System-level Calibration for Data Fusion in Wireless Sensor Networks

RUI TAN, Michigan State University, USA
GUOLIANG XING, Michigan State University, USA
ZHAOHUI YUAN, Huadong Jiao Tong University, P.R. China
XUE LIU, McGill University, Canada
JIANGUO YAO, Shanghai Jiao Tong University, P.R. China

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 compared with the device-level approach. 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 motes and extensive simulations based on synthetic data sets as well as 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.

Categories and Subject Descriptors: B.8.1 [**Performance and Reliability**]: Reliability, Testing, and Fault-Tolerance; C.3 [**Special-purpose and Application-based Systems**]: Signal processing systems

General Terms: Algorithms, Performance

Additional Key Words and Phrases: Sensor calibration, parameter estimation, data fusion, target detection, wireless sensor network

1. INTRODUCTION

Wireless sensor networks (WSNs) are increasingly deployed for mission-critical applications such as target detection [Clouqueur et al. 2004; Li et al. 2002], localization [Li

A preliminary version of this submission was published in the following conference paper:
Rui Tan, Guoliang Xing, Zhaohui Yuan, Xue Liu, Jianguo Yao. System-level Calibration For Fusion-based Wireless Sensor Networks. In *The 31st IEEE Real-Time Systems Symposium (RTSS 2010)*, Nov 30 – Dec 3, San Diego, CA, USA.
This work is supported, in part, by the National Science Foundation under grants CNS-0954039 (CAREER), OIA-1125163, ECCS 1029683, Canadian NSERC Discovery Grant 341823-07, and FQRNT grant 2010-NC-131844.
Author's addresses: R. Tan and G. Xing, Department of Computer Science and Engineering, Michigan State University, East Lansing, MI 48824, USA; Z. Yuan, School of Software, Huadong Jiao Tong University, Nanchang, Jiangxi 310013, P.R. China; X. Liu, School of Computer Science, McGill University, Montreal, Quebec, Canada; J. Yao, School of Software, Shanghai Jiao Tong University, Shanghai 200240, P.R. China.

DOI 10.1145/0000000.0000000 http://doi.acm.org/10.1145/0000000.0000000

and Hu 2003; Sheng and Hu 2005], and security surveillance [He et al. 2004; Gu et al. 2005]. 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 [Ramanathan et al. 2006; Bychkovskiy et al. 2003], complex deployment terrain, and random noise [Bychkovskiy et al. 2003] must be dealt with properly before sensor data can be meaningfully interpreted by users. A common solution 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 [Ramanathan et al. 2006] 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 [Varshney 1996] 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 [He et al. 2004], localization [Li and Hu 2003], and classification [Duarte and Hu 2004]. 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. Our approach has the following two key advantages. First, most processing is performed locally by individual sensor, resulting in low communication overhead. Second, the calibration phase only needs the position of a controlled target as the input to the calibration process. At run time, the calibrated network can achieve the optimal performance for detecting target at any given position. 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 synthetic data sets as well as real data traces collected in a real vehicle detection experiment [Duarte and Hu 2004]. 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 discusses several issues that are not addressed in this paper. Section 10 concludes the paper.

2. RELATED WORK

Sensor calibration is a fundamental problem in WSNs. Ni et al. [Ni et al. 2009] investigate the reasons and categories of sensor biases and faults based on the data collected in a number of WSN deployments. Previous literature on calibration [Balzano and Nowak 2007; Bychkovskiy et al. 2003; Feng et al. 2003; Ihler et al. 2004; Miluzzo et al. 2008; Moses and Patterson 2002; Ramanathan et al. 2006; Whitehouse and Culler 2002] can be categorized into micro-/macro- and non-blind/blind/semi-blind approaches.

The *micro*-calibration refers to the method that tunes each individual sensor to output accurate readings. In [Ramanathan et al. 2006], each chemical sensor is carefully calibrated in laboratory to obtain the mapping from its reading to the true value. In [Miluzzo et al. 2008], an uncalibrated sensor calibrates itself when encountering a calibrated sensor. In [Ihler et al. 2004], each sensor's location is calibrated using the position information of a subset of nodes. In contrast, the *macro*-calibration [Feng et al. 2003; Whitehouse and Culler 2002; Bychkovskiy et al. 2003] focuses on optimizing the overall system performance. In [Feng et al. 2003], the biases of light sensors are estimated by solving a group of equations that integrate all sensors' measurements. Similarly, in [Whitehouse and Culler 2002], the parameters of ranging sensors are estimated by regression based on pair-wise range measurements. In [Bychkovskiy et al. 2003], 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. Different from the above works, in our approach, each sensor estimates a local sensing model and only transmits the model parameters for system-level calibration.

Existing calibration methods can also be classified into non-blind, blind, and semi-blind approaches according to the amount of required ground truth information. In the *non-blind* calibration approaches [Ramanathan et al. 2006; Feng et al. 2003; Whitehouse and Culler 2002], the parameters of sensors are adjusted using sensor measurements and known ground truth inputs. For instance, the readings of extra high-quality

sensors are employed as the ground truth in [Ramanathan et al. 2006; Feng et al. 2003]. In [Whitehouse and Culler 2002], the distance between ranging sensors are known during the calibration phase. In contrast, the *blind* calibration approaches [Bychkovskiy et al. 2003; Balzano and Nowak 2007] do not require ground truth inputs. In [Bychkovskiy et al. 2003], the discrepancies among co-located sensors are eliminated by exploiting the spatial correlation in their measurements, hence the ground truth is unnecessary. It is theoretically proved in [Balzano and Nowak 2007] that sensors can be partially calibrated using blind measurements. Blind approaches often require dense deployment [Bychkovskiy et al. 2003] or known model of the signal [Balzano and Nowak 2007]. As a compromise, the *semi-blind* calibration approaches [Ihler et al. 2004; Moses and Patterson 2002; Miluzzo et al. 2008] only require partial ground truth information. In [Ihler et al. 2004; Moses and Patterson 2002], the sensor locations are calibrated using position information of a subset of nodes. In [Miluzzo et al. 2008], an uncalibrated sensor calibrates itself when rendezvousing a calibrated sensor. The calibration approach presented in this paper falls into semi-blind category, in which sensors can calibrate their signal measurements based on partial ground truth information.

Data fusion [Varshney 1996] 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 [Duarte and Hu 2004; Gu et al. 2005; He et al. 2004; Li and Hu 2003]. In our earlier work [Tan et al. 2010], we calibrate the parameters of data fusion such that a sensor network adapts to various dynamics. Similar to this paper, [Fabeck and Mathar 2007] aims to calibrate a fusion-based WSN for target detection. Specifically, the detection performance of each sensor is calibrated using the ground truth information labeled by an external observer. The optimal decision fusion rule at the fusion center is then constructed based on the calibrated detection performance of sensors. In their approach, the monitored target in the calibration phase and run time must be the same. In contrast, we aim to calibrate the sensor measurements such that the sensors have an identical sensing model that optimizes the system detection performance of the data fusion. In our approach, the target used in the calibration phase can be different from the target at run time. This feature makes our approach more practical. Moreover, the calibrated network can adapt to the targets with different profiles at run time.

The model of signal and the spatiotemporal correlation among sensor readings have been exploited to reduce the communication overhead in collecting periodical sensor readings, e.g., in [Santini and Romer 2006] and the references therein. Our approach also leverages the model of the signal to reduce the communication overhead in calibrating sensors. The data reduction approaches, e.g., [Santini and Romer 2006], can be integrated with our approach to reduce communication overhead of a calibrated network at run time.

