

# System-level Calibration for Fusion-based Wireless Sensor Networks

Rui Tan<sup>1</sup> Guoliang Xing<sup>1</sup> Zhaohui Yuan<sup>2</sup> Xue Liu<sup>3</sup> Jianguo Yao<sup>4</sup>

<sup>1</sup>Michigan State University, USA    <sup>2</sup>Wuhan University, P.R. China

<sup>3</sup>University of Nebraska-Lincoln, USA    <sup>4</sup>Northwestern Polytechnical University, P.R. China

## Abstract

Wireless sensor networks are typically composed of low-cost sensors that are deeply integrated in physical environments. As a result, the sensing performance of a wireless sensor network is inevitably undermined by biases in imperfect sensor hardware and the noises in data measurements. Although a variety of calibration methods have been proposed to address these issues, they often adopt the device-level approach that becomes intractable for moderate- to large-scale networks. In this paper, we propose a two-tier system-level calibration approach for a class of sensor networks that employ data fusion to improve the sensing performance. In the first tier of our calibration approach, each sensor learns its local sensing model from noisy measurements using an online algorithm and only transmits a few model parameters. In the second tier, sensors' local sensing models are then calibrated to a common system sensing model. Our approach fairly distributes computation overhead among sensors and significantly reduces the communication overhead of calibration. Based on this approach, we develop an optimal model calibration scheme that maximizes the target detection probability of a sensor network under bounded false alarm rate. Our approach is evaluated by both experiments on a testbed of TelosB nodes and extensive simulations based on data traces collected in a real vehicle detection experiment. The results demonstrate that our system-level calibration approach can significantly boost the detection performance of sensor networks in the scenarios with low signal-to-noise ratios.

## 1 Introduction

Wireless sensor networks (WSNs) are increasingly deployed for mission-critical applications such as target detection [3, 17], localization [12, 16], and security surveillance [7, 9]. In these applications, low-cost sensors are deeply integrated in physical environments and hence often suffer from significant performance variations. In particular, the measurement inaccuracy caused by hardware drift [2, 15], complex deployment terrain, and random noise [2] must be dealt with properly before sensor data can be meaningfully interpreted by users. A common so-

lution is to calibrate the sensing characteristics of sensors. In traditional calibration approaches, the sensor readings are mapped to the true value by a calibration function obtained under controlled environments [15] or from manufacturer's specification. However, such a *device-level* calibration scheme fails to account for the post-deployment factors such as non-ideal environmental conditions and hardware aging. Moreover, the scheme becomes intractable when the network scales to tens or hundreds of sensors.

Different from the device-level calibration, *system-level* calibration aims to optimize the overall system performance by tuning the sensing parameters of all the sensors in a network. System-level calibration often incurs significantly lower overhead by taking advantage of the knowledge about how the local information gathered by individual sensors is processed in the network. In this paper, we aim to devise system-level calibration algorithms for WSNs that adopt data fusion schemes for information processing. Data fusion [20] is a widely adopted signal processing technique that can improve the system sensing performance by jointly considering the measurements of multiple sensors. In practice, various data fusion schemes have been employed by sensor network systems for target detection [9], localization [12], and classification [5]. The major challenge of calibrating fusion-based networks is to understand and exploit the complex correlation between system performance and the characteristics of individual sensors. In particular, although the system performance of fusion-based networks is tightly coupled with the measurements of multiple sensors, system-level calibration must avoid centralized data collection and processing because of the resource constraints of low-cost sensors.

In this paper, we propose a novel *two-tier* system-level calibration approach for fusion-based WSNs. In the first tier, each sensor learns its *local* sensing model using in-place measurements, and only transmits a few model parameters to the fusion head node. In the second tier, the fusion head calibrates each sensor's model to a common sensing model. The key advantage of this approach is that most processing is performed locally by individual sensors resulting in low communication overhead. Several challenges must be addressed for realizing such a two-tier approach.

First, the measurements of sensors are often corrupted by random noise from physical environment and sensor hardware. Model learning using noisy data is challenging for low-cost sensors with limited computation and memory resources. At the same time, the local sensing model must accurately preserve the systematic bias of a sensor that is needed for system-level calibration. Moreover, in the second-tier calibration, the correlation between biased local sensing models and the overall system performance must be carefully considered in order to achieve the optimal system performance at run time.

We make the following major contributions in this paper.

- We propose a novel two-tier calibration approach for fusion-based WSNs. Our approach fairly distributes the computation overhead among individual sensors. Moreover, each sensor only needs to transmit little information in order to achieve the optimal system-level calibration.
- We formally formulate the problem of system-level calibration for target detection based on the two-tier approach. A linear regression algorithm is proposed to learn the local sensing model at each sensor. The algorithm processes sensor measurements in an online fashion and thus incurs little computation and memory overhead. We then develop an optimal solution that calibrates biased local sensing models and meanwhile the system detection probability is maximized under bounded false alarm rate.
- We evaluate our approach through both experiments on a testbed of TelosB motes and extensive simulations based on real data traces collected in a real vehicle detection experiment [5]. Compared with several baseline approaches, our system-level calibration approach significantly boosts the detection performance of WSNs under a wide range of realistic settings.

The rest of this paper is organized as follows. Section 2 reviews related work. Section 3 introduces the motivation and preliminaries. Section 4 presents the two-tier calibration approach and the problem formulation. The local model estimation and system-level model calibration are in Section 5 and 6, respectively. Section 7 and Section 8 present testbed experiments and extensive simulations, respectively. Section 9 concludes the paper.

## 2 Related Work

Sensor calibration is a fundamental problem in WSNs. Previous literature on calibration [2, 6, 11, 13–15, 21] can be categorized into *micro* and *macro* approaches. The *micro*-calibration refers to the method that tunes each individual sensor to output accurate readings. In [15], each chemical sensor is carefully calibrated in laboratory to obtain the mapping from its reading to the true value. In [13], an uncalibrated sensor calibrates itself when encountering a cal-

ibrated sensor. In [11], each sensor’s location is calibrated using the position information of a subset of nodes. In contrast, the *macro*-calibration [2, 6, 21] focuses on optimizing the overall system performance. In [6], the biases of light sensors are estimated by solving a group of equations that integrate all sensors’ measurements. Similarly, in [21], the parameters of ranging sensors are estimated by regression based on pair-wise range measurements. In [2], the calibration function, which maps two sensors’ outputs, is adjusted so that the consistency among a group of sensors is maximized. These macro-calibration approaches require the transmission of all or partial raw measurements to a central node for computing the calibration parameters. The approach presented in this paper belongs to macro-calibration. However, different from the above works, our approach allows each sensor to estimate a local sensing model and only transmit the model parameters for system-level calibration, which hence leads to significant reduction in communication overhead.

