends to be proprietary and/or classified.

The remainder of the paper is organized as follows. In Section 2 the likelihood framework and the EM algorithm for binary decisions are developed, and properties of these estimates and construction of confidence intervals are discussed. In Section 3, numerical results on the performance of the various methods are presented, along with a discussion of implementation issues. Extensions to tracking a target over time are briefly examined in Section 4. Some concluding remarks are given in Section 5.

2 Methods and Algorithms

In this section we present methods for target localization by a WSN deployed over a region R , based on binary measurements – either the original decisions Y_i , $i = 1, 2, \dots, N$ or LVDF-corrected decisions Z_i . The maximum likelihood estimator for location based on Y_i has been derived before (see Niu and Varshney, 2004), but we include a summary for completeness.

It is assumed that the sensor locations s_i are known and that the attenuation of the target's signal is a known function which is monotonically decreasing in the distance from the target $\delta_i(v) = \|s_i - v\|$, and also depends on an attenuation parameter η . That is, the signal at location s_i is given by

$$S_i = S_0 C_\eta(\delta_i(v)), \quad (1)$$

where $C_\eta(0) = 1$, and $S_0 \in [0, \infty)$ denotes the signal strength at the target's location v .

The following two signal models are used in subsequent sections both for illustrative and performance evaluation purposes:

$$\text{M1: } S_i = S_0 \exp(-(\delta_i(v)/\eta)^2), \quad \text{M2: } S_i = \frac{S_0}{1 + (\delta_i(v)/\eta)^3}.$$

Notice that under M1, the signal decays exponentially, while under M2 the signal exhibits polynomial decay. The first model is appropriate for capturing temperature attenuation patterns, whereas the second one or its close variants are widely used for acoustic signals, with the exponent in the denominator ranging between 2 and 5 (Clouqueur et al., 2001; Li et al., 2002).

The noise is assumed to be Gaussian with mean zero and variance σ^2 . The primary parameters of interest are the target's location v and the signal strength S_0 ; obviously, the noise variance σ^2 and the attenuation parameter η affect the estimation problem. Before we proceed to the methods for estimating these parameters, we give more details on the LVDF algorithm that supplies robust binary decisions.

2.1 The Local Vote Decision Fusion Algorithm

The LVDF algorithm is a fusion mechanism that adjusts binary decisions produced by individual sensors to make them robust to noise. Specifically, assuming that all sensors use the same detection threshold, let $Y_i = I(E_i \geq \tau)$ be the original decision made by sensor i . The threshold τ is connected to the sensor's false alarm probability γ by $\gamma = \mathbb{P}(\epsilon_i > \tau)$. Fusing the Y_i decisions, henceforth referred to as ordinary decision fusion (ODF), is achieved according to the following protocol: (i) sensors transmit positive decisions Y_i to the fusion center and (ii) the fusion center makes the final decision $D = I(\sum_i Y_i \geq T)$ using a pre-specified threshold T that controls the network's false alarm probability F .

The LVDF algorithm, on the other hand, first incorporates information from neighboring sensors to obtain robust decisions Z_i . Specifically, let U_i denote the neighborhood of sensor i that includes either all sensors within a certain radius or a fixed number of nearest neighbors; by definition, $i \in U(i)$, so the sensor's own decision is always taken into account. The sensors exchange their initial decisions with neighbors, and each sensor i adjusts its decision according to a majority vote; i.e. $Z_i = I(\sum_{j \in U(i)} Y_j \geq M_i/2)$, where M_i denotes the size of the neighborhood $U(i)$. Positive decisions Z_i are transmitted to the fusion center which makes the final decision $D_\ell = I(\sum_i Z_i \geq T_\ell)$. The advantage of LVDF is illustrated in Figure 1, where

it can be seen that isolated positive decisions away from the target are correctly adjusted. Further, as is typically the case, if communicating with neighbors incurs a lower transmission cost than transmissions to the fusion center, LVDF also reduces the total communications cost by eliminating false positives, since only positive Z_i 's need to be transmitted. If the in-network transmissions are too expensive for LVDF, it can be applied at the fusion center instead, with the same de-noising effect. The main technical challenge

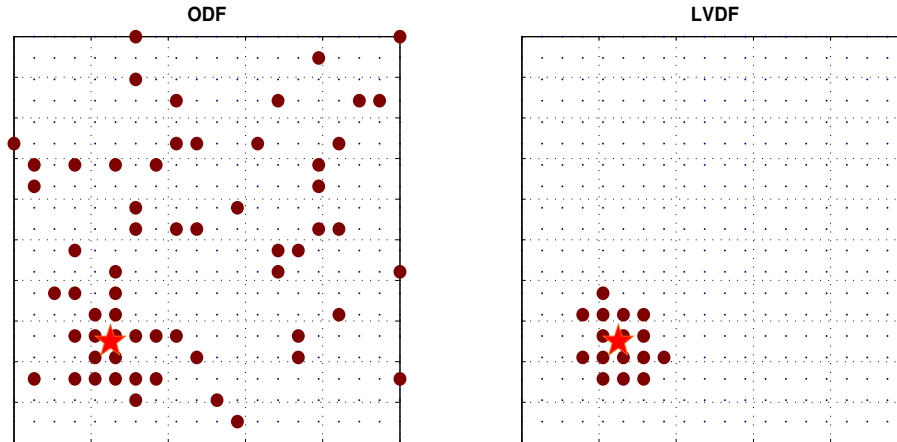


Figure 1: ODF and LVDF decisions on a square grid. The target is marked with a star.

in using LVDF is obtaining the threshold T_ℓ for a given system false alarm rate F due to the dependence of the decisions Z_i , which is accomplished through a normal approximation that uses limit theorems for weakly dependent fields (Katenka et al., 2006). The LVDF algorithm leads to significant gains in target detection probability (see Table 1 in Section 3.2) at low SNRs; it can even outperform detection based on the signals themselves, known as “value fusion” (VF). Value fusion performs detection using the rule $D_E = I(\sum_i E_i \geq T_E)$, where the threshold T_E is straightforward to obtain for a fixed system false alarm probability using the normal assumption. We note that for maximum gain in detection, the neighborhood size should roughly match the scale of the target. In practice, this is often known beforehand if the type of the target is known (vehicle, animal, etc). The issue of neighborhood selection was discussed in detail by Katenka et al. (2006).

2.2 Localization from Original Decisions

As outlined above, each sensor makes an initial decision $Y_i \in \{0, 1\}$ regarding the presence of the target in R . We do not treat the signal itself as random, so the only randomness comes from the i.i.d. noise. Define the

vector of unknown parameters $\theta = (v_x, v_y, S_0, \sigma, \eta)$. Then, the decisions $\{Y_i\}$ are independent Bernoulli random variables with probability of success given by

$$\mathbb{P}(Y_i = 1) \equiv \alpha_i(\theta) = 1 - \Phi(A_i(\theta)),$$

where $\Phi(\cdot)$ denotes the Gaussian cumulative distribution function and $A_i(\theta)$ is the standardized excess energy level given by

$$A_i(\theta) = \frac{\tau - S_0 C_\eta(\delta_i(v))}{\sigma}.$$

The log-likelihood function of the $\{Y_i\}$ is given by:

$$\ell_Y(\theta) = \sum_{i=1}^N [Y_i \log \alpha_i(\theta) + (1 - Y_i) \log(1 - \alpha_i(\theta))]. \quad (2)$$

There are two options for obtaining estimates of the unknown parameters: direct numerical maximization of the log-likelihood function (2) (no closed form solution exists) or the Expectation-Maximization (EM) algorithm (Dempster et al., 1977). The EM can be applied here by viewing $Y_i = I(E_i > \tau)$ as incomplete data on the true energy readings E_i , and consists of an expectation step (E-step), where expected likelihood of the full data conditional on the available data is obtained, and a maximization step (M-step) where the parameters are estimated by maximizing the likelihood from the E-step. The full log-likelihood of the energies, up to an additive constant, is given by

$$\ell_E(\theta) = -\frac{N}{2} \log 2\pi\sigma^2 - \frac{1}{2\sigma^2} \sum_{i=1}^N [E_i - S_0 C_\eta(\delta_i(v))]^2. \quad (3)$$

Maximizing this over S_0 and σ^2 can be done in closed form. This gives the

M-step:

$$\hat{S}_0(v, \eta) = \frac{\sum_{i=1}^N E_i C_\eta(\delta_i(v))}{\sum_{i=1}^N C_\eta^2(\delta_i(v))}, \quad (4)$$

$$\hat{\sigma}^2(v, \eta) = \frac{1}{N} \sum_{i=1}^N (E_i - \hat{S}_0 C_\eta(\delta_i(v)))^2. \quad (5)$$

The other parameters (v and η) are found by numerical optimization of (3) with the expression from (4) and (5) plugged in.