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 error*s (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 de-

ployment. For instance, in the scenario of vehicle detection [Duarte and Hu 2004; He et al. 2004; Gu et al. 2005], 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 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., Gaussian). 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 [Duarte and Hu 2004] 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(a) plots the acoustic energy measurements taken by node 41 and 48, versus the distance from a driving vehicle. We can see from Figure 1(a) 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) [Ramanathan et al. 2006; Feng et al. 2003; Whitehouse and Culler 2002]. However, such an approach often incurs high communication and computation 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 [Clouqueur et al. 2004; Sheng and Hu 2005; Li and Hu 2003] 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, seismic and electromagnetic signals) 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 is given by

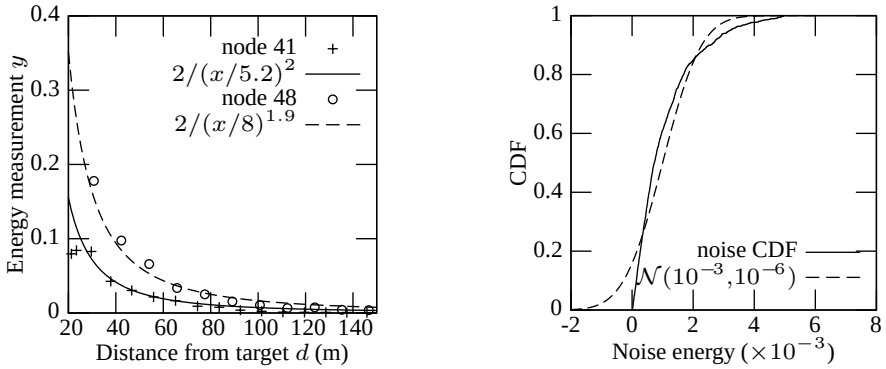
$$s_i = S \cdot w(d_i|\Theta_i), \tag{1}$$

where $w(d_i|\Theta_i)$ is a decreasing function of d_i and Θ_i is a set of parameters of the function $w(\cdot)$. In this paper, the function $w(\cdot)$ is referred to as *signal decay function* and Θ_i is referred to as *signal decay parameters*. We now discuss two instances of the signal decay function, i.e., the *power law decay* and the *exponential decay*.

— The propagation of many mechanical waves (e.g., acoustic and seismic signals) follows the power law decay, which can be expressed by

$$s_i = S \cdot \frac{1}{(d_i/r_i)^{k_i}}, \tag{2}$$

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 [Hata 1980]. In particular, in ideal open space,



(a) (b)

Fig. 1. (a) Normalized energy measurement vs. the distance from the vehicle. (b) The CDF of noise energy and the CDF of $\mathcal{N}(0.001, 0.001^2)$.

inverse-square law (i.e., $k_i = 2$) applies to various 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. For the power law decay, $\Theta_i = \{r_i, k_i\}$ and $w(d_i|\Theta_i) = \frac{1}{(d_i/r_i)^{k_i}}$.

— According to Beer-Lambert law [Fuwa and Valle 1963], the intensity of light attenuates with the travel distance in media and follows the exponential decay. Specifically,

$$s_i = S \cdot \mathrm{e}^{-\lambda_i \cdot d_i}, \tag{3}$$

where λ_i is referred to as Lambert absorption coefficient. For instance, the Lambert absorption coefficient has a wide range from $50\,\mathrm{cm}^{-1}$ to $10000\,\mathrm{cm}^{-1}$ for the infrared light traveling in water [Robertson and Williams 1971]. Aquatic fluorometers can have different Lambert absorption coefficients because of the irregular solution density distribution in water, for instance, caused by algae. For the exponential decay, $\Theta_i = \{\lambda_i\}$ and $w(d_i|\Theta_i) = \mathrm{e}^{-\lambda_i \cdot d_i}$.

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 (≥ 30) samples [Duarte and Hu 2004; Sheng and Hu 2005]. From the central limit theorem, the noise energy n_i follows the normal distribution [Sheng and Hu 2005], formally, $n_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$, where μ_i and σ_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 [Clouqueur et al. 2004; Varshney 1996; Sheng and Hu 2005] and also have been empirically verified [Li and Hu 2003; Hata 1980]. In Figure 1(a), the curves are the fittings of the data points to the power law decay model in (2). 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 $5.2\,\mathrm{m}$ and $8\,\mathrm{m}$, respectively. Figure 1(b) plots the cumulative distribution function (CDF) of background noise measured by a sensor in the SensIT experiments [Duarte and Hu 2004]. We can see that the CDF of measured noise well matches the CDF of the normal distribution $\mathcal{N}(0.001, 0.001^2)$.

Table I summarizes the notations used in this paper.

Table I. Summary of Notations

Symbol	Definition
S_0	energy emitted by the controlled target in training
S	energy emitted by the target at run time
μ_i, σ_i^2	mean and variance of noise
Θ_i	signal decay parameters
r_i, k_i	reference distance and decay factor
λ_i	Lambert absorption coefficient
d_i	distance from the controlled target
$w(d_i\|\Theta_i)$	signal decay function
l_i	distance from the surveillance spot
s_i	attenuated signal energy
n_i	Gaussian noise, $n_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$
H_0 / H_1	hypothesis that the target is absent / present
y_i	energy measurement, $y_i\|H_0 = n_i$, $y_i\|H_1 = s_i + n_i$
b_i	z-intercept of linear fitting of local measurements
N	the number of sensors in a detection cluster
α	upper bound of system false alarm rate

Note: The symbols with subscript i refer to the notation of sensor i.

3.3. Multi-sensor Data Fusion Model

Data fusion [Varshney 1996] 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 [Varshney 1996] 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 cluster head decides H_1; otherwise, it decides H_0. Such a basic data fusion model has been widely employed in previous work [Clouqueur et al. 2004; Tan et al. 2010].

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 using acoustic sensors to illustrate the basic idea of our approach. Note that in this example, the acoustic signal follows the power law decay in (2). In order to calibrate the sensors that are deployed to monitor vehicles as shown in Figure 2, 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

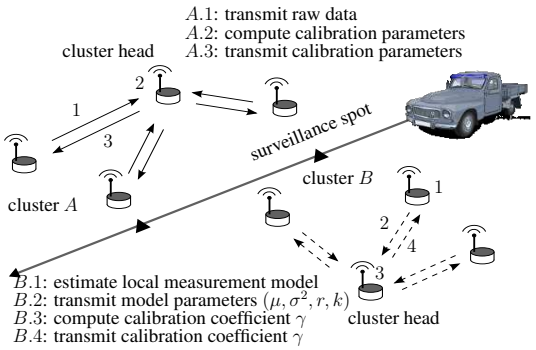


Fig. 2. Two system-level calibration schemes. Cluster A and B run centralized and two-tier calibration algorithms, respectively.

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 2, 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 (2). 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 2, 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. In this paper, 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 γ_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. Other advanced calibration schemes, e.g., using a non-linear mapping function, can outperform the linear calibration scheme. However, due to the

simplicity of the linear calibration scheme, it only imposes small overhead to the network and hence it is suitable for the WSNs composed of resource-constrained sensors.

In this paper, we are only concerned with the detection performance at a number of fixed geographical locations which are referred to as *surveillance spot*s. Surveillance spots can be chosen before network deployment according to application requirements or identified by the network autonomously after deployment. For instance, in Figure 2, the surveillance spots can be chosen along the roads in the monitored area. When we have little prior information about the spatial distribution of the target, we can choose a number of surveillance spots uniformly over the monitored region. For each surveillance spot, the cluster head computes a calibration coefficient γ_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 obtained by existing target localization algorithms [Li and Hu 2003; Sheng and Hu 2005].