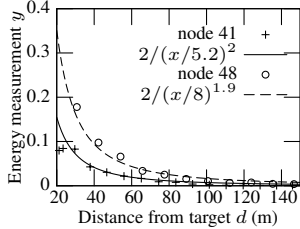
Data fusion [20] has been proposed as an effective signal processing technique to improve the system-wide performance in WSNs. Several data fusion schemes have been employed by sensor network systems designed for surveillance applications [5, 7, 9, 12, 17]. In our earlier work [18, 22], we study the impact of data fusion on spatial and temporal coverage of large-scale sensor networks. In [19], we calibrate the parameters of data fusion such that a sensor network adapts to various dynamics. In this paper, we aim to calibrate the sensor measurements such that the sensors have an identical sensing model that optimizes the system detection performance.

## 3 Problem Statement and Preliminaries

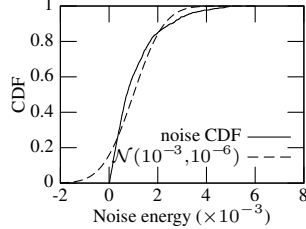
We first introduce the problem of system-level calibration for data fusion in Section 3.1, then describe the technical preliminaries including sensor measurement and data fusion models in Section 3.2 and 3.3, respectively.

### 3.1 Problem Statement

Sensor calibration refers to the process of identifying and correcting *systematic errors* (bias) of sensors. Various factors attribute to the sensor bias. First, manufacturing processes inevitably introduce variation in electric characteristics of sensor circuits. Second, the measurements of sensors are largely affected by the environment of deployment. For instance, in the scenario of vehicle detection [5, 7, 9], complex terrain often causes sensors to yield different sensitivity. In this paper, we do not distinguish the sensor bias due to different causes as it is often infeasible for doing so in large-scale deployments. In addition to systematic bias, random noise from environment and sensor hardware also causes inaccurate measurements. However, random noise often follows certain probabilistic distribution (*e.g.*, Gaus-



**Figure 1. Normalized energy measurement vs. the distance from the vehicle.**



**Figure 2. The CDF of noise energy. The dashed line is the CDF of  $\mathcal{N}(0.001, 0.001^2)$ .**

sian). Our sensor calibration approach exploits such a property and is resilient to the influence of random noise.

We use an example based on the real data traces from the DARPA SensIT vehicle detection experiments [5] to illustrate the variation in sensors’ characteristics. In the experiments, 75 WINS NG 2.0 nodes are deployed to detect military vehicles driving through the monitored area. Figure 1 plots the acoustic energy measurements taken by node 41 and 48, versus the distance from a driving vehicle. We can see from Figure 1 that the energy measurements of both sensors decay with the distance. However, they follow different decay functions. For instance, when the distance from the vehicle is 40 m, the measurements of node 41 and 48 are about 0.05 and 0.1, respectively. This example illustrates that sensors could yield considerably different sensing characteristics after deployment, and hence need calibration.

Calibration in WSNs is a challenging task as sensors are often inaccessible after being deployed. Moreover, the manual device-by-device calibration becomes intractable when the network scales to even tens of sensors. Conventional calibration methods often correct the readings of each sensor to a common benchmark (*e.g.*, readings of high-precision sensors or ground truth) [6, 15, 21]. However, such an approach often incurs high overhead in large-scale sensor networks. In contrast, we adopt a system-level approach that calibrates the sensing models of sensors to optimize the system-wide performance of a network. In particular, we focus on the calibration problem in a class of sensor networks that employ *data fusion* to improve the system performance. Existing data fusion literature [3, 12, 16] often assumes identical sensing characteristics for all sensors and hence cannot be directly used in uncalibrated networks. We aim to devise post-deployment calibration algorithms that can optimize the data fusion performance of WSNs.

### 3.2 Sensor Measurement Model

The energy of many physical signals (*e.g.*, acoustic and electromagnetic) attenuates with the distance from the signal source. Suppose sensor  $i$  is  $d_i$  meters away from the target that emits a physical signal with the source energy of

$S$ . The attenuated signal energy  $s_i$  at the position of sensor  $i$  follows the power law decay model given by

$$s_i = \frac{S}{(d_i/r_i)^{k_i}}, \quad (1)$$

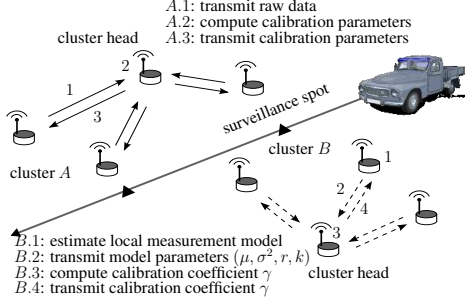
where  $r_i$  is the *reference distance* determined by sensor shape,  $k_i$  is the *decay factor* which typically ranges from 1.0 to 5.0 [8]. We note that the energy of most mechanical and electromagnetic waves follows the power law decay during propagation. In particular, in ideal open space, inverse-square law (*i.e.*,  $k_i = 2$ ) applies to various physical signals such as sound and radiation. However, in practice, the reference distance and decay factor vary with sensors because of the reasons discussed in Section 3.1.

The sensor measurements are contaminated by additive random noise. Depending on the hypothesis that the target is absent ( $H_0$ ) or present ( $H_1$ ), the signal energy measurement of sensor  $i$ , denoted by  $y_i$ , is given by  $y_i|H_0 = n_i$  or  $y_i|H_1 = s_i + n_i$ , where  $n_i$  is the energy of noise experienced by sensor  $i$ . In practice, a sensor’s measurement is computed by the average over a number of ( $\geq 30$ ) samples [5, 16]. From the central limit theorem, the noise energy  $n_i$  follows the normal distribution [16], formally,  $n_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$ , where  $\mu_i$  and  $\sigma_i^2$  are the mean and variance of  $n_i$ . We assume that the noises,  $\{n_i|\forall i\}$ , are spatially independent across sensors.

The above signal decay and noise models have been widely assumed in the literature of signal processing [3, 16, 20] and also have been empirically verified [8, 12]. In Figure 1, the curves are the fittings of the data points to the signal decay model in (1). We can see that the model parameters, *i.e.*,  $r_i$  and  $k_i$ , vary with sensors. For instance, the reference distances for node 41 and 48 (*i.e.*,  $r_{41}$  and  $r_{48}$ ) are 5.2 m and 8 m, respectively. Figure 2 plots the cumulative distribution function (CDF) of background noise measured by a sensor in the SensIT experiments [5]. We can see that the CDF of measured noise well matches the CDF of the normal distribution  $\mathcal{N}(0.001, 0.001^2)$ .