The likelihood (3) comes from a curved exponential family in θ , so the M-step shows that there are just

two quantities that need to be computed in the E-step: $\hat{E}_i = \mathbb{E}[E_i|Y] = \mathbb{E}[E_i|Y_i]$ and $\hat{E}_i^2 = \mathbb{E}[E_i^2|Y] = \mathbb{E}[E_i^2|Y_i]$. Note that each E_i only depends on Y_i rather than all Y because S_i is not random, and ϵ_i 's are independent. These expectations are straightforward to derive: for instance, letting $p_{E_i}(x)$ denote the density of the energies, we obtain

$$\mathbb{E}[E_i|Y_i = 0] = \frac{\int_{-\infty}^{\tau} x p_{E_i}(x) dx}{\int_{-\infty}^{\tau} p_{E_i}(x) dx} = S_0 C_{\eta}(\delta_i(v)) - \frac{\sigma \exp\left(-\frac{A_i(\theta)^2}{2}\right)}{\sqrt{2\pi} \Phi\left(-\frac{A_i(\theta)^2}{2}\right)} \quad (6)$$

Combining analogous computations for $\mathbb{E}[E_i|Y_i = 1]$ and $\mathbb{E}[E_i^2|Y_i]$ gives the

E-step:

$$\hat{E}_i = S_0 C_{\eta}(\delta_i(v)) + \frac{\sigma \exp\left(-\frac{A_i(\theta)^2}{2}\right)}{\sqrt{2\pi}} B_i(\theta, Y) \quad (7)$$

$$\hat{E}_i^2 = S_0 C_{\eta}(\delta_i(v))(\tau - \hat{E}_i) - \hat{E}_i \tau, \quad (8)$$

where

$$B_i(\theta, Y) = \frac{Y_i - 1}{\Phi\left(-\frac{A_i(\theta)^2}{2}\right)} + \frac{Y_i}{1 - \Phi\left(-\frac{A_i(\theta)^2}{2}\right)}.$$

The EM algorithm consists of iterating between the E-step and the M-step until convergence, which tends to be computationally more expensive than direct numerical optimization of the likelihood; however, it tends to produce much more accurate results (see Section 3.2). In both cases, good initial values for the parameters are important; we briefly discuss this issue in Section 3.3.

2.3 Localization from LVDF Decisions

The adjusted decisions Z_i produced by the LVDF algorithm are correlated, which renders the form of the likelihood function in (2) invalid. Computing the full likelihood that would take all dependencies into account requires Gibbs sampling, and is impractical for the application. Thus we adopt a pseudo-likelihood formulation (Besag, 1986; Liang and Yu, 2003), by assuming that all adjusted decisions Z_i are *independent*. Further, we make the following simplifying assumption:

For neighbors $j \in U(i)$, $\mathbb{P}(Y_j = 1) \approx \mathbb{P}(Y_i = 1)$.

Letting $\beta_i(\theta) = \mathbb{P}(Z_i = 1)$, this gives

$$\beta_i(\theta) = \mathbb{P}\left(\sum_{j \in U(i)} Y_j \geq \frac{M}{2}\right) \approx \sum_{k=\lceil M/2 \rceil}^M \binom{M}{k} \alpha_i(\theta)^k (1 - \alpha_i(\theta))^{M-k}.$$

The pseudo-loglikelihood function for the adjusted decisions Z_i is given by:

$$\ell_Z(\theta) = \sum_{i=1}^N [Z_i \log \beta_i(\theta) + (1 - Z_i) \log(1 - \beta_i(\theta))]. \quad (9)$$

Maximum likelihood estimates based on (9) can again be obtained through direct maximization. For the EM algorithm, the M-step is the same as before. The E-step requires calculating the first and second conditional moments $\mathbb{E}[E_i|Z]$ and $\mathbb{E}[E_i^2|Z]$. We first compute the conditional distribution of E_i given all the decisions Z . Write

$$\begin{aligned} \mathbb{P}[E_i|Z] &= \frac{1}{\mathbb{P}(Z)} \sum_{k=0,1} \mathbb{P}(E_i, Z|Y_i = k) \mathbb{P}(Y_i = k) = \\ &= \frac{1}{\mathbb{P}(Z)} \sum_{k=0,1} \mathbb{P}(E_i|Y_i = k) \mathbb{P}(Z|Y_i = k) \mathbb{P}(Y_i = k) \end{aligned} \quad (10)$$

where the last equality follows because conditional on the value of Y_i the energy reading E_i is independent of the vector of corrected decisions Z (recall again that all randomness comes from the noise ϵ_i , not the signal). Integrating (10) gives

$$\mathbb{E}[E_i|Z] = \sum_{k=0,1} \mathbb{E}(E_i|Y_i = k) \mathbb{P}(Y_i = k|Z) \quad (11)$$

$$\mathbb{E}[E_i^2|Z] = \sum_{k=0,1} \mathbb{E}(E_i^2|Y_i = k) \mathbb{P}(Y_i = k|Z) \quad (12)$$

Since we have already obtained $\mathbb{E}[E_i|Y_i]$ and $\mathbb{E}[E_i^2|Y_i]$ in the E-step for ODF, all that remains to be calculated is

$$\mathbb{P}(Y_i = 1|Z) = \frac{\mathbb{P}(Y_i = 1) \mathbb{P}(Z|Y_i = 1)}{\mathbb{P}(Z)} \approx \alpha_i \prod_{j=1}^N \frac{\mathbb{P}(Z_j|Y_i = 1)}{\mathbb{P}(Z_j)} = \alpha_i \prod_{j:i \in U(j)} \frac{\mathbb{P}(Z_j|Y_i = 1)}{\mathbb{P}(Z_j)}, \quad (13)$$

where the first equality is the Bayes' rule, the second comes from the pseudo-likelihood approximation, and the third follows from the fact that only corrected decisions that come from a neighborhood containing sensor i depend on Y_i . For compactness of notation, we suppress the dependence on θ . Once again using

the assumption $\alpha_j \approx \alpha_i$ for $j \in U(i)$, we get

$$\tilde{\beta}_{ji} = \mathbb{P}(Z_j = 1 | Y_i = 1) = \mathbb{P}\left(\sum_{k \in U(j), k \neq i} Y_k \geq \frac{M}{2} - 1\right) \approx \sum_{q=[M/2-1]}^{M-1} \binom{M-1}{q} \alpha_j^q (1 - \alpha_j)^{M-1-q}, \quad (14)$$

and finally

$$\mathbb{P}(Y_i = 1 | Z) = \alpha_i \prod_{j: i \in U(j)} \left(\frac{\tilde{\beta}_{ji}}{\beta_j}\right)^{Z_j} \left(\frac{1 - \tilde{\beta}_{ji}}{1 - \beta_j}\right)^{1-Z_j}. \quad (15)$$

Substituting (15) into (11) and (12) completes the E-step for the LVDF decisions.

2.4 Hybrid Maximum Likelihood Estimates

Hybrid maximum likelihood estimation methods are motivated by the trade-off between energy consumption and accuracy of target localization. As pointed out above, ML estimates based on the full energy measurements are theoretically the most accurate, but also power consuming. The idea of hybrid methods is to use actual energy readings *only* from the sensors with positive decisions, while imputing those for the remaining sensors. This strategy leads to reduced communication costs compared to using all the energies, and improved target localization compared to using binary decisions only.

The proposed hybrid expectation maximization (HEM) algorithm is a straightforward extension of the original EM, where the energies corresponding to $Y_i = 0$ or $Z_i = 0$ are imputed using formulas (7), (8), or formulas (11), (12), (15), respectively, and the energies corresponding to positive decisions are plugged in directly into the M-step.