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 parameters of the Gaussian noise model and the signal decay model, i.e., μ_i, σ_i^2 and Θ_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 parameters 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 $\mathcal{Y}_i(t) = \{y_i[1], y_i[2], \ldots, y_i[M]\}$. How to compute the parameters of the Gaussian noise and signal decay models (μ_i, σ_i^2, Θ_i) of sensor i using its noisy measurements $\{d_i(t), \mathcal{Y}_i(t) | t = 1, 2, \ldots\}$?*

Problem 1 is an over-provision parameter estimation problem [Astrom and Wittenmark 1994], 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

$$\hat{s}_i = \hat{S} \cdot w(l_i | \Theta), \tag{4}$$

where $\hat{S}$ is the calibrated source energy, Θ is the set of *common* signal decay parameters for all calibrated sensors. As the common signal decay model use the same signal decay function, it preserves the physical law governing the signal propagation. As a result, the detection algorithms designed based on the signal decay model can be applied in the calibrated network. We now discuss two instances of the common model. For the power law decay,

$$\hat{s}_i = \hat{S} \cdot \frac{1}{(l_i/r)^k}, \quad \Theta = \{r, k\}, \tag{5}$$

where r and k are the common reference distance and decay factor for all calibrated sensors, respectively. For the exponential decay, $\hat{s_i} = \hat{S} \cdot \mathrm{e}^{-\lambda \cdot d_i}$ and $\Theta = \{\lambda\}$, where λ is the common Lambert absorption coefficient for all calibrated sensors.

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 [Varshney 1996]. 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 the 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 $\{\mu_i, \sigma_i^2, \Theta_i | 1 \leq i \leq N\}$ and the distances from the surveillance spot $\{l_i | 1 \leq i \leq N\}$, to find a common signal decay model $(\hat{S}, \Theta)$, such that the detection probability P_D is maximized subject to that 1) the false alarm rate P_F is upper-bounded by α where $\alpha \in (0, 1)$, and 2) the calibrated signal energy of each sensor follows the common model given in (4).*

If the optimal solution to Problem 2 (i.e., $(\hat{S}, \Theta)$) is found, we can calculate the calibration coefficients. 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 [Clouqueur et al. 2004; Sheng and Hu 2005; Li and Hu 2003] 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,

$$\mu_i = \frac{1}{M} \sum_{j=1}^{M} y_i[j] | H_0, \quad \sigma_i^2 = \frac{1}{M-1} \sum_{j=1}^{M} (y_i[j] | H_0 - \mu_i)^2, \tag{6}$$

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

Now we discuss how to estimate the signal decay parameters Θ_i 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 [Zhuang et al. 2007]. 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 η. 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 η to M. The setting of the threshold η will be discussed in Section 5.3. Hence, sensor i has a statistic $P_{Di}(t)$ for each time step t. Sensor i then estimates the signal decay parameters Θ_i by least squares techniques using the statistics $\{P_{Di}(t), d_i(t)|t = 1, 2, \ldots\}$ 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 [Duarte and Hu 2004], 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 [Duarte and Hu 2004]. According to the weak law of large numbers, to achieve a precision of ϵ with a probability of at least p in estimating P_{Di}, M should be larger than $\frac{1}{4(1-p)\epsilon^2}$. We note that the sensor's sampling rate is often high (e.g., 4960 Hz for acoustic sensors in [Duarte and Hu 2004]) and hence M can be large enough to achieve a high precision in estimating P_{Di}.

We now derive the relationship between the local detection probability P_{Di} and the signal decay parameters. 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 η is the detection threshold and $Q(\cdot)$ is the complementary CDF of the standard normal distribution, formally, $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty \mathrm{e}^{-t^2/2}\mathrm{d}t$. The relationship between P_{Di} and Θ_i depends on the specific form of the signal decay function. We now derive the relationships under the power law decay and the exponential decay, respectively. The basic idea of our approach is to linearize the relationship and then apply least squares fitting to estimate Θ_i.

If the signal follows the power law decay, by replacing s_i with (2) and taking logarithmic transformation, we have

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

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 $z_i(t) = \ln\left(\eta - \mu_i - \sigma_i Q^{-1}(P_{Di}(t))\right)$, $x_i(t) = -\ln d_i(t)$ and $b_i = \ln\left(S_0 r_i^{k_i}\right)$, we have

$$z_i(t) = k_i \cdot x_i(t) + b_i. \tag{7}$$

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

given by

$$k_i = \frac{\mathrm{cov}(x_i, z_i)}{\sigma_{x_i}^2}, \quad b_i = \bar{z}_i - k_i \cdot \bar{x}_i, \tag{8}$$

where $\bar{z}_i$ and $\bar{x}_i$ are the sample means, $\mathrm{cov}(x_i, z_i)$ and $\sigma_{x_i}^2$ are the covariance and variance (with respect to x_i) of the data points, respectively. As $b_i = \ln\left(S_0 r_i^{k_i}\right)$, the reference distance r_i can be computed by the estimates of k_i and b_i, i.e., $r_i = \left(\frac{\mathrm{e}^{b_i}}{S_0}\right)^{\frac{1}{k_i}}$. 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.

If the signal follows the exponential decay, we have

$$z_i(t) = -\lambda_i \cdot d_i(t) + \ln S_0, \tag{9}$$

where $z_i(t)$ has the same definition as for the power law decay. Hence, we can estimate λ_i by the least squares fit based on the transformed data points $\{z_i(t), d_i(t) | t = 1, 2, \ldots\}$. The measurement model of sensor i can be represented by a 3-tuple $(\mu_i, \sigma_i^2, \lambda_i)$.

5.2. Online Model Estimation

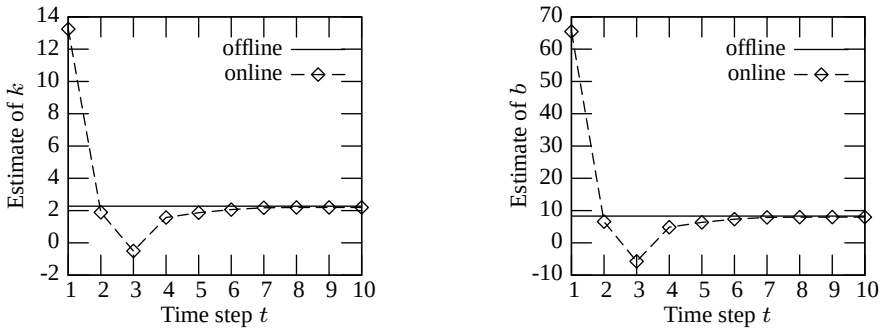
In Section 5.1, we adopted the least squares technique to estimate the slope k_i and z-intercept b_i in (7), and λ_i in (9), where sensor i has to store all previous data points to compute the estimates. Such an approach is not suitable for memory-constrained sensors. For instance, if a sensor generates a data pair (P_{Di}, d_i) every 0.5 seconds and the duration of the training process is 5 minutes, the sensor needs to store 600 data pairs, which is significant overhead for low-cost sensors with only a few Kbytes of memory [Crossbow]. For the power law decay and the exponential decay, the relationship between P_{Di} and Θ_i can be linearized by the data transformation approach presented in Section 5.1. In the linear fitting, the second-order statistics (i.e., variance and covariance) can be calculated incrementally without storing previous data. However, for other decay models, the relationship might not be linearized. Therefore, in this section, we present a general online improvement of the least squares approach for estimating Θ_i. The online algorithm works in a real-time fashion, i.e., the estimates of Θ_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 [Astrom and Wittenmark 1994]. For the power law decay, we reformulate (7) with vectors as $z_i(t) = \phi_i^T(t) \cdot \theta_i$ where $\phi_i(t) = [x_i(t), 1]^T$ and $\theta_i = [k_i, b_i]^T$. For the exponential decay, $\phi_i(t) = [-d_i(t), 1]^T$ and $\theta_i = [\lambda_i, \ln S_0]^T$. The recursive estimator of θ_i is given by

$$\begin{aligned} \theta_i(t) &= \theta_i(t-1) + L(t)\left(z_i(t) - \phi_i^T(t)\theta_i(t-1)\right), \\ L(t) &= P(t-1)\phi(t)\left(1 + \phi^T(t)P(t-1)\phi(t)\right)^{-1}, \\ P(t) &= (I - L(t)\phi^T(t))P(t-1), \end{aligned} \tag{10}$$

where I is the identity matrix. Equation (10) updates the estimates at each time step based on the error between the model output and the predicted output.