### 3.3 Multi-sensor Data Fusion Model

Data fusion [20] has been proposed as an effective signal processing technique to improve the system performance of sensor networks. A sensor network that employs data fusion is often organized into clusters. The cluster head is responsible for making a decision regarding the presence of target by fusing the information gathered by member sensors. As sensors can only carry out limited processing due to resource constraints, we adopt a basic data fusion scheme [20] as follows. Each cluster head makes the detection decision by comparing the sum of measurements reported by member sensors against a detection threshold  $T$ . Suppose there are  $N$  member sensors in a cluster, the sum of measurements, denoted by  $Y$ , is  $Y = \sum_{i=1}^N y_i$ . If  $Y \geq T$ , the



**Figure 3. Two system-level calibration schemes. Cluster A and B run centralized and two-tier calibration algorithms, respectively.**

cluster head decides  $H_1$ ; otherwise, it decides  $H_0$ . Such a basic data fusion model has been widely employed in previous work [3, 19].

The detection of a target is inherently stochastic due to random noises in sensor measurements. The system detection performance is jointly characterized by two metrics, namely, the false alarm rate ( $P_F$ ) and the detection probability ( $P_D$ ).  $P_F$  is the probability of making a positive decision when *no* target is present, and  $P_D$  is the probability that a present target is correctly detected. Under the aforementioned data fusion model,  $P_F$  and  $P_D$  are given by  $P_F = \mathbb{P}(Y \geq T | H_0)$  and  $P_D = \mathbb{P}(Y \geq T | H_1)$ , respectively.

## 4 Overview of Approach

In this section, we first present the architecture of our system-level calibration approach, and then formally formulate the calibration problem as a constrained optimization problem.

### 4.1 System Architecture

Although our calibration approach can be applied to various scenarios of event detection, we use an example of vehicle surveillance to illustrate the basic idea of our approach. In order to calibrate the sensors that are deployed to monitor vehicles as shown in Figure 3, a vehicle acts as the target and drives through the monitored region. The network then calibrates the sensors based on their in-place measurements of the *controlled target* such that the system’s performance of detecting targets at run time is maximized. Note that our approach does not require the knowledge of the signal profile of the controlled target as long as its location information is available. We assume that sensors and the controlled target know their positions through GPS or a localization service in the network.

There are two possible schemes for such a calibration problem. A straightforward scheme works in a centralized fashion. For example, in Figure 3, each member sensor in cluster A sends the raw measurements to the cluster head,

which computes the calibration parameters for each sensor. However, a large number of measurements are often required to accurately characterize the sensing model of a sensor such as the signal decay model in (1). Hence, such a scheme will introduce high communication overhead.

The second scheme is based on a *two-tier* architecture adopted in this paper. In Figure 3, each sensor in cluster B learns its sensing model which is characterized by a few model parameters based on the raw measurements, and only transmits the model parameters to the cluster head. The cluster head then calibrates each sensor’s sensing model such that the expected system performance of detecting targets at run time is maximized. Such a scheme not only distributes the computation load to each sensor, but also avoids costly transmission of raw data.

Specifically, the two-tier architecture consists of *local calibration* and *system-level calibration*. In the *local calibration* (*i.e.*, the first tier), each sensor  $i$  estimates its noise and signal decay models using in-place measurements of the controlled target. The sensors periodically (*e.g.*, every 5 seconds) measure the energy of signal emitted by the controlled target that appears in the deployment region. To reduce the impact of noise, each sensor takes a number of measurements when the target is at a certain position. Several parameter estimation methods such as maximum likelihood estimation can be used to estimate the signal decay model. In this paper, we adopt an online least squares method due to its low computation and memory overhead. Each sensor only transmits the estimated sensing model parameters to the cluster head.

In the *system-level calibration* (*i.e.*, the second tier), the cluster head computes the calibration parameters for each sensor such that the overall system performance is maximized. Due to the resource constraints of sensors, we adopt a simple linear calibration scheme as follows. The calibrated measurement of sensor  $i$ , denoted by  $\hat{y}_i$ , is given by  $\hat{y}_i = \gamma_i \cdot y_i$ , where  $\gamma_i$  is the *calibration coefficient* of sensor  $i$ . The objective is to determine the calibration coefficients of all sensors involved in the data fusion, such that the system detection performance is maximized.

In this paper, we are only concerned with the detection performance at a number of fixed geographical locations which are referred to as *surveillance spots*. Surveillance spots can be chosen before network deployment according to application requirements or identified by the network autonomously after deployment. For instance, in Figure 3, the surveillance spots can be chosen along the roads in the monitored area. For each surveillance spot, the cluster head computes a calibration coefficient  $\gamma_i$  for each sensor  $i$  such that the detection performance at the surveillance spot is maximized. We note that our calibration approach can be easily extended to dynamic surveillance spot, *i.e.*, the estimated position of the target at run time, which can be ob-

tained by existing target localization algorithms [12, 16].

At run time, the calibrated network detects target as follows. Each sensor  $i$  sends its measurement  $y_i$  to the cluster head. To detect whether a target is present at a particular surveillance spot, the cluster head compares the sum of calibrated measurements, *i.e.*,  $\sum_{i=1}^N \gamma_i \cdot y_i$ , against the threshold  $T$  to make a detection decision regarding whether a target is present at the surveillance spot.

## 4.2 Problem Formulation

The objective of local calibration at each sensor is to learn the Gaussian noise model and the signal decay model in (1), which are characterized by  $\mu_i$ ,  $\sigma_i^2$ ,  $r_i$ , and  $k_i$ . For a single sensor  $i$ , the input is a collection of data pairs  $(d_i, y_i)$ , *i.e.*, the measurement  $y_i$  when the controlled target is  $d_i$  meters away. The major challenge is how to learn these models using noisy measurements. To cope with the noise, sensor  $i$  samples a number of energy measurements when the controlled target is at a certain position. These measurements can be used to compute a statistic such as average so that the impact of noise is mitigated. The local calibration problem is formally formulated as follows.

**Problem 1.** *Suppose at time step  $t$ , sensor  $i$  is  $d_i(t)$  meters away from the controlled target and takes  $M$  measurements  $\mathcal{Y}_i(t) = \{y_i[1], y_i[2], \dots, y_i[M]\}$ . How to compute the noise and signal decay models  $(\mu_i, \sigma_i^2, r_i, k_i)$  of sensor  $i$  using its noisy measurements  $\{d_i(t), \mathcal{Y}_i(t) | t = 1, 2, \dots\}$ ?*

Problem 1 is an over-provision parameter estimation problem [1], as a large number of measurements are usually available to train the models. However, the parameter estimation method must incur low computation and memory overhead due to the resource constraints of sensors. The solution to Problem 1 is discussed in Section 5.

With the model parameters obtained in each sensor's local calibration, the system-level calibration aims to calibrate each sensor's sensing model to maximize the system detection performance at each surveillance spot. We focus on a single surveillance spot in the rest of this paper.