Numerical simulations show that the HEM algorithm proves to be computationally less expensive than the EM algorithm based on binary decisions, since, on average, it takes fewer iterations to converge. Finally, if one wants to avoid iterative computations, one can replace the energies corresponding to negative decisions by the threshold τ and then proceed to optimize the likelihood function. This variant of the algorithm is referred to as hybrid maximum likelihood (HML).

2.5 Construction of Confidence Intervals for Target Location

The ML and EM estimates for both the ODF and LVDF mechanisms have the usual properties of consistency and asymptotic normality. The latter property, together with a Cramer-Rao bound, can be used to construct

confidence intervals for the parameters of interest. However, due to the pseudo-likelihood approximation employed and occasionally the small sample sizes involved, such intervals may not give correct coverage. We therefore use a parametric bootstrap procedure for uncertainty assessment, which also relies on asymptotic normality but does not use the information bound. We discuss next how to construct a two-dimensional confidence region for the main parameter of interest, target location v .

Let $\hat{v} = (\hat{v}_x, \hat{v}_y)$ be the coordinates of the estimate of the true target location, with $\hat{v} \sim \mathcal{N}(v, \Sigma_v)$, where $\Sigma_v = \text{Var}(\hat{v})$. A two-dimensional confidence region Q satisfies $\mathbb{P}(v \in Q) = 1 - \zeta$, with $1 - \zeta$ denoting the confidence level. Standardizing the location estimate yields

$$\tilde{v} = \Sigma_v^{-1/2}(\hat{v} - v) \sim \mathcal{N}(0, I_2), \quad (16)$$

which in turn implies that the desired confidence region \tilde{Q} for \tilde{v} is a circle of radius r that satisfies $\mathbb{P}(\|\tilde{v}\|^2 \leq r^2) = 1 - \zeta$. The appropriate value of r is given by the $(1 - \zeta)$ -quantile of the χ^2 distribution with two degrees of freedom. The region \tilde{Q} can then be inverted to obtain Q using (16).

This procedure requires an estimate of the covariance matrix $\Sigma = \text{Var}(\hat{\theta})$, with $\hat{\theta} = (\hat{v}, \hat{S}_0)$. Rather than using the asymptotic Cramer-Rao bound which may not be sufficiently accurate for smaller samples, and in particular for the pseudo-likelihood framework employed, we use a parametric bootstrap procedure (Efron, 1994) to obtain a measure of variability of the estimates, as follows.

(i) Energies are simulated from the posited model with parameters set to the maximum likelihood estimates: simulate K samples from the assumed signal attenuation model to obtain

$$E_{i,k}^* = \hat{S}_0 C_{\hat{\eta}}(\delta_i(\hat{v})) + \epsilon_{i,k}^*,$$

where $\epsilon_{i,k}^* \sim \mathcal{N}(0, \hat{\sigma}^2)$, are i.i.d. noise, $i = 1, \dots, N$, $k = 1, \dots, K$.

(ii) The simulated energies are used to obtain bootstrap estimates of the parameters $\hat{\theta}_k^*$, $k = 1, \dots, K$.

(iii) The empirical covariance of the estimates $\hat{\theta}_k^*$ across the K samples gives an estimate for Σ .

This is the procedure followed for obtaining uncertainty estimates presented in the next section.

3 Performance Evaluation

We examine next the performance of the proposed algorithms, focusing on the target location v and the signal amplitude S_0 . In order to limit the number of comparisons, it is assumed that the parameters η and σ^2 are known, although the proposed likelihood framework allows their estimation. Further, note that the parameter η scales the attenuation of the signal within the monitored region R and essentially determines the effective size of the target.

In the simulation study, the signal at each sensor location s_i was contaminated by mean zero, variance σ^2 Gaussian noise and the following procedures were compared: the maximum likelihood estimates based on the original decisions Y_i (ODF) obtained through direct optimization and through the EM algorithm, as well as the hybrid versions, denoted by ML(Y), EM(Y), HML(Y) and HEM(Y), respectively, and the corresponding estimates based on the adjusted decisions (LVDF), denoted by ML(Z), EM(Z), HML(Z) and HEM(Z). In addition, maximum likelihood estimates based on the measured energies (ML(E)) were obtained to serve as a 'gold standard' for comparison purposes. Their computation is equivalent to the M-step in the EM algorithms, except real rather than expected energies are used.

There are two main deployment mechanisms for sensor networks considered in the literature – regular grid or random deployment. Random deployment is often the only feasible option (when, for example, the sensors are “sprinkled” from airplanes), and one expects target localization and tracking to be more challenging due to random gaps in network coverage. Throughout this section, the results will be shown for either a WSN deployed on a 20×20 grid in the unit square, or 400 sensors deployed at random (uniformly distributed in the unit square). We fix the same arbitrarily selected random deployment throughout; additional simulations confirmed that the results are very similar if we average over many random deployments instead. The true target location is set to $v = (1/4, 1/4)$, $S_0 = 2$, the individual sensor's false alarm $\gamma = 0.1$, and the network's false alarm $F = 0.1$. The value of σ is determined from the selected value of the signal-to-noise ratio $\text{SNR} = S_0/\sigma$.

3.1 Detection Performance

Notice that from the detection point of view, there are only three distinct algorithms: value fusion (energies), ODF (original decisions), and LVDF. Their detection performance was compared extensively by Katenka et al. (2006), and thus not included in this study. However, it is important to put all the algorithms on an

equal footing when comparing their localization performance, since they all have different detection rates. To accomplish this, 500 replications of the noise were generated, and only those cases where all three procedures were able to detect the target were kept for further calculations. Table 1 shows how many cases were detected by each method, and how many were detected by all in each case. It can be seen that LVDF exhibits a superior detection performance overall, particularly for low SNRs.

SNR	VF	ODF	LVDF	all	VF	ODF	LVDF	all
	Model M1				Model M2			
	Grid deployment							
2	240	181	378	115	435	340	474	315
5	481	426	500	416	500	499	500	499
10	500	492	500	492	500	500	500	500
	Random deployment							
2	218	163	378	100	418	306	433	259
5	460	369	500	352	499	500	500	499
10	500	491	500	491	500	500	500	500

Table 1: Number of times (out of 500) the target is detected by different methods (VF, ODF, LVDF) and all methods together (all), with $\eta = 0.1$.

3.2 Localization and Signal Estimation Accuracy

The accuracy of the various algorithms for target localization is given in Table 2. The results are obtained for a low (2), medium (5) and high (10) SNR scenarios, with η set to 0.1. It can be seen that the LVDF localization algorithms clearly outperform their ODF counterparts for both models, with the exception of model M2 at SNR=10, where the EM version of ODF performs slightly better than the corresponding LVDF algorithm. Further, in the low SNR regime they clearly outperform the “gold standard” ML(E), while for the medium and high SNR regimes they exhibit a competitive performance. The HEM algorithms tend to be the most accurate, followed by EM, while both ML and HML tend to be less accurate. All algorithms do somewhat better on model M2, since the slower signal decay allows more sensors to pick up the target. It is also worth noting that for the ODF based algorithms, the EM version significantly outperforms the one based on numerical optimization; this is primarily due to the sensitivity of the numerical solver to the selection of initial values, which in the case of the adjusted decisions is not an issue due to the de-noising nature of LVDF (see discussion in 3.3). As expected, for larger values of SNR the accuracy of all the algorithms improves, and for random deployments the pattern remains the same but all methods are somewhat less accurate.

In Table 3, the quality of the estimates of the signal S_0 is given in terms of root mean squared error (RMSE).

SNR	ML(E)	ML(Y)	EM(Y)	HML(Y)	HEM(Y)	ML(Z)	EM(Z)	HML(Z)	HEM(Z)
Model M1, grid deployment									
2	0.208	0.576	0.223	0.329	0.236	0.077	0.056	0.075	0.089
5	0.011	0.502	0.113	0.275	0.066	0.019	0.020	0.012	0.012
10	0.005	0.421	0.028	0.221	0.006	0.019	0.016	0.010	0.006
Model M2, grid deployment									
2	0.164	0.388	0.119	0.226	0.111	0.066	0.050	0.093	0.049
5	0.011	0.101	0.018	0.031	0.012	0.022	0.021	0.012	0.012
10	0.005	0.064	0.017	0.021	0.005	0.020	0.020	0.006	0.005
Model M1, random deployment									
2	0.230	0.744	0.346	0.424	0.326	0.128	0.077	0.221	0.082
5	0.012	0.621	0.233	0.300	0.171	0.052	0.037	0.095	0.020
10	0.006	0.488	0.078	0.283	0.020	0.043	0.024	0.090	0.008
Model M2, random deployment									
2	0.191	0.625	0.197	0.329	0.187	0.092	0.052	0.158	0.052
5	0.012	0.170	0.040	0.216	0.023	0.049	0.031	0.106	0.017
10	0.007	0.096	0.016	0.109	0.006	0.025	0.021	0.043	0.006

Table 2: Average distance from the true location v as a function of SNR with $\eta = 0.1$.