The convergence of RLSE depends on the initialization of $\theta_i(t)$ and $P(t)$. If we have the prior knowledge about θ_i, its initial estimate $\theta_i(0)$ can be set accordingly so that



(a) Estimate of k (b) Estimate of b

Fig. 3. The convergence of online local calibration algorithm. $k = 2.3$, $r = 2.5\,\text{m}$, $S_0 = 500$, $\mu = 0$, $\sigma^2 = 2$, $\theta(0) = [0, 0]^T$, $P(0) = 10^5 \cdot I$. The controlled target moves 2 m toward the sensor at each time step. A total of 10 data points are used for calculating the offline results.

the estimator can quickly converge. For instance, for the power law decay, $\theta_i(0)$ can be set to be the best guesses of k_i and b_i. Otherwise, the typical initialization is $\theta_i(0) = 0$. In RLSE, $P(t)$ is defined as $P(t) = \left(\sum_{j=1}^{t} \phi_i(j)\phi_i^T(j)\right)^{-1}$ [Astrom and Wittenmark 1994]. It has been proved in [Astrom and Wittenmark 1994] that $P(t)$ converges to $\left(\sum_{j=1}^{t} \phi_i(j)\phi_i^T(j)\right)^{-1}$ if $P(0) = \zeta I$ where ζ is sufficiently large. More detailed derivation and convergence proof of RLSE can be found in [Astrom and Wittenmark 1994]. 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., $\theta_{2\times1}$, $L_{2\times1}$, and $P_{2\times2}$ need to be stored. Such computation and memory overhead is affordable for low-cost wireless sensors such as MICA2 motes [Crossbow]. Figure 3 plots the numerical results that demonstrate the convergence of RLSE for the case of power law decay. From the figure, we can see that the results of RLSE converge to the offline results in ten time steps. Therefore, RLSE is highly accurate in local calibration, as a large number of ($\gg 10$) training data points are usually available for each sensor in practice [Duarte and Hu 2004].

5.3. Local Calibration Algorithm

The pseudo code of the local calibration procedure for the case of power law decay is shown in Algorithm 1. The algorithm runs at a sensor locally. 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 $W = 5\,\text{s}$ under the settings of the SensIT experiments [Duarte and Hu 2004]. 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. The algorithm for the case of exponential decay can be obtained by slightly changing Algorithm 1.

We now discuss several considerations on the implementation of Algorithm 1. We first discuss the setting of the threshold η at any time step t. If η is set improperly, the local detection probability $P_{Di}(t)$ will saturate. That is, if a too large or small η

Algorithm 1 Online local calibration of a sensor for the case of power law decay

event command `ESTIMATE_NOISE` is received **do**
  sample M measurements, $\{y[1], y[2], \ldots, y[M]\}$
  compute μ and σ^2 using (6)
end event

event command `TRAINING_BEGIN` is received **do**
  start a periodical timer `cali_timer` with timeout of W seconds
end event

event `cali_timer` is fired **do**
  query the current position of the controlled target
  $x \leftarrow -\ln(d)$ where d is the distance from the controlled target
  sample M measurements, $\{y[1], y[2], \ldots, y[M]\}$
  $\eta \leftarrow \sum_{j=1}^{M} y[j]/M$
  /* compute the fraction of measurements that exceed η */
  $P_D \leftarrow \#(y[j] \geq \eta, j \in [1, M])/M$
  $z \leftarrow \ln(\eta - \mu - \sigma Q^{-1}(P_D))$
  update k and b using (10) with x and z
end event

event command `TRAINING_END` is received **do**
  stop `cali_timer`
  transmit μ, σ^2, k, b to the cluster head
end event

is set, $P_{Di}(t)$ will be 0 or 1. In our approach, η is set to be the mean of $\mathcal{Y}_i(t)$ to avoid the saturation of $P_{Di}(t)$. Under this setting, $P_{Di}(t)$ will be around 0.5. We then discuss two approaches to implementing $Q^{-1}(\cdot)$ on resource-constrained sensors. First, there exist approximate formulae for calculating $Q^{-1}(\cdot)$ efficiently with known precision, e.g., its Maclaurin series. Second, $Q^{-1}(\cdot)$ can be implemented as a pre-computed look-up table. Suppose we pre-compute $Q^{-1}(P_D)$ where P_D takes 100 points evenly within $(0, 1)$ and each value is represented by 4 bytes. The size of the look-up table will be only 0.4 KBytes, which is affordable for low-cost wireless sensors such as MICA2 motes [Crossbow]. Moreover, due to the setting of η, only the values of $Q^{-1}(P_D)$ over a small range around 0.5 will be used. Therefore, the size of the look-up table can be further reduced.

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 [Varshney 1996], it is maximized when P_F is set to be the upper bound α [Varshney 1996]. Let $P_F = \alpha$, the optimal detection threshold T^* is derived as

$$T^* = \sum_{i=1}^{N}\gamma_i\mu_i + Q^{-1}(\alpha)\cdot\sqrt{\sum_{i=1}^{N}\gamma_i^2\sigma_i^2}. \tag{11}$$

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 (11) to bound the false alarm rate, we have

$$P_D = Q\left(Q^{-1}(\alpha) - \frac{\sum_{i=1}^{N}\gamma_i s_i}{\sqrt{\sum_{i=1}^{N}\gamma_i^2\sigma_i^2}}\right). \tag{12}$$

6.2. Optimal System-level Calibration

From the derivation in Section 6.1, if the detection threshold at the cluster head is set to be T^* given by (11), the system false alarm rate is α and the detection probability of the calibrated network is given by (12). Therefore, Problem 2 formulated in Section 4.2 reduces to finding the common signal decay model $(\hat{S}, \Theta)$ and calibration coefficients $\{\gamma_i|1 \leq i \leq N\}$ such that the detection probability given by (12) is maximized. We have the following lemma.

LEMMA 1. *If the target signal follows the signal decay model in (1), the system detection probability of the calibrated network is independent of the source energy of the controlled target S_0 and the calibrated source energy $\hat{S}$.*

PROOF. By replacing $\gamma_i = \hat{s}_i/s_i$, s_i with (1) and $\hat{s}_i$ with (4), respectively, Equation (12) becomes

$$P_D = Q\left(Q^{-1}(\alpha) - S\cdot\frac{\sum_{i=1}^{N}w(l_i|\Theta)}{\sqrt{\sum_{i=1}^{N}\sigma_i^2\frac{w^2(l_i|\Theta)}{w^2(l_i|\Theta_i)}}}\right). \tag{13}$$

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 the sensing parameters Θ_i are independent of the controlled target. Therefore, from (13), P_D is independent of S_0 and $\hat{S}$. □

If the signal follows the power law decay, Lemma 1 and the following lemma hold.

LEMMA 2. *If the target signal follows the power law decay in (2), the system detection probability of the calibrated network is independent of the reference distance r in the common signal decay model.*

PROOF. By replacing $\gamma_i = \hat{s}_i/s_i$, s_i with (2) and $\hat{s}_i$ with (5), respectively, Equation (12) becomes

$$P_D = Q\left(Q^{-1}(\alpha) - S \cdot \frac{\sum_{i=1}^{N} l_i^{-k}}{\sqrt{\sum_{i=1}^{N} \sigma_i^2 r_i^{-2k_i} l_i^{2k_i-2k}}}\right), \tag{14}$$

From (14), P_D is independent of r. □

From Lemma 1, the optimal solution to Problem 2 is given by the following theorem.