When the target is present at the surveillance spot, the calibrated measurement of sensor  $i$  is given by  $\hat{y}_i | H_1 = \gamma_i \cdot y_i | H_1 = \gamma_i \cdot s_i + \gamma_i \cdot n_i$ , where  $s_i$  follows the signal decay model in (1) and  $n_i$  is the Gaussian noise. We denote the calibrated signal energy  $\hat{s}_i = \gamma_i \cdot s_i$ . Suppose sensor  $i$  is  $l_i$  meters from the surveillance spot. After the system-level calibration, the calibrated signal energy of each sensor should follow a *common* signal decay model given by

$$\hat{s}_i = \frac{\hat{S}}{(l_i/r)^k}, \quad (2)$$

where  $\hat{S}$  is the calibrated source energy,  $r$  and  $k$  are the common reference distance and decay factor for all calibrated sensors, respectively.

There exists a trade-off between the false alarm rate  $P_F$  and detection probability  $P_D$ . Specifically, a higher  $P_D$  is always achieved at the price of higher  $P_F$  [20]. Therefore, a common requirement of many systems is to maximize the  $P_D$  while the  $P_F$  is bounded. Our objective is to find the calibration coefficients to maximize the system detection performance, *i.e.*, to maximize the  $P_D$  subject to an upper bound of  $P_F$  specified by application. The system-level calibration problem can be formally formulated as follows.

**Problem 2.** *Suppose there are  $N$  sensors in a detection cluster. Given each sensor's sensing model  $\{\mu_i, \sigma_i^2, r_i, k_i | 1 \leq i \leq N\}$  and the distances from the surveillance spot  $\{l_i | 1 \leq i \leq N\}$ , to find a common signal decay model  $(\hat{S}, r, k)$  and a list of calibration coefficients  $\{\gamma_i | 1 \leq i \leq N\}$ , such that the detection probability  $P_D$  is maximized subject to that 1) the false alarm rate  $P_F$  is upper-bounded by  $\alpha$  where  $\alpha \in (0, 1)$ , and 2) the calibrated signal energy of each sensor follows the common model given in (2).*

In Problem 2, all local sensing models are calibrated to a common sensing model that maximizes the detection performance at the surveillance spot. Such an approach significantly simplifies the design of fusion-based WSNs as existing data fusion algorithms can be adopted without accounting for the differences in sensors' characteristics after calibration. In fact, existing data fusion literature [3, 12, 16] often assumes an identical sensing model for the same sensor modality. The optimal solution of Problem 2 is discussed in Section 6.

## 5 Online Local Calibration

In this section, we present our approach of local calibration. We propose a solution based on linear regression technique in Section 5.1 and its online improvement in Section 5.2.

### 5.1 Measurement Model Estimation

We first present how to estimate the noise model. When no target is present, sensor  $i$  only measures noise. Accordingly, each sensor can estimate the noise model using a number of measurements in the absence of target. In practice, the measurements can be treated as noises when the target is far enough away from the sensor. The noise model can be estimated by the sample mean and variance, respectively. Specifically,

$$\mu_i = \frac{1}{M} \sum_{j=1}^M y_i[j] | H_0, \quad \sigma_i^2 = \frac{1}{M-1} \sum_{j=1}^M (y_i[j] | H_0 - \mu_i)^2, \quad (3)$$

where  $M$  is the number of samples that are used to estimate the noise model.

Now we discuss how to estimate the signal decay model in (1). As the measurements are noisy, they cannot be directly used to estimate the signal decay model. A common approach reducing the impact of noise is to average multiple measurements [23]. However, such a method requires a large number of samples when the sensor experiences heavy noise. In the following, we propose an approach which exploits the relationship between the local detection probability and the noise distribution.

As described in Section 4.2, at time step  $t$ , sensor  $i$  takes  $M$  measurements (*i.e.*,  $\mathcal{Y}_i(t)$ ) when the controlled target is  $d_i(t)$  meters away from it. For each measurement  $y_i \in \mathcal{Y}_i(t)$ , sensor  $i$  makes a detection by comparing  $y_i$  against a threshold  $\eta$ . The detection probability at time step  $t$ , denoted by  $P_{Di}(t)$ , can be estimated by the ratio of the number of measurements that exceed  $\eta$  to  $M$ . The threshold  $\eta$  can be set to be the mean of  $\mathcal{Y}_i(t)$  to avoid the saturation of  $P_{Di}(t)$ , *i.e.*, the case where  $P_{Di}(t)$  equals 0 or 1. Hence, sensor  $i$  has a statistic  $P_{Di}(t)$  for each time step  $t$ . Sensor  $i$  then estimates the signal decay model by least squares techniques using the statistics  $\{P_{Di}(t), d_i(t) | t = 1, 2, \dots\}$  according to the relationship between the statistics and the signal decay model, which will be derived in this section.

The movement distance of the controlled target during consecutive  $M$  measurements is often small. For instance, in the SensIT experiments [5], the average speed of vehicles is 5 m/s. If the sensor samples  $M$  measurements in 0.5 seconds, the average moving distance is only 2.5 m and hence can be ignored as the distance between the vehicle and a sensor is usually tens of meters [5]. We note that the sensor's sampling rate is often high (*e.g.*, 4960 Hz for acoustic sensors in [5]) and hence  $M$  can be large enough to assure the statistical significance of  $P_{Di}$ .

We now derive the relationship between the local detection probability  $P_{Di}$  and the signal decay model. When the controlled target is present, the measurement of sensor  $i$  follows the normal distribution, *i.e.*,  $y_i | H_1 = s_i + n_i \sim \mathcal{N}(s_i + \mu_i, \sigma_i^2)$ . Therefore, under the aforementioned detection rule, the detection probability of sensor  $i$  is  $P_{Di} = \mathbb{P}(y_i \geq \eta | H_1) = Q\left(\frac{\eta - s_i - \mu_i}{\sigma_i}\right)$ , where  $\eta$  is the detection threshold and  $Q(\cdot)$  is the complementary CDF of the standard normal distribution, formally,  $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-t^2/2} dt$ . By replacing  $s_i$  with (1) and taking logarithmic transformation, we have

$$\ln(\eta - \mu_i - \sigma_i Q^{-1}(P_{Di})) = -k_i \ln d_i + \ln(S_0 r_i^{k_i}),$$

where  $S_0$  is the source energy of the controlled target and  $Q^{-1}(\cdot)$  is the inverse function of  $Q(\cdot)$ . For time step  $t$ , by letting  $z_i(t) = \ln(\eta - \mu_i - \sigma_i Q^{-1}(P_{Di}(t)))$ ,  $x_i(t) = -\ln d_i(t)$  and  $b_i = \ln(S_0 r_i^{k_i})$ , we have

$$z_i(t) = k_i \cdot x_i(t) + b_i. \quad (4)$$

Hence, the transformed data points  $\{z_i(t), x_i(t) | t = 1, 2, \dots\}$  should lie on the straight line with slope of  $k_i$  and  $z$ -intercept of  $b_i$ . The least squares fitting of the data points are given by