Once again, the LVDF based algorithms outperform their ODF counterparts in almost all situations, and for low SNR they exhibit smaller RMSE than the gold standard. The only exception again is model M2 at SNR=10 with the EM algorithm. Finally, for larger values of SNR all the algorithms become more accurate, with the largest improvement for the gold standard. For both ODF and LVDF, the EM algorithm outperforms the corresponding hybrid EM version estimates for low SNR. For medium and high SNR, the HEM(Z) algorithm exhibits the best performance in terms of accuracy of localization and signal estimation (essentially the same as the "gold standard") (Table 2).

We compare next the accuracy of target localization for the various algorithms as a function of effective target size η for a low SNR=2 regime and a moderate one with SNR=5 (see Figures 2). The HML(Z) and HML(Y) algorithms are omitted from the figures to reduce clutter since they are outperformed by HEM(Z) and HEM(Y), respectively, in every case. In general, larger η corresponds to a bigger target which is easier to detect and localize. The patterns for both models are very similar, and the LVDF algorithms outperform the ODF ones; moreover, in a very noisy environment (SNR=2) the LVDF algorithms also outperform the energy based MLE, while for the case of SNR=5 they perform equally well, especially for larger targets.

The plots in Figure 3 assess the quality of the estimates for S_0 as a function of the target size. One can see that for larger targets the LVDF based algorithms perform well for both models, and somewhat outperform the gold standard at SNR = 2. For smaller size targets, the pattern is not that clear, although all the decision

SNR	ML(E)	ML(Y)	EM(Y)	HML(Y)	HEM(Y)	ML(Z)	EM(Z)	HML(Z)	HEM(Z)
Model M1, grid deployment									
2	0.923	1.297	0.933	1.068	0.736	0.634	0.718	1.017	0.537
5	0.163	1.555	0.923	0.722	0.450	0.558	0.521	0.223	0.169
10	0.077	1.702	0.546	0.861	0.083	0.661	0.460	0.355	0.082
Model M2, grid deployment									
2	0.798	1.104	0.717	1.798	0.747	0.475	0.464	1.491	0.487
5	0.120	0.733	0.314	0.446	0.183	0.415	0.345	0.354	0.124
10	0.060	0.623	0.267	0.262	0.082	0.325	0.325	0.119	0.065
Model M1, random deployment									
2	0.924	1.377	1.016	1.317	0.703	0.729	0.608	1.134	0.718
5	0.197	1.599	1.218	0.663	0.782	0.773	0.734	0.454	0.233
10	0.092	1.599	0.825	0.916	0.308	0.702	0.651	0.573	0.115
Model M2, random deployment									
2	0.796	1.328	0.868	2.449	0.877	0.423	0.518	2.003	0.465
5	0.151	0.855	0.406	0.521	0.276	0.544	0.489	0.458	0.160
10	0.076	0.686	0.234	0.531	0.097	0.314	0.264	0.377	0.079

Table 3: Root Mean Squared Error of estimating S_0 as a function of SNR with $\eta = 0.1$.

based algorithms exhibit similar performance. As before, the HEM versions tend to be the most accurate ones for both algorithms.

3.3 Starting Values

All the decision based algorithms are iterative in nature and require starting values for the parameters of interest. Experience shows that an inferior choice of starting values can slow down convergence and/or lead to poor quality estimates; in fact, the poor performance of localization based on Y is primarily due to this issue. Notice that the starting values have to be a function of the information available for the method, and a good initial guess for the target's location is the centroid of the positive decisions, given by

$$v_0(Y) = \frac{\sum_i s_i I(Y_i = 1)}{\sum_i I(Y_i = 1)}$$

for ODF and

$$v_0(Z) = \frac{\sum_i s_i I(Z_i = 1)}{\sum_i I(Z_i = 1)}$$

for LVDF. Because LVDF eliminates many distant false positives, $v_0(Z)$ tends to be significantly more accurate than $v_0(Y)$. For the benchmark ML(E), where all energies are available, a natural choice of starting value is the location $v_0(E)$ of the maximum energy reading $\max_i E_i$. Table 4 gives average distance from

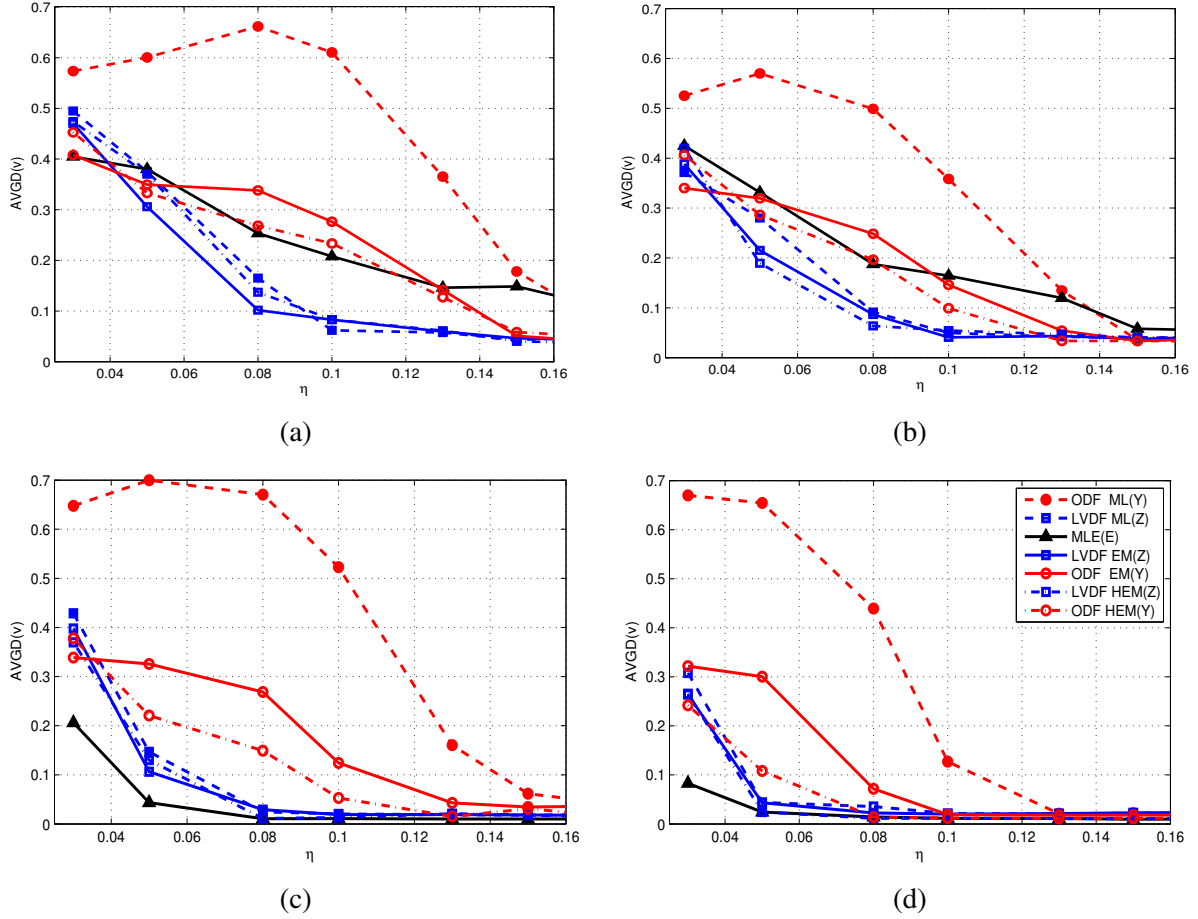


Figure 2: Average distance from true target location v as a function of η for square grid deployment. (a) SNR = 2, model M1; (b) SNR = 2, model M2; (c) SNR = 5, model M1; (d) SNR = 5, model M2.