THEOREM 1. *Given the signal decay parameters of all sensors, i.e., $\{\Theta_i|\forall i\}$, the optimal calibration coefficient of sensor i is given by*

$$\gamma_i^* = \gamma \cdot \frac{w(l_i|\Theta^*)}{w(l_i|\Theta_i)}, \tag{15}$$

where γ is a constant for all sensors and the optimal common parameters Θ^* maximizes the following function

$$\Lambda(\Theta) = \frac{\sum_{i=1}^{N} w(l_i|\Theta)}{\sqrt{\sum_{i=1}^{N} \sigma_i^2 \frac{w^2(l_i|\Theta)}{w^2(l_i|\Theta_i)}}}. \tag{16}$$

PROOF. From (13), the maximum point of (16) maximizes P_D. Accordingly, the optimal calibration coefficient is given by $\gamma_i^* = \frac{\hat{s}_i}{s_i} = \frac{\hat{S}}{S} \cdot \frac{w(l_i|\Theta^*)}{w(l_i|\Theta_i)}$. Note that $\frac{\hat{S}}{S}$ is a common proportion for all sensors. According to (12), proportionally scaling all calibration coefficients has no impact on the system detection performance. Therefore, we can replace $\frac{\hat{S}}{S}$ with an arbitrary constant γ and have (15). □

We now apply Theorem 1 to the power law decay and the exponential decay to obtain the following two corollaries.

COROLLARY 1. *If the target signal follows the power law decay, the optimal calibration coefficient of sensor i is given by*

$$\gamma_i^* = \gamma \cdot l_i^{k_i-k^*} \cdot \mathrm{e}^{-b_i}, \tag{17}$$

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

$$\Lambda(k) = \frac{\sum_{i=1}^{N} l_i^{-k}}{\sqrt{\sum_{i=1}^{N} \sigma_i^2 \mathrm{e}^{-2b_i} l_i^{2k_i-2k}}}. \tag{18}$$

PROOF. By replacing r_i in (14) with its estimate, i.e., $r_i = \left(\frac{\mathrm{e}^{b_i}}{S_0}\right)^{\frac{1}{k_i}}$, 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

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

where r_i is replaced with $r_i = \left(\frac{\mathrm{e}^{b_i}}{S_0}\right)^{\frac{1}{k_i}}$. As $\frac{\hat{S} S_0 r^{k^*}}{S}$ is a common proportion for all sensors, we can replace $\frac{\hat{S} S_0 r^{k^*}}{S}$ with an arbitary constant γ and have (17). □

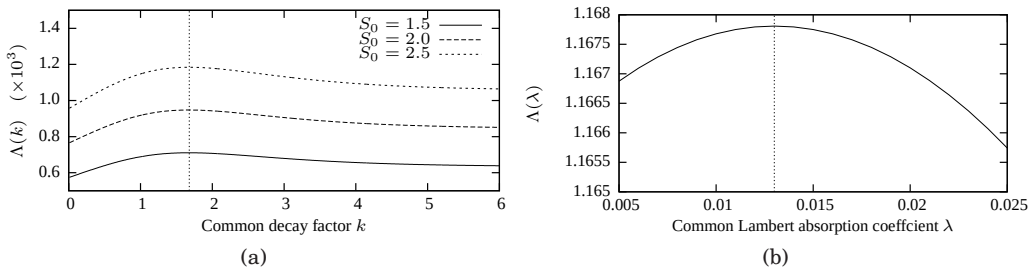


Fig. 4. (a) $\Lambda(k)$ vs. common decay factor k; (b) $\Lambda(\lambda)$ vs. common Lambert absorption coefficient λ.

COROLLARY 2. *If the target signal follows the exponential decay, the optimal system-level calibration coefficient is* $\gamma_i^* = \gamma \cdot \mathrm{e}^{l_i(\lambda_i - \lambda^*)}$, *where* γ *is an arbitrary constant for all sensors and* $\lambda^* = \operatorname{argmax}_{\lambda} \Lambda(\lambda)$, *where* $\Lambda(\lambda) = \frac{\sum_{i=1}^{N} \mathrm{e}^{-\lambda l_i}}{\sqrt{\sum_{i=1}^{N} \sigma_i^2 \cdot \mathrm{e}^{2 l_i (\lambda_i - \lambda)}}}$.

Corollary 2 can be easily proved by replacing the signal decay function $w(\cdot)$ in Theorem 1 with $w(l_i|\lambda_i) = \mathrm{e}^{-\lambda_i \cdot l_i}$. The detailed proof is omitted here.

6.3. System-level Calibration Algorithm

Maximizing $\Lambda(\Theta)$ in (16) 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 Θ^* also suffices if the dimension of Θ is low and each component of Θ has finite range. For instance, for the power law decay, the only parameter needs to be optimized, i.e., k, usually has a finite range, e.g., $(0, 10]$. Moreover, the computation overhead can be controlled by the search granularity.

We now illustrate the above optimization through two numerical examples. Only two nodes are involved in these examples and their distances from the surveillance spot are $8\,\mathrm{m}$ and $18\,\mathrm{m}$, respectively. In the first example, the nodes' power law decay models are given in Figure 1(a). Figure 4(a) plots the objective function $\Lambda(k)$ versus the common decay factor k given various S_0. The numerical results show that k^* is 1.68, which is independent of S_0. Moreover, the calibration coefficients of these two sensors satisfy $\gamma_{41} = 1.98 \cdot \gamma_{48}$. Hence, we can choose $\gamma_{41} = 1.98$ and $\gamma_{48} = 1$. In the second example, the Lambert absorption coefficients of the two sensors are set to be $5000\,\mathrm{cm}^{-1}$ and $8000\,\mathrm{cm}^{-1}$, respectively. Figure 4(b) plots $\Lambda(\lambda)$ versus the common Lambert coefficient λ. The result shows that λ^* is 0.013 or $7692\,\mathrm{cm}^{-1}$ equivalently. Note that $\Lambda(\lambda)$ is independent of S_0. In this example, $\gamma_{41} = 1.067 \cdot \gamma_{48}$.

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 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. SIMULATED 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 simulated testbed experiments in this section and then the simulations in Section 8.

Algorithm 2 System-level calibration and detection

1: **event** $\{\mu_i, \sigma_i^2, \Theta_i | i \in [1, N]\}$ are received **do**
2: **for each** surveillance spot j in all surveillance spots **do**
3: numerically maximize $\Lambda(\Theta)$ given by (16)
4: compute $\gamma_{j,i}$ for each sensor i by (15)
5: compute the optimal detection threshold T_j by (11) 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**

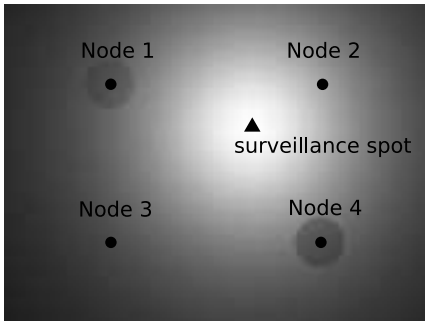


Fig. 5. A screenshot when target appears.

7.1. Experiment Methodology and Settings

In our experiments, four TelosB motes are attached against a LCD screen with resolution of 1024×768 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 power law decay by setting the grayscale of pixels around the light spot, as shown in Figure 5. However, the difference in the sensing performance of the light sensors on the TelosB motes is negligible. To simulate the sensor bias, we reduce the grayscale of the pixels around the sensors by different percentages. Specifically, the grayscale of the pixels within a certain distance (60 pixels in our experiments) from sensor i is reduced by $\delta_i\%$. This approach simulates the bias caused by environment. The experiment settings are listed in Table II. We note that a similar experiment methodology is employed for studying the sensing performance of WSNs in [Hwang et al. 2007; Tan et al. 2010]. The sensors measure the light intensity every 250 milliseconds. Algorithm 1 is implemented in TinyOS. The timeout of the periodical timer in Algorithm 1 is set to be 10 seconds. A sink node is attached on a laptop computer that runs Algorithm 2 implemented using Java. 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.