$$k_i = \frac{\text{cov}(x_i, z_i)}{\sigma_{x_i}^2}, \quad b_i = \bar{z}_i - k_i \cdot \bar{x}_i, \quad (5)$$

where  $\bar{z}_i$  and  $\bar{x}_i$  are the sample means,  $\text{cov}(x_i, z_i)$  and  $\sigma_{x_i}^2$  are the covariance and variance (with respect to  $x_i$ ) of the data points, respectively. As  $b_i = \ln(S_0 r_i^{k_i})$ , the reference distance  $r_i$  can be computed by the estimates of  $k_i$  and  $b_i$ , *i.e.*,  $r_i = \left(\frac{e^{b_i}}{S_0}\right)^{\frac{1}{k_i}}$ . The source energy of the controlled target,  $S_0$ , is usually unknown in practice. Therefore, we cannot compute the exact value of  $r_i$  from the linear fitting. However, our analysis in Section 6 shows that the system detection performance only depends on  $b_i$  and the exact value of  $r_i$  is unnecessary. Therefore, the measurement model of sensor  $i$  can be represented by a 4-tuple  $(\mu_i, \sigma_i^2, k_i, b_i)$ . Sensor  $i$  then transmits such a 4-tuple to the cluster head after its local calibration.

## 5.2 Online Model Estimation

In Section 5.1, we introduce the basic idea of estimating the measurement model by least squares technique. However, the estimation approach presented in Section 5.1 has to store all previous data points. In this section, we describe an online improvement of the linear fitting to estimate  $k_i$  and  $b_i$  in (4). The online algorithm works in a real-time fashion, *i.e.*, the estimates of  $k_i$  and  $b_i$  are updated for each time when sensor  $i$  obtains a data point. The estimates converge after the sensor obtains sufficient data points. We adopt the widely used recursive least squares estimation (RLSE) approach [1]. After reformulating (4) with vectors as  $z_i(t) = \phi_i^T(t) \cdot \theta_i$  where  $\phi_i(t) = [x_i(t), 1]^T$  and  $\theta_i = [k_i, b_i]^T$ , the recursive estimator of  $\theta_i$  is given by

$$\theta_i(t) = \theta_i(t-1) + L(t) (z_i(t) - \phi_i^T(t) \theta_i(t-1)), \quad (6)$$

$$L(t) = P(t-1) \phi(t) (1 + \phi^T(t) P(t-1) \phi(t))^{-1},$$

$$P(t) = (I - L(t) \phi^T(t)) P(t-1),$$

where  $I$  is the identity matrix. Equation (6) updates the estimates at each time step based on the error between the model output and the predicted output. The initial estimate  $\theta_i(0)$  can be set to be the best guesses of  $k_i$  and  $b_i$  so that the estimator can quickly converge. More detailed derivation and convergence proof of RLSE can be found in [1]. In each time step of the RLSE, only 22 float multiplications and 17 float additions are needed to update the estimator state. Moreover, only the estimator state, *i.e.*,  $\theta_{2 \times 1}$ ,  $L_{2 \times 1}$ , and  $P_{2 \times 2}$  need to be stored. Such computation and memory overhead is affordable for low-cost wireless sensors such as MICA2 motes [4].

---

**Algorithm 1** Online local calibration of a sensor
 

---

```

1: event command ESTIMATE_NOISE is received do
2:   sample  $M$  measurements,  $\{y[1], y[2], \dots, y[M]\}$ 
3:   compute  $\mu$  and  $\sigma^2$  using (3)
4: end event
5:
6: event command TRAINING_BEGIN is received do
7:   start a periodical timer cali_timer with timeout of  $W$  seconds
8: end event
9:
10: event cali_timer is fired do
11:   query the current position of the controlled target
12:    $x \leftarrow -\ln(d)$  where  $d$  is the distance from the controlled target
13:   sample  $M$  measurements,  $\{y[1], y[2], \dots, y[M]\}$ 
14:    $\eta \leftarrow \sum_{j=1}^M y[j]/M$ 
15:   /* compute the fraction of measurements that exceed  $\eta$  */
16:    $P_D \leftarrow \#(y[j] \geq \eta, j \in [1, M])/M$ 
17:    $z \leftarrow \ln(\eta - \mu - \sigma Q^{-1}(P_D))$ 
18:   update  $k$  and  $b$  using (6) with  $x$  and  $z$ 
19: end event
20:
21: event command TRAINING_END is received do
22:   stop cali_timer
23:   transmit  $\mu, \sigma^2, k, b$  to the cluster head
24: end event

```

---

The pseudo code of the local calibration procedure that runs at a sensor is shown in Algorithm 1. When the network is deployed and no target is present, the cluster head issues the **ESTIMATE\_NOISE** command and each member sensor estimates the noise model (Line 2-3). When the controlled target appears (e.g., the controlled vehicle is approaching), the cluster head issues the **TRAINING\_BEGIN** command and each member sensor starts a periodical timer with timeout of  $W$  seconds (Line 7). For instance, we can let  $W = 5$  s under the settings of the SensIT experiments [5]. We note that sensors are not required to be synchronized. Each sensor iteratively updates the local signal decay model for every  $W$  seconds (Line 11-18). When the controlled target disappears, the cluster head issues the **TRAINING\_END** command and each sensor reports its local model parameters.

## 6 Optimal System-level Model Calibration

In this section, we first derive the system detection performance of a calibrated network under the data fusion model described in Section 3.3, and then discuss how to find the optimal system-level calibration coefficients.

### 6.1 Calibrated System Detection Performance

When no target is present, the calibrated measurement of sensor  $i$  follows the normal distribution, i.e.,  $\hat{y}_i|H_0 = \gamma_i n_i \sim \mathcal{N}(\gamma_i \mu_i, \gamma_i^2 \sigma_i^2)$ . Therefore, the sum of calibrated measurements follows the normal distribution, i.e.,  $\hat{Y}|H_0 = \sum_{i=1}^N \hat{y}_i|H_0 \sim \mathcal{N}\left(\sum_{i=1}^N \gamma_i \mu_i, \sum_{i=1}^N \gamma_i^2 \sigma_i^2\right)$ . Hence, the system false alarm rate is given by  $P_F = \mathbb{P}(\hat{Y} \geq T|H_0) = Q\left(\frac{T - \sum_{i=1}^N \gamma_i \mu_i}{\sqrt{\sum_{i=1}^N \gamma_i^2 \sigma_i^2}}\right)$ , where  $T$  is the detection threshold of

the data fusion model. As  $P_D$  is a non-decreasing function of  $P_F$  [20], it is maximized when  $P_F$  is set to be the upper bound  $\alpha$  [20]. Let  $P_F = \alpha$ , the optimal detection threshold  $T^*$  is derived as

$$T^* = \sum_{i=1}^N \gamma_i \mu_i + Q^{-1}(\alpha) \cdot \sqrt{\sum_{i=1}^N \gamma_i^2 \sigma_i^2}. \quad (7)$$