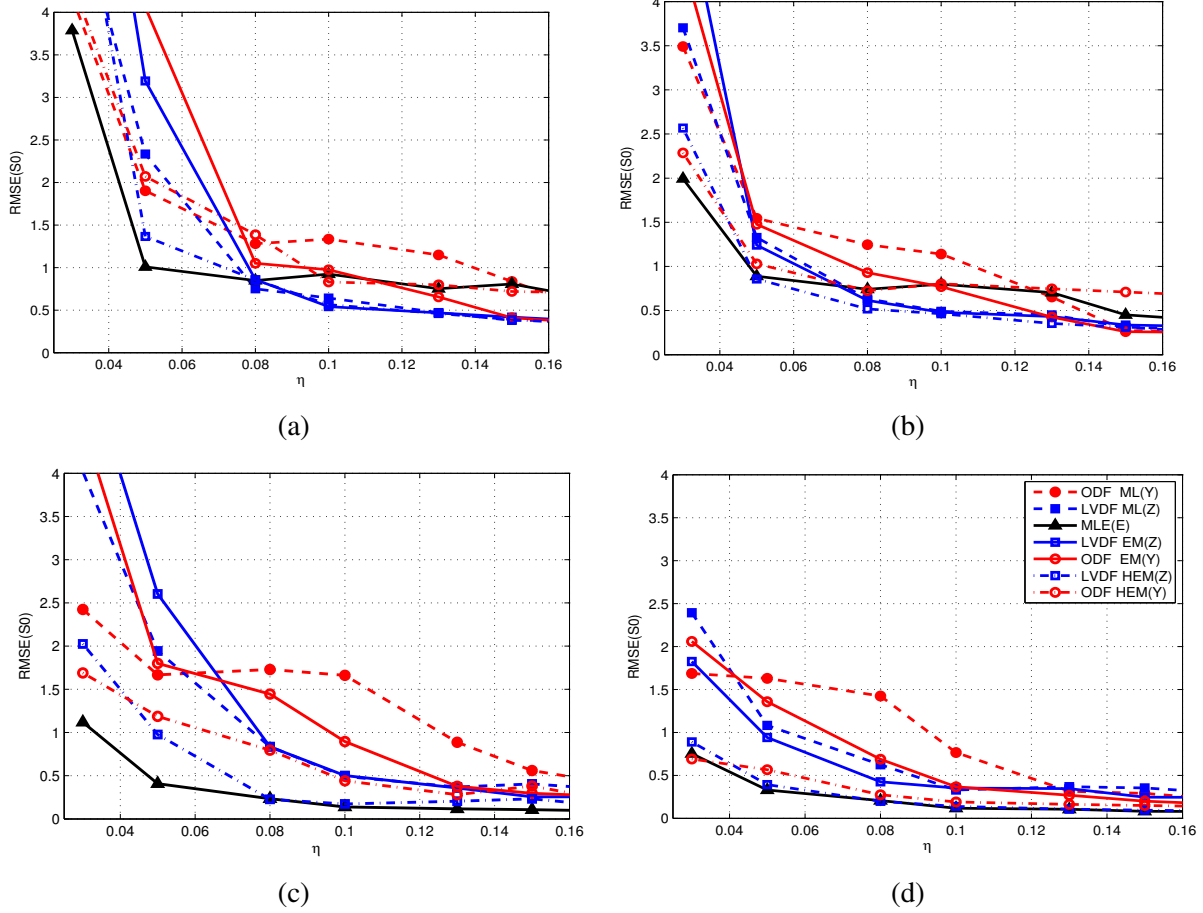


Figure 3: Root Mean Squared Error of estimating S_0 as a function of η for square grid deployment. (a) SNR = 2, model M1; (b) SNR = 2, model M2; (c) SNR = 5, model M1; (d) SNR = 5, model M2.

the starting location for each class of methods (energies, original decisions, and LVDF decisions) to the truth. It can be seen that all methods improve at higher SNR, but the starting value for Y is, on average, much further from the truth than the starting value for Z ; for energies, the starting value based on maximum energy works well at higher SNRs, but not at SNR=2.

SNR	$v_0(E)$	$v_0(Y)$	$v_0(Z)$	$v_0(E)$	$v_0(Y)$	$v_0(Z)$
	Model M1			Model M2		
	Grid deployment					
2	0.244	0.306	0.094	0.214	0.282	0.086
5	0.033	0.266	0.033	0.039	0.220	0.030
10	0.024	0.231	0.022	0.031	0.183	0.026
	Random deployment					
2	0.252	0.321	0.104	0.170	0.305	0.107
5	0.027	0.285	0.070	0.029	0.236	0.057
10	0.022	0.245	0.048	0.024	0.196	0.031

Table 4: Average distance of starting values from the true location as a function of SNR with $\eta = 0.1$.

If better starting values are available from some prior information or external knowledge, performance of all methods will improve, but particularly that of Y . We found that if the search is initialized very close to the truth, the ML(E) generally provides the best localization, as expected, followed closely by Y and Z . However, the starting values we use are integral to the methods and are unlikely to be improved upon without extra information.

3.4 Robustness to Model Misspecification

The performance of all algorithms may change when aspects of the true model are misspecified. Of particular interest is the change in performance of the best algorithms, and changes in relative performance of the different algorithms. First, we test robustness to signal model misspecification by investigating how the quality of the estimates is affected when the signal is generated by model M2, but the assumed signal model is M1. The performance of the algorithms relative to each other remains exactly the same (results not shown, but see the pattern in Figures 2 and 3): LVDF is still the most accurate for low SNRs followed by ML(E) and ODF, and ML(E) is the best for higher SNRs, closely followed by LVDF. To assess changes in absolute accuracy of each algorithm, the differences between the error under misspecified model and the error under the true model are shown in Figure 4, for SNR =5. The two binary likelihood methods (ML(Y) and ML(Z)) are omitted as the least accurate in their class. Note also that we plot absolute rather than relative differences

because some of the errors under the true model are very small, and will lead to unstable ratios. One can see that for target localization, the performance of both ML(E) and LVDF is very robust, whereas ODF performs somewhat worse, though the differences are small. For signal estimation, methods that use at least some energy information (ML(E), HEM(Y), and HEM(Z)) do well, whereas both methods based on binary decisions only (EM(Y) and EM(Z)) deteriorate more. These differences may be larger for more drastically different models.

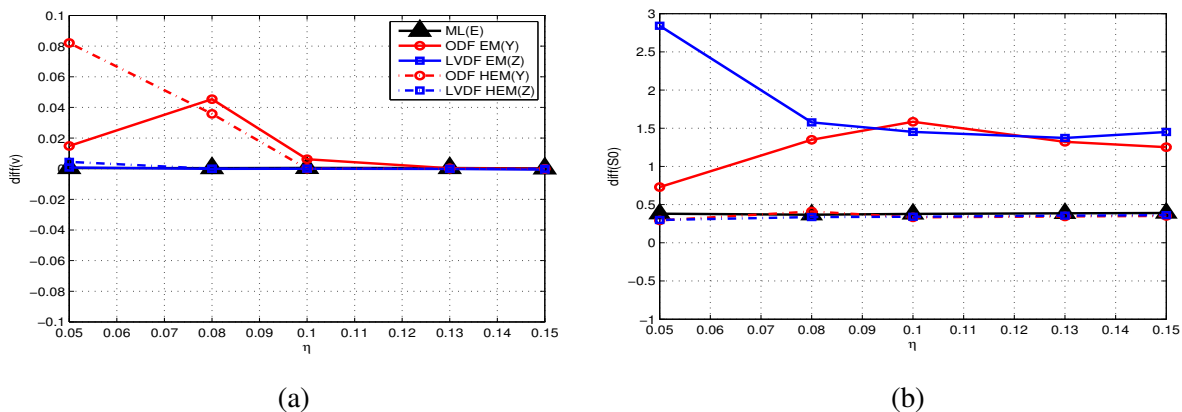


Figure 4: True model M2 misspecified as M1 with SNR=5. (a) The difference between average distances from true v for misspecified and true models; (b) The difference between RMSE of \hat{S}_0 for misspecified and true models.