Section 2 reviews various calibration approaches. However, as they are often designed for specific applications with unique characteristics, they cannot be readily

Table II. Settings and calibration coefficients

Node	Position	$\delta_i(\%)$	l_i	μ_i	σ_i^2	k_i	b_i	γ_i^*	γ_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

employed as the baselines to make fair comparisons. In this paper, 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 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. If sensor i is l_i meters from the surveillance spot, the calibration coefficient for sensor i is calculated as $\gamma_i = s_{\text{std}}(l_i)/s_i(l_i)$, where $s_{\text{std}}(l_i)$ and $s_i(l_i)$ are the signal energies received by the standard sensor and sensor i when the distance from the controlled target is l_i, respectively. The detection threshold is set according to (11) to meet the first constraint of Problem 2. In this intuitive but *ad hoc* approach, the random noises received by sensors are amplified as well. As a result, the performance of the *device-cal* approach can be low. On our testbed, in addition to running Algorithm 1, sensors also transmit all raw readings to the sink to implement the *device-cal* approach. In the *opt-no-cal* approach, the readings of sensors are fused without calibration. The detection threshold is set according to (11), where $\gamma_i = 1$ and sensors' noise models have already estimated by Algorithm 1. The *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. Note that *opt-no-cal* is similar to the approach in [Fabeck and Mathar 2007]. Both approaches aim to construct the optimal data fusion rule at the cluster head based on the estimated sensor models. However, they employ different data fusion models.

7.2. Experiment Results

Several results including sensors' noise and signal decay models as well as the calibration coefficients are reported in Table II. Note that γ_i^* and γ_i' are the calibration coefficients yielded by the *opt-sys-cal* and *device-cal* approaches, respectively. We can draw three 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, the first and fourth sensors have smaller decay factors than others. This is consistent with the simulated sensor biases as shown in Figure 5, i.e., the grayscales of the pixels around Node 1 and 4 are reduced. Third, in our approach, the sensors close to the surveillance spot are allocated with high calibration coefficients. According to (17), 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 6 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 7 plots the

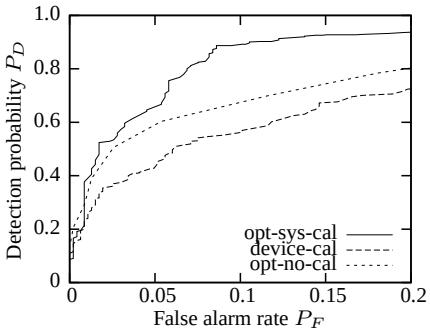


Fig. 6. Receiver operating characteristic (ROC) curves.

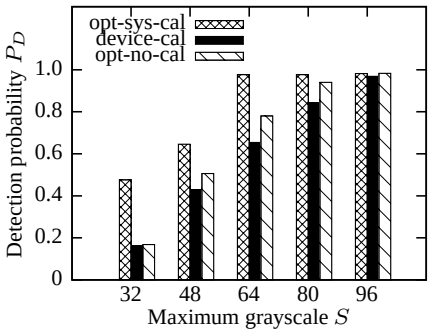


Fig. 7. P_D vs. the grayscale of the light spot.

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 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 synthetic data as well as real data traces collected in the DARPA SensIT vehicle detection experiments [Duarte and Hu 2004].

8.1. Trace-driven Simulations

8.1.1. Settings and Methodology. We use the real data traces collected in the DARPA SensIT vehicle detection experiments [Duarte and Hu 2004], where 75 WINS NG 2.0 nodes are deployed to detect Amphibious Assault Vehicles (AAVs) driving through several intersecting roads. We refer to [Duarte and Hu 2004] 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 8 plots the layout of the sensors and surveillance spots as well as the trajectories of AAV3-5. From Figure 8, we can see that several sensors are close to the trajectories of the vehicle runs. However, for most of the time, the distance between the vehicle and a sensor is large (e.g., tens of meters) such that we can ignore the impact of vehicle movement on the sensor measurement model estimation.

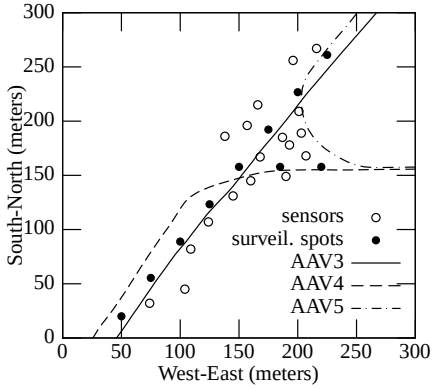


Fig. 8. Sensor layout and trajectories of AAV3-5.

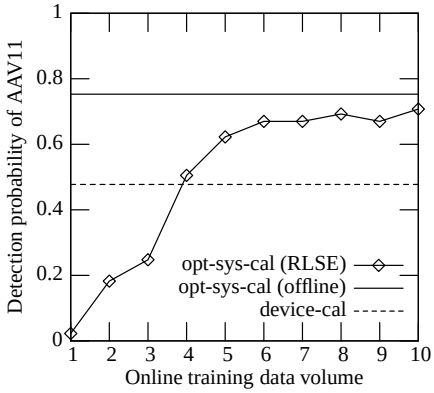


Fig. 9. Convergence of P_D.

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 $300 \times 300\,\mathrm{m}^2$ region shown in Figure 8, 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.

8.1.2. 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 9 plots the detection probability versus the number of data points used by the RLSE algorithm 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 9 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, which is consistent with the performance evaluation of RLSE in Figure 3. 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

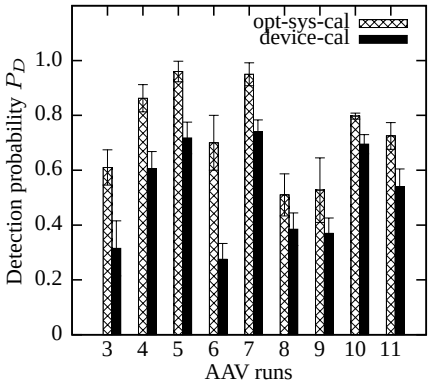


Fig. 10. P_D of different AAVs.

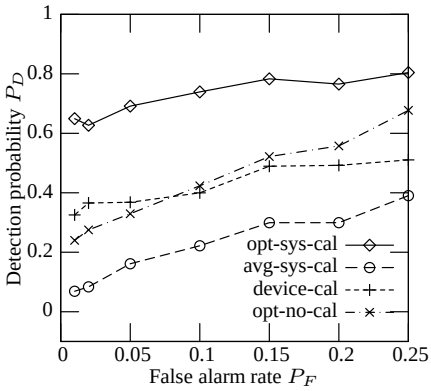


Fig. 11. ROC curves.

various runs. Figure 10 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.

8.2. Simulations based on Synthetic Data

In addition to the trace-driven simulations, we also conduct simulations based on synthetic data, which allow us to evaluate our approach under various settings.

8.2.1. Numerical Settings. A number of sensors are uniformly deployed into a circular region with diameter of 20 meters. The surveillance spot is chosen at the center of the circular region and the sensor closest to the surveillance spot is chosen as the cluster head. The decay factor k_i and reference distance r_i for each sensor i are randomly drawn from $[1.0, 5.0]$ and $[1.0, 2.0]$, respectively. The mean and variance of the Gaussian noise generator for each sensor are set to be one.

