Resilient Localization for Sensor Networks in Outdoor Environments

YoungMin Kwon Kirill Mechitov Sameer Sundresh Wooyoung Kim
Gul Agha
Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL 61801
{ykwon4,mechitov,sundresh,wooyoung,agha}@cs.uiuc.edu

Abstract

The process of computing the physical locations of nodes in a wireless sensor network is known as localization. Self-localization is critical for large-scale sensor networks because manual or assisted localization is often impractical due to time requirements, economic constraints or inherent limitations of deployment scenarios. We have developed a service for reliably localizing wireless sensor networks in environments conducive to ranging errors by using a custom hardware-software solution for acoustic ranging and a family of self-localization algorithms. The ranging solution improves on previous work, extending the practical measurement range threefold (20–30m) while maintaining a distance-invariant median measurement error of about 1% of maximum range (33cm). The localization scheme is based on least squares scaling with soft constraints. Evaluation using ranging results obtained from sensor network field experiments shows that the localization scheme is resilient against large-magnitude ranging errors and sparse range measurements, both of which are common in large-scale outdoor sensor network deployments.

1. Introduction

Localization is the process which assigns location information to the nodes of a wireless sensor network (WSN). This is a fundamental middleware service, since many WSN applications assume the availability of sensor location information. For example, knowledge of the positions of individual sensor nodes is essential for intrusion detection and target tracking applications. Similarly, geographic routing relies on node locations to forward packets. In general, *self-localization* is critical for deployment of large-scale sensor networks, because manual surveying and entry of node coordinates is often impractical, and equipping each node with a specialized positioning device such as a GPS receiver is usually too costly or does not provide sufficient accuracy.

While many algorithms have been proposed for self-localization of sensor networks, most have only been studied under simulation with idealized conditions. Thus, in order to provide a reliable self-localization service to use in real environments conducive to ranging errors, we have developed a custom hardware-software solution for acoustic ranging and a family of localization algorithms suited to medium- to large-scale outdoor wireless sensor networks. These services have been implemented and evaluated on the MICA2 mote platform [4]. This report describes the fully-functional ranging and localization services, including results from experimental evaluation on a medium-scale outdoor WSN ($3300\mathrm{m}^2$). Additional simulation studies based on experimental parameters demonstrate performance under a wider range of conditions.

The ranging service is based on the time-difference of arrival (TDoA) method – distance is calculated based on the difference in propagation times of radio and acoustic signals. Because sound propagation is simple and predictable in an open environment, this technique allows us to achieve good accuracy and significant range using inexpensive, readily available speakers and microphones. Since the standard sounder-microphone pair on the MICA sensor board yields a detection range of less than 3m on grass, we have retrofitted the sensors with compact loud speakers. To further extend the measurement range and improve reliability, we have incorporated a number of signal processing, statistical filtering and consistency checking techniques.

Based on the experimental characteristics of our ranging service, we have designed a centralized localization algorithm which tolerates sparse measurement data, as well as a distributed variant which enables scalable deployment. Our localization scheme is based on least squares scaling

(LSS) [10], a multidimensional scaling (MDS) technique. Unlike classical MDS, LSS does not require that distance measurements between all pairs of nodes be available. We have further extended LSS with soft constraints to exploit deployment-specific requirements, such as minimum node spacing. Experimental evaluation shows that the localization schemes are robust and resilient not only against sparse range measurements, but also against large-magnitude ranging errors. Such resilience makes the schemes particularly well-suited for large-scale, sparse sensor networks, such as outdoor deployments.

2. Related Work

The Global Positioning System (GPS) is by far the most popular standard for electronic outdoor localization [8]. However, GPS units are typically either too costly or imprecise for wireless sensor nodes ($\pm 6.3\mathrm{m}$ error with 95% confidence when selective availability is turned off [2]). As such, many hybrid methods employ a two-tiered approach in which a set of *anchor* nodes is used to localize non-anchor nodes. Anchors are assumed to know their own locations, while the remaining nodes estimate their distances to anchors and perform multilateration to determine their own locations. These systems primarily differ in how distances to anchors are measured.

The Cricket location support system provides locationing services for indoor mobile nodes [15]. Pre-installed active anchors broadcast their location information over an RF channel together with an ultrasonic pulse. Passive receivers use TDoA to estimate their distances to the anchors. The GPS-less localization algorithm of [3] uses a fixed number of anchors with overlapping coverage. Anchors periodically broadcast radio signals, while mobile nodes use a simple connectivity metric to infer proximity to a subset of these anchors, and localize themselves to the centroid of this set.