When the target is present, the calibrated measurement of sensor  $i$  follows the normal distribution, i.e.,  $\hat{y}_i|H_1 = \hat{s}_i + \gamma_i n_i \sim \mathcal{N}(\hat{s}_i + \gamma_i \mu_i, \gamma_i^2 \sigma_i^2)$ . Therefore, the sum of calibrated measurements also follows the normal distribution, i.e.,  $\hat{Y}|H_1 = \sum_{i=1}^N \hat{y}_i|H_1 \sim \mathcal{N}\left(\sum_{i=1}^N \hat{s}_i + \sum_{i=1}^N \gamma_i \mu_i, \sum_{i=1}^N \gamma_i^2 \sigma_i^2\right)$ . Hence, the system detection probability is given by  $P_D = \mathbb{P}(\hat{Y} \geq T|H_1) = Q\left(\frac{T - \sum_{i=1}^N \hat{s}_i - \sum_{i=1}^N \gamma_i \mu_i}{\sqrt{\sum_{i=1}^N \gamma_i^2 \sigma_i^2}}\right)$ . By replacing  $T$  with the optimal detection threshold  $T^*$  given by (7) to bound the false alarm rate, we have

$$P_D = Q\left(Q^{-1}(\alpha) - \frac{\sum_{i=1}^N \gamma_i \hat{s}_i}{\sqrt{\sum_{i=1}^N \gamma_i^2 \sigma_i^2}}\right). \quad (8)$$

### 6.2 Optimal Calibration

From the derivation in Section 6.1, if the detection threshold at the cluster head is set to be  $T^*$  given by (7), the system false alarm rate is  $\alpha$  and the detection probability of the calibrated network is given by (8). Therefore, Problem 2 formulated in Section 4.2 reduces to finding the common signal decay model  $(\hat{S}, r, k)$  and calibration coefficients  $\{\gamma_i | 1 \leq i \leq N\}$  such that the detection probability given by (8) is maximized. We have the following lemma.

**Lemma 1.** *The system detection probability of the calibrated network is independent of the source energy of the controlled target  $S_0$ , the calibrated source energy  $\hat{S}$ , and the reference distance  $r$  in the common signal decay model.*

*Proof.* By replacing  $\gamma_i = \hat{s}_i/s_i, s_i$  with (1) and  $\hat{s}_i$  with (2), respectively, Equation (8) becomes

$$P_D = Q\left(Q^{-1}(\alpha) - S \cdot \frac{\sum_{i=1}^N l_i^{-k}}{\sqrt{\sum_{i=1}^N \sigma_i^2 r_i^{-2k_i} l_i^{2k_i - 2k}}}\right), \quad (9)$$

where  $S$  is the source energy of the target that is present at the surveillance spot and  $l_i$  is the distance between sensor  $i$  and the surveillance spot. Note that  $k_i$  and  $r_i$  are the sensing parameters of a sensor, which are independent of the controlled target. From (9),  $P_D$  is independent of  $S_0, \hat{S}$ , and  $r$ .  $\square$

From Lemma 1, the detection performance of the calibrated network with regard to a surveillance spot only depends on the common decay factor  $k$  and the target's source

---

**Algorithm 2** System-level calibration and detection
 

---

```

1: event  $\{\mu_i, \sigma_i^2, k_i, b_i | i \in [1, N]\}$  are received do
2:   for each surveillance spot  $j$  in all surveillance spots do
3:     numerically maximize  $\Lambda(k)$  given by (11)
4:     compute  $\gamma_{j,i}$  for each sensor  $i$  by (10)
5:     compute the optimal detection threshold  $T_j$  by (7) with  $\{\gamma_{j,i} | i \in [1, N]\}$ 
6:   end for each
7: end event
8:
9: event detection request for surveillance spot  $j$  is received do
10:  retrieve readings from member sensors,  $\{y_i | i \in [1, N]\}$ 
11:   $\hat{Y} \leftarrow \sum_{i=1}^N \gamma_{j,i} \cdot y_i$ 
12:  report ( $\hat{Y} \geq T_j ? H_1 : H_0$ )
13: end event
  
```

---

energy  $S$  at run time. The optimal solution to Problem 2 is given by the following theorem.

**Theorem 1.** *The optimal calibration coefficient of sensor  $i$ ,*

$$\gamma_i^* = \gamma \cdot l_i^{k_i - k^*} \cdot e^{-b_i}, \quad (10)$$

where  $\gamma$  is a constant for all sensors and the optimal common decay factor  $k^*$  maximizes the following function

$$\Lambda(k) = \frac{\sum_{i=1}^N l_i^{-k}}{\sqrt{\sum_{i=1}^N \sigma_i^2 e^{-2b_i} l_i^{2k_i - 2k}}}. \quad (11)$$

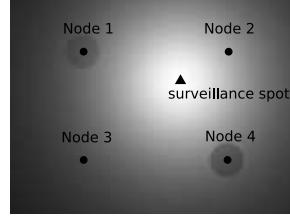
*Proof.* By replacing  $r_i$  in (9) with its estimate, i.e.,  $r_i = \left(\frac{e^{b_i}}{S_0}\right)^{\frac{1}{k_i}}$ , we have  $P_D = Q\left(Q^{-1}(\alpha) - \frac{S}{S_0} \cdot \Lambda(k)\right)$ . As  $Q(\cdot)$  is an decreasing function, the maximum of  $\Lambda(k)$  maximizes  $P_D$ . Therefore,  $k^*$  maximizes the system detection probability and the optimal calibration coefficient is given by

$$\gamma_i^* = \frac{\hat{s}_i}{s_i} = \frac{\hat{S} r_i^{k^*} l_i^{k_i}}{S r_i^{k_i} l_i^{k^*}} = \frac{\hat{S} S_0 r^{k^*}}{S} \cdot l_i^{k_i - k^*} e^{-b_i},$$

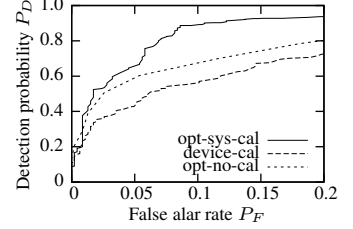
where  $r_i$  is replaced with  $r_i = \left(\frac{e^{b_i}}{S_0}\right)^{\frac{1}{k_i}}$ . Note that  $\frac{\hat{S} S_0 r^{k^*}}{S}$  is a common proportion for all sensors. According to (8), proportionally scaling all calibration coefficients has no impact on the system detection performance. Therefore, we can replace  $\frac{\hat{S} S_0 r^{k^*}}{S}$  with any constant  $\gamma$  and have (10).  $\square$

Maximizing  $\Lambda(k)$  in (11) is an unconstrained numerical optimization problem. There are various algorithms for solving such a problem such as the Newton's method. However, the naive search of  $k^*$  also suffices as  $k$  usually has a narrow value domain, e.g.,  $(0, 10]$ . Moreover, the computation overhead can be controlled by the search granularity.