The controlled target moves straight through the region and crosses the region center at a constant speed of 1 m/s. Each sensor i samples energy at a frequency of 100 Hz and generates a statistics P_{Di} every second as described in Section 5.1. In the run time, a target randomly appears in the circular region and all sensors fuse their measurements to detect the target. To measure the system detection performance, the target appears for a large number of times (4000 in our simulations). The detection probabil-

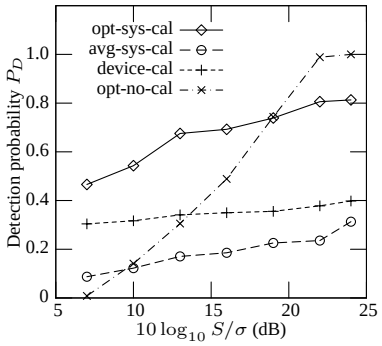


Fig. 12. P_D vs. PSNR.

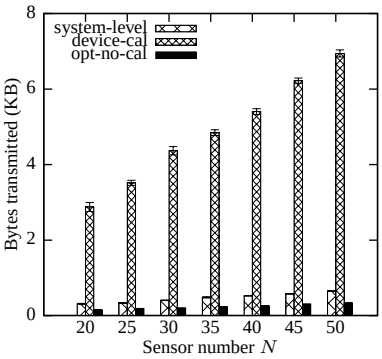


Fig. 13. Communication overhead.

ity is calculated as the fraction of successful detections. The results presented below are the average over 20 runs. In this section, we employ an additional baseline approach, namely, *avg-sys-cal*, where the common r and k are set to be the average of each sensor's r_i and k_i, respectively. We note that the reference distance r_i is usually unknown in practice as discussed in Section 5.1.

8.2.2. Simulation Results. We first evaluate the ROC of the calibrated networks. Total 20 sensors are deployed and the source energy of the targets at run time is 50. Figure 11 plots the ROC curves of various calibration approaches. We can see that our approach significantly outperforms the other three baseline approaches.

We next evaluate the detection performance under various peak signal-to-noise ratios (PSNRs). PSNR is defined as the ratio of source energy to the standard deviation of noise in decibel, i.e., PSNR $= 10\log_{10}\frac{S}{\sigma}$. Figure 12 plots the detection probability versus PSNR when total 20 sensors are deployed. We can see that the detection probability increases with PSNR. This conforms to the intuition that a louder target can be detected more easily. Moreover, our approach significantly outperforms the other two calibration approaches, i.e., *device-cal* and *avg-sys-cal*. Note that *opt-no-cal* is an optimal detection approach without enforcing the identical sensing model for all sensors. Without the second constraint of Problem 2, *opt-no-cal* might outperform our approach in term of detection performance. We observe that the uncalibrated network outperforms the calibrated network when the PSNR is greater than 19 dB. However, as discussed in Section 3.1, many existing data fusion algorithms cannot be readily applied in uncalibrated networks where sensors have different sensing models. Moreover,

in the vehicle detection experiments based on low-cost motes, the PSNRs are usually low to moderate (≤ 17 dB) [Gu et al. 2005].

Finally, we evaluate the communication overhead of various calibration approaches. We employ the link model in [Zuniga and Krishnamachari 2004] with the settings of MICA2 mote to compute the packet reception rate of each link in the network. The multi-hop routing paths to the cluster head are chosen according to the shortest path criterion. The weight of each link is the expected number of transmissions which is a widely adopted routing metric for lossy wireless links [Woo et al. 2003]. We assume that a float number is represented by 2 bytes. Figure 13 plots the total number of data transmissions in the training phase versus the number of deployed sensors. Note that we only account for the communication overhead incurred by the calibration algorithms. Both our *opt-sys-cal* and the *avg-sys-cal* algorithms belong to the *system-level* approach, in which only the parameters of signal decay and noise models are transmitted. In *opt-no-cal*, only the noise parameters are transmitted so that the cluster head can compute the optimal detection threshold. Therefore, the communication overhead for these three approaches is affordable for low-cost sensors such as MICA2 motes [Crossbow]. However, in *device-cal* approach, each signal-distance data pair is transmitted so that the cluster head can centrally compute the mapping functions. Hence, the communication overhead is significantly higher. Moreover, it increases with the volume of data used in the training phase. This result clearly demonstrates the advantage of system-level calibration in reducing communication overhead over device-level approaches.

9. DISCUSSION

In this section, we discuss several issues that have not been addressed in this paper.

9.1. Handling Faulty Sensors

A WSN deployed in a harsh environment likely contains faulty sensors. Different from sensor noises that often follow certain probabilistic distributions, faulty readings are the anomalous data that exceeds normal expected behavior [Clouqueur et al. 2004; Ni et al. 2009], such as outliers. In the context of calibrating sensors' sensing models, a *fault* can be explained as the reading that significantly deviates from the signal decay model. A sensor is said to be a *faulty sensor* if a considerable portion of its readings are faults. The local measurement model estimated by a faulty sensor in the training phase and the faulty readings at run time jeopardize the performance of system-level calibration and run-time detection. From Theorem 1 and Corollary 1, the optimal calibration coefficients of sensors are highly correlated with the local measurement models estimated by sensors. Therefore, a faulty sensor measurement model can lead to incorrect calibration coefficients. To address this issue, we can identify the faulty sensors and exclude them from the system-level calibration and the run-time detection. We now discuss a simple approach to identifying faulty sensors. As the faulty readings significantly deviate from the signal decay model, the transformed data points used to estimate the measurement model at the sensor, (e.g., $\{z_i(t), x_i(t) | t = 1, 2, \ldots\}$ for the power decay law), will not lie on a straight line. Therefore, the linear regression presented in Section 5.1 and Section 5.2 will have large errors. The sum of squared residuals (SSE) is often used to characterize the error of linear regression. For instance, for the power law decay, $\mathrm{SSE} = \sum_t \left(z_i(t) - k_i \cdot x_i(t) - b_i\right)^2$. In addition to the measurement model $(\mu_i, \sigma_i^2, \Theta_i)$, sensor i also transmits its SSE to the cluster head. The cluster head can identify faulty sensors according to sensors' SSEs, for instance, by comparing against a pre-defined threshold. The above approach is a simple heuristics without provable performance. In our future work, we plan to integrate existing advanced outlier detection

algorithms (e.g., [Rousseeuw et al. 1987]) and robust regression algorithms (e.g., [Andersen 2007]) to enhance the robustness of our calibration approach in the presence of faults and faulty sensors.

9.2. Overhead of Calibration Approaches

In Section 8.2, we have evaluated the communication overhead of various approaches in a multi-hop network with realistic link model. In this section, we discuss several insightful analytical results on the communication overhead. To simplify the discussion, we assume that each node can directly communicate with the cluster head and the link is ideal. Moreover, we ignore the communication overhead caused by the calibration process control. Suppose the controlled target broadcasts its position for L times, the communication overhead of the *system-level*, *device-cal*, and *opt-no-cal* approaches are $O((L+1)\cdot N)$, $O(L\cdot N\cdot M)$ and $O(N)$, respectively, where N is the number of sensors and M is the number of measurements when the controlled target is at a certain position. In the *system-level* approach (i.e., *opt-sys-cal* and *avg-sys-cal*), the sensors receive total $L \cdot N$ messages containing the target position during the calibration phase and transmit total N messages containing the local model parameters. In the *device-cal* approach, sensors transmit all raw data with a volume of $O(L \cdot N \cdot M)$. The *opt-no-cal* only needs sensors' noise models, which can be collected by $O(N)$ messages. We can see that the communication overhead of the *device-cal* approach is about M times of the *system-level* approach.