We next look at another type of potential misspecification, which is when the noise distribution is misspecified. In the simulation, M1 generates the signal and is used by the algorithms, but the noise comes from a t -distribution with 3 degrees of freedom, while Gaussian distribution is assumed by the algorithms. Note that this t -distribution has significantly heavier tails than the Gaussian, but the false alarm rate of the individual sensor is fixed at the same value $\gamma = 0.1$, and the two distributions are scaled to have the same variance. In this case, both versions of the LVDF algorithm perform very well and prove the most robust. The ODF errors also remain similar, with somewhat erratic behavior for smaller targets where ODF may perform better under a misspecified noise model. The energy-based MLE is the most sensitive to distribution misspecification, as one would expect.

3.5 Confidence Region Estimation

The coverage and area of bootstrap confidence regions for the target location v are summarized in Tables 5 and 6. The setting was a target located at $v = (1/4, 1/4)$, with sensor false alarm rate $\gamma = 0.1$ and target size determined by $\eta = 0.1$. Table 5 gives the coverage for the various methods for models M1 and

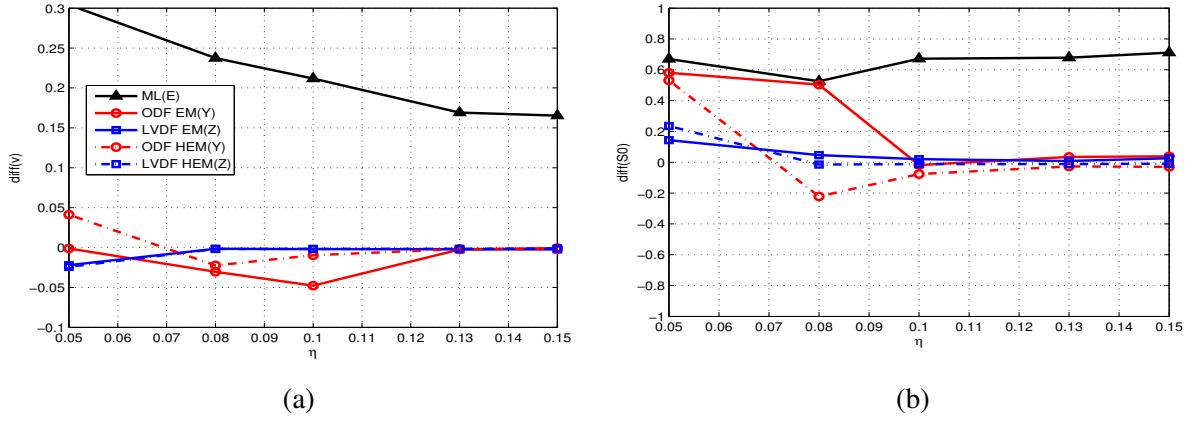


Figure 5: True noise distribution t_3 misspecified as Gaussian, with SNR=5. (a) The difference between average distances from true v for misspecified and true models; (b) The difference between RMSE of \hat{S}_0 for misspecified and true models.

M2. It can be seen that the energy based ML algorithm, together with all variants of the LVDF algorithm except HML, approximately achieve the nominal level of 95% for all SNRs examined, whereas the ODF ones fall short, particularly for the more localized target of model M1. Table 6 gives the average area of the confidence regions. For SNR=2, the gold standard is outperformed by all of LVDF algorithms in all cases, and in some cases by some of the ODF methods as well. For larger SNRs, ML(E) is the best. The HML(Z) method gives small confidence regions, but taken coverage probabilities into account, we conclude that of decision-based methods, EM(Z) produces the most reliable confidence regions, with HEM(Z) also showing good performance. We note that the bootstrap procedure does require extra computing time, but it typically takes less than a minute to compute a confidence region on an ordinary PC.

3.6 Computational Costs

Some computational cost estimates are shown in Figure 6, which give the distribution of the number of iterations over 100 replications of the ML (direct optimization), EM, and HEM algorithms. One iteration for each method takes approximately the same amount of time, so comparing the number of iterations directly provides a reasonable estimate of relative computational costs. Comparisons are shown for a grid deployment with $\eta = 0.1$ and SNR = 5. Note that the EM algorithms were stopped at 300 iterations if they did not achieve convergence using a 10^{-4} tolerance. On average, the LVDF algorithms converge faster than their ODF counterparts; however, it takes the optimization about 1/10 of the iterations to converge on average, compared to the EM versions (recall that the M-step requires a numerical optimization; the number of iterations shown for EM is the sum of the optimization iterations at each M-step and the EM

SNR	ML(E)	ML(Y)	EM(Y)	HML(Y)	HEM(Y)	ML(Z)	EM(Z)	HML(Z)	HEM(Z)
Model M1, grid deployment									
2	96	76	47	62	49	94	97	77	90
5	93	72	78	84	96	97	96	98	95
10	93	87	97	82	98	91	96	99	98
Model M2, grid deployment									
2	97	81	62	54	73	97	95	46	95
5	97	90	91	92	87	91	94	88	90
10	93	99	88	96	94	87	94	88	96
Model M1, random deployment									
2	92	61	22	50	28	95	91	55	96
5	94	65	50	63	66	93	77	82	91
10	94	78	85	59	97	91	91	85	95
Model M2, random deployment									
2	95	55	59	26	58	96	94	63	97
5	92	91	93	55	86	76	90	81	91
10	91	97	93	78	94	91	95	92	97

Table 5: Number of 95% confidence regions (out of 100) containing the true target location.

SNR	ML(E)	ML(Y)	EM(Y)	HML(Y)	HEM(Y)	ML(Z)	EM(Z)	HML(Z)	HEM(Z)
Model M1, grid deployment									
2	1.0000	2.3195	0.2692	0.6120	0.1977	0.3041	0.3204	0.0415	0.2411
5	0.0029	2.1307	0.1614	0.7897	0.0510	0.0491	0.0070	0.0212	0.0019
10	0.0003	2.1063	0.0497	0.6587	0.0010	0.0089	0.0037	0.0459	0.0004
Model M2, grid deployment									
2	0.8738	1.9541	0.1667	0.0583	0.0365	0.3745	0.1768	0.0074	0.1506
5	0.0016	0.8758	0.0106	0.0202	0.0014	0.0086	0.0066	0.0013	0.0015
10	0.0003	0.3162	0.0031	0.0061	0.0003	0.0134	0.0049	0.0003	0.0003
Model M1, random deployment									
2	1.0294	2.0788	0.2313	0.5898	0.1495	0.7190	0.3516	0.2851	0.2850
5	0.0027	2.0278	0.2754	0.6529	0.1773	0.1928	0.0294	0.2949	0.0159
10	0.0005	1.9758	0.1605	0.6260	0.0401	0.0735	0.0077	0.2707	0.0007
Model M2, random deployment									
2	0.8657	1.8346	0.2374	0.1478	0.0653	0.6836	0.2319	0.1445	0.2329
5	0.0019	1.0153	0.0378	0.2354	0.0024	0.0828	0.0128	0.1850	0.0027
10	0.0004	0.6465	0.0031	0.1237	0.0004	0.0482	0.0053	0.1035	0.0005

Table 6: Average area of the 95% confidence regions.

iterations). Given the significantly higher accuracy of the EM algorithms, this represents the usual trade-off between computational complexity and accuracy. The hybrid EM algorithms converge faster than their EM counterparts as one would expect.

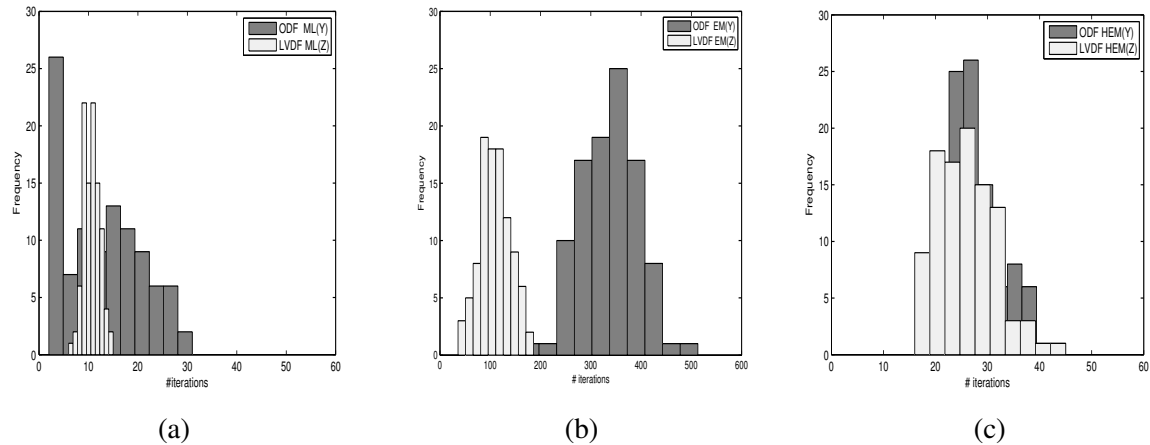


Figure 6: Distribution of iterations to convergence. (a) ML algorithms; (b) EM algorithms; (c) HEM algorithms.