The pseudo code of the system-level calibration procedure that runs at the cluster head is shown in Algorithm 2. Upon receiving local model parameters from all member sensors, the cluster head computes the optimal calibration coefficients and detection threshold for each surveillance spot (Line 2-6). When the network is required to detect whether a target is present at a surveillance spot, the cluster



**Figure 4.** A screenshot when target appears.



**Figure 5.** Receiver operating characteristic curves.

head first retrieves the readings from member sensors and then compares the sum of readings against the corresponding detection threshold to make a decision (Line 10-12).

## 7 Testbed Experiments

To evaluate the performance of our system-level calibration approach, we have conducted both experiments on a testbed of TelosB motes as well as extensive simulations. We first present the testbed experiments in this section and then the simulations in Section 8.

### 7.1 Experiment Methodology and Settings

In our experiments, four TelosB motes are attached against a LCD screen with resolution of  $1024 \times 768$  to detect a light spot displayed on the LCD. The light spot simulates the target and its display is controlled by a program. We simulate the signal decay by setting the grayscale of pixels around the light spot, as shown in Figure 4. Moreover, we simulate the sensor bias by reducing the grayscale of the pixels around the sensor by different percentages. Specifically, the grayscale of the pixels within a certain distance (60 pixels in our experiments) from sensor  $i$  is reduced by  $\delta_i\%$ . The experiment settings are listed in Table 1. We note that a similar experiment methodology is employed for studying the sensing performance of WSNs in [10, 19]. The sensors measure the light intensity every 250 milliseconds. The timeout of the periodical timer in Algorithm 1 is set to be 10 seconds. A sink node calibrates the four sensors in the training phase and performs data fusion at run time. In the training phase, the light spot appears at random positions and lasts for 10 seconds at each position. A surveillance spot is selected at  $(600, 300)$ . At run time, the sink fuses the readings received within every 250 milliseconds to make detections. The system false alarm rate and detection probability are estimated using the detection results when the light spot is absent or present at the surveillance spot, respectively.

We compare our optimal system-level calibration approach (referred to as *opt-sys-cal* in the following) with two baseline approaches, namely, *device-cal* and *opt-no-cal*. In

**Table 1. Settings and calibration coefficients**

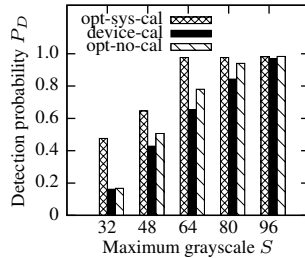
Node	Position	$\delta_i(\%)$	$l_i$	$\mu_i$	$\sigma_i^2$	$k_i$	$b_i$	$\gamma_i^*$	$\gamma_i'$
1	(256,192)	15	313	20.1	3.2	1.6	11.8	4.2	1.5
2	(768,192)	0	243	17.1	2.7	1.7	12.9	14.9	1.0
3	(256,576)	0	403	46.9	8.1	1.9	13.3	1.0	2.0
4	(768,576)	25	352	23.0	4.0	1.6	11.4	3.3	2.4

the *device-cal* approach, the impact of noise is mitigated by averaging the readings at local sensors and the sensor with the highest sensitivity (*i.e.*, the highest signal-distance curve) is manually chosen to be the standard sensor. We then compute the mapping functions for other sensors to the standard sensor. In the *opt-no-cal* approach, the readings of sensors are fused without calibration. Each sensor only needs to estimate its noise model to choose the *optimal* detection threshold, which is given by (7) with  $\gamma_i = 1$ . Note that *opt-no-cal* is an optimal detection approach without enforcing the identical sensing model for all sensors (*i.e.*, the second constraint of Problem 2). However, as discussed in Section 3.1, many existing data fusion algorithms cannot be readily applied in a network where sensors have different sensing models.

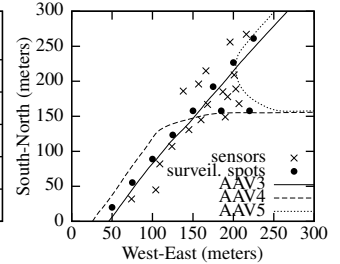
## 7.2 Experiment Results

Several results including sensors' noise and signal decay models as well as the calibration coefficients are reported in Table 1. Note that  $\gamma_i^*$  and  $\gamma_i'$  are the calibration coefficients yielded by the *opt-sys-cal* and *device-cal* approaches, respectively. We can draw two observations from the table. First, sensors have different noise profiles. In particular, the noise profile of the third sensor is significantly different from others. Second, in our approach, the sensors close to the surveillance spot are allocated with high calibration coefficients. According to (10), the calibration coefficients of our approach jointly accounts for sensor's sensing model and the distance from the surveillance spot. As a result, the high-quality measurements from the sensors that are close to the surveillance spot can contribute more to the detection results. In contrast, the *device-cal* approach only accounts for the mapping of measurements with respect to the standard sensor and may yield relatively low system performance.

We now evaluate the receiver operating characteristic (ROC) of the calibrated network, which is a widely adopted performance measure for detection systems. Figure 5 plots the ROC curves of various calibration approaches when the grayscale at the center of the light spot is 48. We can see that our optimal approach significantly outperforms the baseline approaches in terms of  $P_D$  for any given  $P_F$ . Figure 6 plots the detection probability achieved by various calibration approaches versus the grayscale at the center of the light spot when the false alarm rate is 5%. This result clearly demonstrates the effectiveness of our approach for the scenarios



**Figure 6.  $P_D$  vs. the grayscale of the light spot.**



**Figure 7. Sensor layout and trajectories of AAV3-5.**

with low signal-to-noise ratios.

## 8 Trace-driven Simulations

In addition to the testbed experiments in Section 7, we also conduct extensive simulations based on real data traces collected in the DARPA SensIT vehicle detection experiments using acoustic sensors [5].

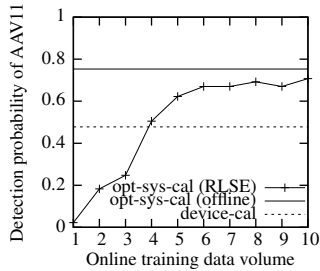
### 8.1 Settings and Methodology

We use the real data traces collected in the DARPA SensIT vehicle detection experiments [5], where 75 WINS NG 2.0 nodes are deployed to detect Amphibious Assault Vehicles (AAVs) driving through several intersecting roads. We refer to [5] for detailed setup of the experiments. The dataset used in our simulations includes the ground truth data and the acoustic time series recorded at a frequency of 4960 Hz. The ground truth data include the positions of sensors and the trajectory of the AAV recorded by a GPS device. We use the data traces recorded for 9 AAV runs (AAV3-11). We choose 10 surveillance spots along the roads. Figure 7 plots the layout of the sensors and surveillance spots as well as the trajectories of AAV3-5.