As discussed in Section 2, among various existing calibration approaches, the closest one to our work is [Fabeck and Mathar 2007], which calibrates the detection performance of each sensor to construct the optimal decision fusion rule at the cluster head. We now briefly compare our approach with [Fabeck and Mathar 2007] in terms of communication and computation overhead. In [Fabeck and Mathar 2007], the detection decisions of sensors are collected to the cluster head during L' detections, causing a total of $O(N \cdot L')$ messages. Note that L and L' are large enough numbers for ensuring statistical significance in estimating the signal decay model in our approach and sensor's detection performance in [Fabeck and Mathar 2007]. If L and L' are in the same order, our approach and the approach in [Fabeck and Mathar 2007] have comparable communication overhead. In [Fabeck and Mathar 2007], each sensor only needs to make detections. In our approach, the sensors calculate their local sensing models by regression, which typically needs more computation than detection. However, as proved in Section 6, the local sensing models enable our calibration approach to adapt to the targets with different signal energies at run time. Moreover, we can reconfigure the surveillance spots to adapt to the change of target locations without extra calibration. These features are not available in the approach of [Fabeck and Mathar 2007].

10. 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.

The analytical results and algorithms in this paper are specific to the data fusion model in Section 3.3. Although the fusion model is suitable for the efficient implementation on mote-class cluster heads, in our future work, we plan to extend our approach to address more data fusion models. For instance, we can consider a general data fu-

sion model where the sensor readings are first quantized and then fused with weights. Such a model represents a class of widely adopted decision fusion schemes.

REFERENCES

ANDERSEN, R. 2007. *Modern methods for robust regression*. Sage Publications, Inc.

ASTROM, K. AND WITTENMARK, B. 1994. *Adaptive Control*. Addison-Wesley.

BALZANO, L. AND NOWAK, R. 2007. Blind calibration of sensor networks. In *International Conference on Information Processing in Sensor Networks (IPSN)*. 79–88.

BYCHKOVSKIY, V., MEGERIAN, S., ESTRIN, D., AND POTKONJAK, M. 2003. A collaborative approach to in-place sensor calibration. In *International Conference on Information Processing in Sensor Networks (IPSN)*. 301–316.

CLOUQUEUR, T., SALUJA, K. K., AND RAMANATHAN, P. 2004. Fault tolerance in collaborative sensor networks for target detection. *IEEE Transactions on Computers 53,* 3, 320–333.

CROSSBOW. Mica and mica2 wireless measurement system datasheets.

DUARTE, M. AND HU, Y.-H. 2004. Vehicle classification in distributed sensor networks. *Journal of Parallel and Distributed Computing 64,* 7, 826–838.

FABECK, G. AND MATHAR, R. 2007. In-situ calibration of sensor networks for distributed detection applications. In *The 3rd International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*.

FENG, J., MEGERIAN, S., AND POTKONJAK, M. 2003. Model-based calibration for sensor networks. In *The 2nd IEEE International Conference on Sensors (IEEE Sensors)*. 737–742.

FUWA, K. AND VALLE, B. 1963. The Physical Basis of Analytical Atomic Absorption Spectrometry. The Pertinence of the Beer-Lambert Law. *Analytical Chemistry 35,* 8, 942–946.

GU, L., JIA, D., VICAIRE, P., YAN, T., LUO, L., TIRUMALA, A., CAO, Q., HE, T., STANKOVIC, J. A., ABDELZAHER, T., AND KROGH, B. H. 2005. Lightweight detection and classification for wireless sensor networks in realistic environments. In *The 3rd ACM Conference on Embedded Networked Sensor Systems (SenSys)*. 205–217.

HATA, M. 1980. Empirical formula for propagation loss in land mobile radio services. *IEEE Transactions on Vehicular Technology 29,* 3, 317–325.

HE, T., KRISHNAMURTHY, S., STANKOVIC, J. A., ABDELZAHER, T., LUO, L., STOLERU, R., YAN, T., GU, L., HUI, J., AND KROGH, B. 2004. Energy-efficient surveillance system using wireless sensor networks. In *The 2nd International Conference on Mobile Systems, Applications, and Services (MobiSys)*. 270–283.

HWANG, J., HE, T., AND KIM, Y. 2007. Exploring in-situ sensing irregularity in wireless sensor networks. In *The 5th ACM Conference on Embedded Networked Sensor Systems (SenSys)*. 547–561.

IHLER, A. T., FISHER, J. W., MOSES, R. L., AND WILLSKY, A. S. 2004. Nonparametric belief propagation for self-calibration in sensor networks. In *International Conference on Information Processing in Sensor Networks (IPSN)*. 225–233.

LI, D. AND HU, Y.-H. 2003. Energy-based collaborative source localization using acoustic microsensor array. *EUROSIP Journal on Applied Signal Processing 2003,* 4, 321–337.

LI, D., WONG, K. D., HU, Y. H., AND SAYEED, A. M. 2002. Detection, classification and tracking of targets in distributed sensor networks. *IEEE Signal Processing Magazine 19,* 2, 17–30.

MILUZZO, E., LANE, N., CAMPBELL, A., AND OLFATI-SABER, R. 2008. CaliBree: A Self-calibration System for Mobile Sensor Networks. In *The 4th IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS)*. 314–331.

MOSES, R. AND PATTERSON, R. 2002. Self-calibration of sensor networks. In *SPIE: Unattended Ground Sensor Technologies and Applications IV*. Vol. 4743. 491.

NI, K., RAMANATHAN, N., CHEHADE, M., BALZANO, L., NAIR, S., ZAHEDI, S., KOHLER, E., POTTIE, G., HANSEN, M., AND SRIVASTAVA, M. 2009. Sensor network data fault types. *ACM Transactions on Sensor Networks 5,* 3, 25.

RAMANATHAN, N., BALZANO, L., BURT, M., ESTRIN, D., HARMON, T., HARVEY, C., JAY, J., KOHLER, E., ROTHENBERG, S., AND SRIVASTAVA, M. 2006. Rapid deployment with confidence: Calibration and fault detection in environmental sensor networks. Tech. rep., Center for Embedded Networked Sensing, UCLA.

ROBERTSON, C. AND WILLIAMS, D. 1971. Lambert absorption coefficients of water in the infrared. *Journal of the Optimal Society of America 61,* 10, 1316–1320.

ROUSSEEUW, P., LEROY, A., AND WILEY, J. 1987. *Robust regression and outlier detection*. Vol. 3. Wiley Online Library.

SANTINI, S. AND ROMER, K. 2006. An adaptive strategy for quality-based data reduction in wireless sensor networks. In *The 3rd International Conference on Networked Sensing Systems (INSS)*. 29–36.

SHENG, X. AND HU, Y.-H. 2005. Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks. *IEEE Transations on Signal Processing 53,* 1, 44–53.

TAN, R., XING, G., LIU, X., YAO, J., AND YUAN, Z. 2010. Adaptive calibration for fusion-based wireless sensor networks. In *The 29th Conference on Computer Communications (INFOCOM)*. 1–9.

VARSHNEY, P. K. 1996. *Distributed Detection and Data Fusion*. Springer.

WHITEHOUSE, K. AND CULLER, D. 2002. Calibration as parameter estimation in sensor networks. In *The 1st ACM International Workshop on Wireless Sensor Networks and Applications (WSNA)*. 59–67.

WOO, A., TONG, T., AND CULLER, D. 2003. Taming the underlying challenges of reliable multihop routing in sensor networks. In *The 1st ACM Conference on Embedded Networked Sensor Systems (SenSys)*. 14–27.

ZHUANG, Y., CHEN, L., WANG, X., AND LIAN, J. 2007. A weighted moving average-based approach for cleaning sensor data. In *International Conference on Distributed Computing Systems (ICDCS)*.

ZUNIGA, M. AND KRISHNAMACHARI, B. 2004. Analyzing the transitional region in low power wireless links. In *The 1st IEEE International Conference on Sensor and Ad Hoc Communications and Networks (SECON)*. 517–526.