4 Tracking of Moving Targets

In this section, we examine performance of the various algorithms when it comes to tracking the position of a target moving through the monitoring region R . The problem of tracking targets over time has received considerable attention in the literature (Wang et al., 2005; Chen et al., 2003, 2005; Yang et al., 2006). The focus has been primarily on scheduling algorithms which allow the network to activate only sensors close to the target and save energy by keeping other sensors idle, and various approaches to optimizing this process have been proposed. Although this is an important issue, it is beyond the scope of this paper. In the remainder, we focus on temporally fusing the binary decisions available from all the sensors to improve target localization and tracking. This general idea of temporal fusion can be easily adapted to situations where only a subset of the sensors provide data at any given time point.

The setting we investigate here is the following: sensors record energy readings at time slots $t = 1, 2, \dots$ and make decisions at each time slot. In our previous work (Katenka et al., 2006), temporal decision fusion was introduced and shown to lead to significant improvements in terms of detection probability for both stationary and moving targets. Temporal decision fusion combines decisions over time using an exponentially weighted moving average scheme. The same idea can be extended to fuse information about the location of

the target over time. Specifically, we have $\tilde{v}_0 = \hat{v}_0$, and

$$\tilde{v}_t = \lambda \hat{v}_t + (1 - \lambda) \tilde{v}_{t-1}, \quad t = 1, 2, \dots \quad (17)$$

where \hat{v}_t denotes the location estimate obtained from measurements at time t . A more detailed discussion on the use of exponentially weighted moving average and the choice of λ can be found in Katenka et al. (2006). An analogous scheme can be used for the signals' magnitude. In addition, we use \tilde{v}_{t-1} as a starting value for localization at time t , instead of the centroid of positive decisions at time t .

The first scenario examined using temporal decision fusion is of a target moving from west to east through the middle of the monitoring region $R = [0, 1]^2$. A 20×20 WSN is deployed on a grid and the target's speed is set to one hop per time slot; hence, it takes the target 20 time periods to cross R . The remaining parameters are set to $S_0 = 2$, $\lambda = 0.5$ and $\eta = 0.1$. The trajectories estimated by the various algorithms for a moderately noisy environment with $\text{SNR}=2$ under model M1 are shown in Figure 7, along with average error as a function of time. As expected from the previous results, the LVDF algorithms track the target particularly well, whereas the ODF one exhibit the worst performance. It is worth noting that ML(E) suffers more from errors at the beginning of the monitoring period. For larger SNRs (results not shown), ML(E) catches up and eventually outperforms LVDF, whereas the ODF algorithms continue to exhibit an inferior performance. Similar findings (not shown) have been obtained for estimating the (constant) signal of the moving target, as well as for random sensor deployments.

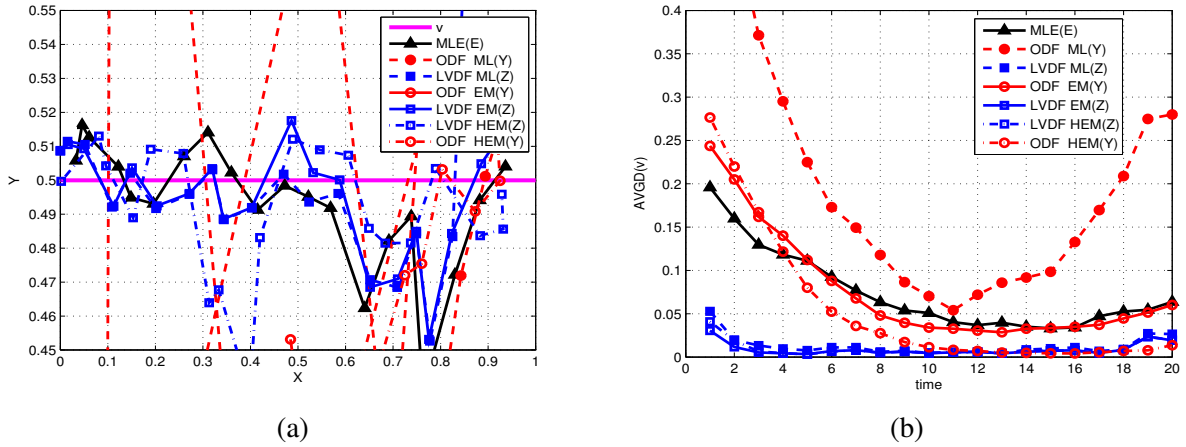


Figure 7: Tracking a moving target: (a) Estimated target trajectory for a single realization; (b) Average distance from the true trajectory (over 100 replications).

In the second scenario, the target is positioned at $v = (0.25, 0.25)$ and does not move. However, the signal's

amplitude evolves over time according to the model $S_0(t) = 2/(1 + 0.01(t - 10)^2)$, with $t = 1, 2, \dots, 20$, and $\sigma = 0.4$ held constant, which results in SNR going from 2.5 at time 0 to 5 at time 10 and down to 2.5 again by time 20. Other settings are the same as in the first scenario (grid deployment, $\eta = 0.1$, signal model M1, $\lambda = 0.5$). The performance of the algorithms is summarized in Figure 8. Once again, the LVDF gives the best results at all points in time, followed by the gold standard and the ODF algorithms. At higher SNRs, the ML(E) algorithm becomes the best method, a result consistent with previous ones.

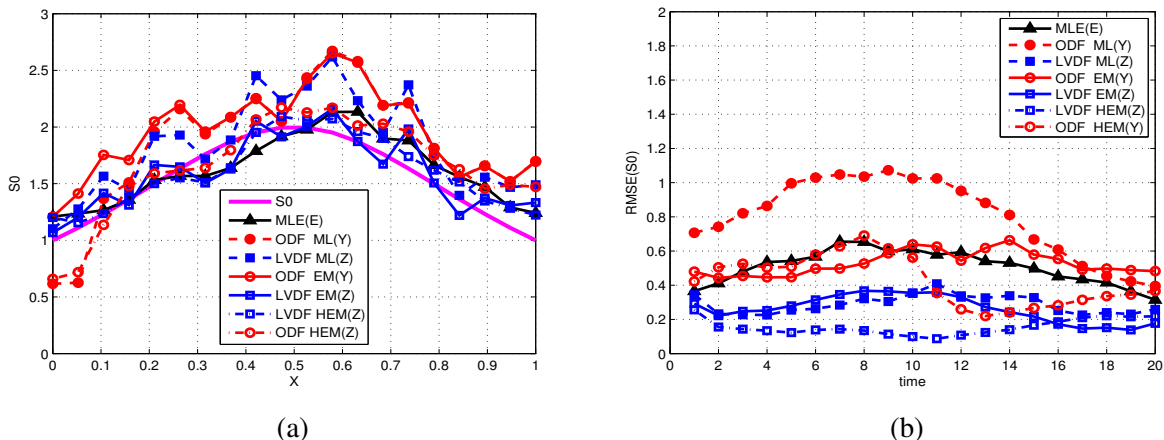


Figure 8: Tracking a stationary target with evolving signal: (a) Estimated signal amplitude for a single realization; (b) Average RMSE of temporally fused S_0 (over 100 replications).

5 Concluding Remarks

In this paper, the problem of estimating the location of a target and its signal amplitude by a WSN was examined and a number of algorithms introduced and compared. The results show that the LVDF algorithms provide highly accurate estimates, and in the presence of high levels of noise they consistently outperform the energy based ML algorithm, chiefly due to their highly localized positive decisions. The hybrid expectation-maximization algorithm based on LVDF decisions, where only the energies from sensors that made positive decisions are transmitted, was shown to provide essentially the same accuracy for target localization and signal magnitude estimation as the algorithm based on all the energies. Since LVDF makes substantially fewer positive decisions than ODF, this algorithm offers a substantial communications cost saving.