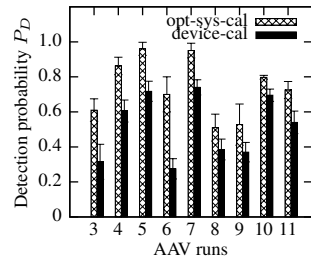
In our *opt-sys-cal* calibration approach, a sensor generates a data point  $(P_{Di}, d_i)$  using the time series during 0.75 seconds. To evaluate the detection performance of the calibrated network, we measure the detection probability at each surveillance spot as follows. When an AAV enters the  $300 \times 300 \text{ m}^2$  region shown in Figure 7, for each surveillance spot, the network detects the vehicle by fusing sensors' measurements for every 0.75 seconds and the detection probability is computed as the fraction of successful detections. In this section, we report the average detection probability over all surveillance spots.

### 8.2 Simulation Results

We first evaluate the impact of the online model estimation algorithm (*i.e.*, RLSE) in Section 5.2. We use the data traces of AAV3 and AAV11 for training and testing, respectively. Figure 8 plots the detection probability versus the number of data points used by the RLSE algorithm



**Figure 8. Convergence of  $P_D$ .**



**Figure 9.  $P_D$  of different AAVs.**

at each sensor. Note that we use all training data for the *device-cal* approach and our *opt-sys-cal* approach with the offline model estimation algorithm in Section 5.1. The results shown in Figure 8 are the average of 10 runs. We can see that our *opt-sys-cal* approach with the RLSE algorithm converges to the offline results with small error in ten steps. Moreover, our *opt-sys-cal* approach outperforms the *device-cal* approach.

We then use the data traces of various AAV runs to evaluate the effectiveness of our approach. We train the network using AAV3 and measure the detection probability of various runs. Figure 9 plots the error bar of detection probability. We can see that our *opt-sys-cal* approach outperforms the *device-cal* approach in each run. The measured false alarm rates of the *opt-sys-cal* and *device-cal* approaches are 5.2% and 5.1% (with standard deviations of less than 2%), respectively.

## 9 Conclusion

Data fusion is an effective technique for improving system sensing performance by enabling efficient collaboration among sensors. In this paper, we propose a two-tier system-level calibration approach for fusion-based WSNs. Our approach introduces low computation and communication overhead and therefore is suitable for low-cost wireless sensors and is more scalable. The testbed experiments and extensive simulations based on real data traces show that our approach can significantly boost the system detection performance of WSNs.

## Acknowledgment

This work is supported, in part, by the National Science Foundation under a CAREER grant CNS-0954039, Canadian NSERC Discovery Grant 341823-07, and FQRNT grant 2010-NC-131844.

## References

- [1] K. Astrom and B. Wittenmark. *Adaptive Control*. Addison-Wesley, 1994.
- [2] V. Bychkovskiy, S. Megerian, D. Estrin, and M. Potkonjak. A collaborative approach to in-place sensor calibration. In *IPSN*, 2003.

- [3] T. Clouqueur, K. K. Saluja, and P. Ramanathan. Fault tolerance in collaborative sensor networks for target detection. *IEEE Trans. Comput.*, 53(3), 2004.
- [4] Crossbow Technology, Inc. MICA and MICA2 wireless measurement system datasheets.
- [5] M. Duarte and Y.-H. Hu. Vehicle classification in distributed sensor networks. *J. Parallel and Distributed Computing*, 64(7), 2004.
- [6] J. Feng, S. Megerian, and M. Potkonjak. Model-based calibration for sensor networks. In *IEEE Sensors*, 2003.
- [7] L. Gu, D. Jia, P. Vicaire, T. Yan, L. Luo, A. Tirumala, Q. Cao, T. He, J. A. Stankovic, T. Abdelzaher, and B. H. Krogh. Lightweight detection and classification for wireless sensor networks in realistic environments. In *SenSys*, 2005.
- [8] M. Hata. Empirical formula for propagation loss in land mobile radio services. *IEEE Trans. Veh. Technol.*, 29, 1980.
- [9] T. He, S. Krishnamurthy, J. A. Stankovic, T. Abdelzaher, L. Luo, R. Stoleru, T. Yan, L. Gu, J. Hui, and B. Krogh. Energy-efficient surveillance system using wireless sensor networks. In *MobiSys*, 2004.
- [10] J. Hwang, T. He, and Y. Kim. Exploring in-situ sensing irregularity in wireless sensor networks. In *SenSys*, 2007.
- [11] A. T. Ihler, J. W. Fisher, R. L. Moses, and A. S. Willsky. Nonparametric belief propagation for self-calibration in sensor networks. In *IPSN*, 2004.
- [12] D. Li and Y.-H. Hu. Energy based collaborative source localization using acoustic micro-sensor array. *EUROSIP Journal on Applied Signal Processing*, (4), 2003.
- [13] E. Miluzzo, N. Lane, A. Campbell, and R. Olfati-Saber. CaliBree: A Self-calibration System for Mobile Sensor Networks. In *DCOSS*, 2008.
- [14] R. Moses and R. Patterson. Self-calibration of sensor networks. In *SPIE*, volume 4743, 2002.
- [15] N. Ramanathan, L. Balzano, M. Burt, D. Estrin, T. Harmon, C. Harvey, J. Jay, E. Kohler, S. Rothenberg, and M. Srivastava. Rapid deployment with confidence: Calibration and fault detection in environmental sensor networks. Technical report, Center for Embedded Networked Sensing, 2006.
- [16] X. Sheng and Y.-H. Hu. Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks. *IEEE Trans. Signal Process.*, 53(1), 2005.
- [17] R. Tan, G. Xing, J. Chen, W.-Z. Song, and R. Huang. Quality-driven volcanic earthquake detection using wireless sensor networks. In *RTSS*, 2010.
- [18] R. Tan, G. Xing, B. Liu, and J. Wang. Impact of data fusion on real-time detection in sensor networks. In *RTSS*, 2009.
- [19] R. Tan, G. Xing, X. Liu, J. Yao, and Z. Yuan. Adaptive calibration for fusion-based wireless sensor networks. In *INFOCOM*, 2010.
- [20] P. K. Varshney. *Distributed Detection and Data Fusion*. Springer, 1996.
- [21] K. Whitehouse and D. Culler. Calibration as parameter estimation in sensor networks. In *WSNA*, 2002.
- [22] G. Xing, R. Tan, B. Liu, J. Wang, X. Jia, and C.-W. Yi. Data fusion improves the coverage of wireless sensor networks. In *MobiCom*, 2009.
- [23] Y. Zhuang, L. Chen, X. Wang, and J. Lian. A weighted moving average-based approach for cleaning sensor data. In *ICDCS*, 2007.