The presence of multiple targets in the sensed region R introduces several new challenges, such as estimating their number and location, classifying them according to type and tracking them over time. A simple way

to extend the localization algorithms in this paper to a situation with multiple targets is to apply a clustering method to the locations of positive decisions, as a pre-processing steps. The number of clusters may be assumed known (if the number of targets is known) or estimated in the process of clustering. Then any of the localization and signal estimation algorithms described above can be applied to each cluster separately. More integrated approaches are also possible. These tasks offer a wealth of problems for statisticians and constitute topics of current research.

Acknowledgments: The authors would like to thank the Editor, an Associate Editor and two anonymous referees for many useful comments and suggestions.

References

- Abdel-Samad, A. and Tewfik, A. (1999), "Search Strategies for Radar Target Localization," in *Proceedings of International Conference on Image Processing*, vol. 3, pp. 862–866.
- Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., and Cayirci, E. (2002), "Wireless sensor networks: a survey," *Computer Networks*, 38, 393–422.
- Besag, J. (1986), "On the Statistical Analysis of Dirty Pictures," *Journal of the Royal Statistical Society: Series B*, 48, 259–302.
- Blatt, D. and Hero, A. (2006), "Energy-Based Sensor Network Source Localization via Projection Onto Convex Sets," *IEEE Transactions on Signal Processing*, 54, 3614–3619.
- Cardell-Oliver, R., Smettem, K., Kranz, M., and Mayer, K. (2005), "A reactive soil moisture sensor network: Design and field evaluation," *International Journal of Distributed Sensor Networks*, 1, 149–163.
- Chen, J., Hudson, R., and Yao, K. (2002), "Maximum-Likelihood Source Localization and Unknown Sensor Location Estimation for wideband signals in the near-field," *IEEE Transactions on Signal Processing*, 50, 1843–1854.
- Chen, T., Liao, W., Huang, M., and Tsai, H. (2005), "Dynamic Object Tracking in Wireless Sensor Networks," in *Proceedings of 13th IEEE International Conference on Networks, 2005*, vol. 1.
- Chen, W., Hou, J., and Sha, L. (2003), "Dynamic Clustering for Acoustic Target Tracking in Wireless Sensor Networks," in *Proceedings of 11th IEEE International Conference on Network Protocols*.

- Clouqueur, T., Ramanathan, P., Saluja, K., and Wang, K. (2001), “Value-Fusion versus Decision-Fusion for Fault-tolerance in Collaborative Target Detection in Sensor Networks,” in *Proceedings of 4th Annual Conference on Information Fusion*, pp. TuC2/25–TuC2/30.
- Dempster, A., Laird, N., and Rubin, D. (1977), “Maximum Likelihood from Incomplete Data via the EM Algorithm,” *Journal of the Royal Statistical Society: Series B*, 39, 1–38.
- Efron, B. Tibshirani, R. (1994), *An Introduction to the Bootstrap*, CRC Press.
- Ermis, E. and Saligrama, V. (2006), “Detection and Localization in Sensor Networks Using Distributed FDR,” in *Proceedings of Conference on Information Sciences and Systems*.
- Estrin, D. (2006), “Wireless sensing systems: from eco-systems to human-systems,” in *Feedback and Dynamics in Nature Workshop held at the Grace Hopper Celebration of Women in Computing Conference*.
- Fitch, J., Coyle, E., and Gallagher, N. (1984), “Median filtering by threshold decomposition,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 32, 1183 – 1188.
- Gallagher, N., J. and Wise, G. (1981), “A theoretical analysis of the properties of median filters,” *IEEE Transactions on Acoustics, Speech, and Signal Processing [see also IEEE Transactions on Signal Processing]*, 29, 1136 – 1141.
- Hwang, H. and Haddad, R. (1995), “Adaptive median filters: new algorithms and results,” *IEEE Transactions on Image Processing*, 4, 499–502.
- Ji, X. and Zha, H. (2004), “Sensor Positioning in Wireless Ad-hoc Sensor Networks with Multidimensional Scaling,” in *Infocom*.
- Kaplan, L., Le, Q., and Molnar, P. (2001), “Maximum-Likelihood Methods for Bearings-Only Target Localization,” in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, pp. 3001–3016.
- Katenka, N., Levina, E., and Michailidis, G. (2006), “Local Vote Decision Fusion for Target Detection in Wireless Sensor Networks,” *IEEE Transactions on Signal Processing*, tentatively accepted by IEEE Transactions on Signal Processing.
- Kim, S., Pakzad, S., Culler, D., Demmel, J., Fennes, G., Glaser, S., and Turon, M. (2006), “Wireless sensor networks for structural health monitoring,” in *Proceedings of the 4th international conference on Embedded networked sensor systems*, New York, NY, USA: ACM Press, pp. 427–428.

- Li, D., Wong, K., Hu, Y., and Sayeed, A. (2002), "Detection, Classification, and Tracking of Targets," *IEEE Signal Processing Magazine*, 19, 17–29.
- Liang, G. and Yu, B. (2003), "Maximum Pseudo-likelihood Estimation in Network Tomography," *IEEE Transactions on Signal Processing*, 51, 2043–2053.
- Mainwaring, A., Polastre, J., Szewczyk, R., Culler, D., and Anderson, J. (2002), "Wireless sensor networks for habitat monitoring," in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, New York, NY, USA: ACM Press, pp. 88–97.
- Meesookho, C. and Narayanan, S. (2005), "Distributed Range Difference Based Target Localization in Sensor Network," in *Proceedings of the 39th Asilomar Conference on Signals, Systems and Computers*, pp. 205–209.
- Niu, R. and Varshney, P. (2004), "Target Location Estimation in Wireless Sensor Networks Using Binary Data," in *Conference on Information Sciences and Systems*.
- Noel, M., Joshi, P., and Jannett, T. (2006), "Improved Maximum Likelihood Estimation of Target Position in Wireless Sensor Networks using Particle Swarm Optimization," in *Proceedings of 3rd International Conference on Information Technology: New Generations*.
- Padhy, P., Martinez, K., Riddoch, A., Ong, H. L. R., and Hart, J. K. (2005), "Glacial environment monitoring using sensor networks," in *Proceedings of Real-World Wireless Sensor Networks*, ACM Press.
- Sheng, X. and Hu, Y. (2003), "Energy Based Acoustic Source Localization," in *Proceedings of 3rd International Workshop on Information Processing in Sensor Networks*, pp. 286–300.
- Sohrabi, K., Gao, J., Ailawadhi, V., and Pottie, G. J. (2000), "Protocols for Self-Organization of a Wireless Sensor Network," *IEEE Personal Communications*, 7, 16–27.
- Tukey, J. (1977), *Exploratory Data Analysis*, Addison-Wesley.
- Wang, H., Yao, K., and Estrin, D. (2005), "Information-Theoretic Approaches for Sensor Selection and Placement in Sensor Networks for Target Localization and Tracking," *Journal of Communication and Networks*, 7, 438–448.
- Wang, W., Srinivasan, V., Wang, B., and Chua, K. (2006), "Coverage for Target Localization in Wireless Sensor Networks," in *Proceedings on Information Processing in Sensor Networks*, pp. 118–125.

- Xu, N., Rangwala, S., Chintalapudi, K., Ganesan, D., Broad, A., Govindan, R., and Estrin, D. (2004), “A wireless sensor network for structural monitoring,” in *Proceedings of the 4th international conference on Embedded networked sensor systems*, New York, NY, USA: ACM Press.
- Yang, L., Feng, C., Rozenblit, J., and Qiao, H. (2006), “Adaptive Tracking in Distributed Wireless Sensor Networks,” in *Proceedings of 13th Annual IEEE International Symposium and Workshop on Engineering of Computer Based Systems*.
- Zeng, B. (1994), “Convergence properties of median and weighted median filters,” *IEEE Transactions on Signal Processing*, 42, 499–502.
- Zou, Y. and Chakrabarty, K. (2003), “Target Localization Based on Energy Considerations in Distributed Sensor Networks,” *IEEE Transactions on Signal Processing*, 5, 51–58.