The Ad-hoc Positioning System (APS) is a family of distributed localization algorithms based on trilateration [13]. The basic idea is to perform multi-hop propagation of distances to anchors throughout the network, so that every node can trilaterate its position. Four different distance metrics were developed, ranging from minimum hop count and sum of hop lengths – for isotropic, uniform density networks – to local geometric constructions, which require higher anchor density to control error propagation. Another variant of APS relies on sensor nodes able to measure the angle-of-arrival of a signal from an anchor [14].

Another localization method uses anchors which broadcast pre-encoded location information along with the transmission power level [1]. Sensor nodes use a simple power attenuation model to infer distance based on the difference between transmission power level and received signal strength. However, the small, uncalibrated antennas and radios of inexpensive wireless sensor nodes makes it difficult to achieve acceptable precision with this technique.

Convex optimization has been proposed as the basis for a constraint-based localization scheme [5]. In this algorithm, measured data such as RF communication range or angular information from optical devices are used to constrain the feasible node positions. Semidefinite programming (SDP) is then used to find a solution to the localization problem.

The 4-cliques of the measurement graph which constitute *robust quadrilaterals*, invulnerable to discontinuous flexes and flips, have been used for unique localization [12]. Robust quads with an overlap of at least 3 nodes are incrementally merged. Unfortunately, this algorithm localizes very few nodes in the case of sparse, noisy measurements, as found in large outdoor environments, because very few robust quads are found in the data. As such, an approximate solution which localizes more nodes is preferable.

Multidimensional Scaling (MDS) has been proposed as the basis for a centralized robust localization algorithm given a set of distance measurements [18]. One problem with this centralized approach is that it requires distances between all pairs of nodes. As a remedy, distributed approaches have been proposed. One approach is to apply a local MDS-MAP for each node along with its hop count-limited neighborhood [19]. Neighborhoods are then incrementally merged into a global coordinate system. As an alternative to classical MDS, the local map may be computed by directly minimizing the discrepancies between measurements and distances in the estimated map [9].

3. Long Distance Acoustic Ranging

Wireless sensor nodes featuring a sounder and a microphone can use acoustic actuation and sensing to measure inter-node distances. We have chosen the acoustic medium for ranging purposes for three reasons. First, acoustic signal propagation tends to be isotropic on open terrain and to have predictable signal attenuation. Second, acoustic ranging yields reasonable accuracy even at significant measurement ranges. Finally, acoustic sensors (*i.e.*, microphones) and actuators (*i.e.*, sounders) are inexpensive and commonly available on WSN platforms, *e.g.*, MICA2 motes with MTS310 sensor boards [4, 7].

3.1. Hardware Extension

In outdoor settings, where signal-absorbing obstacles are common, and particularly in grassy, uneven and wooded areas, signal attenuation is one of the main challenges for long-distance ranging. Thus sensor nodes must generate sufficiently powerful acoustic signals if we are to achieve reliable ranging measurements at longer distances.

To increase the maximum measurement range of the MICA2 platform, we have augmented the MTS310 sensor board with an inexpensive ($5), off-the-shelf piezo-electric buzzer unit [16]. This unit provides output power of 105dB, compared to 88dB for the original buzzer. The higher output power improves the signal-to-noise ratio (SNR) in high-attenuation environments, allowing nodes to detect acoustic signals at much longer distances. The increased measurement range is critical for large-scale outdoor deployments of sensor networks, where high sensor density is often too costly or impractical to realize.

3.2. TDoA Ranging

Our ranging service is based on the time difference of arrival (TDoA) between radio and acoustic signals, which utilizes the fact that these signals propagate at known but significantly different speeds: approximately 340m/s for sound and almost instantaneously at short distances for radio waves. The TDoA method measures the arrival time difference between radio and acoustic signals originating at the same point to estimate the distances between nodes.

A bare-bones TDoA ranging service operates as follows. Sender and receiver nodes synchronize their clocks, and the sender broadcasts a radio message followed by an acoustic signal (chirp). Each receiving node then detects the chirp and computes the difference in arrival times of the signals, and consequently the distance.

Acoustic ranging in the context of resource-constrained WSN platforms, however, is significantly more complex than this outline. The sensor nodes' processing and storage capacities are tightly limited, as are the capabilities of inexpensive sensors and actuators typically found on such platforms. This in turn places significant limitations on the signal detection methods of the ranging service. We also have to consider several sources of error encountered in TDoA ranging, which in practice turn out to be very significant: *(1)* Clock synchronization and timing effects, *(2)* Acoustic sensing and actuation delays, *(3)* Unit-to-unit variation, *(4)* Signal attenuation, *(5)* Environmental noise, *(6)* Multi-path propagation effects, and *(7)* Unreliable sensing.

Some of these errors have very distinctive characteristics, which we can take into account when making distance estimates. *E.g.*, we expect errors from sources 1, 2 and 3 follow a Gaussian distribution with a fairly small variance. Errors due to 4 and 5 are likely to be geographically correlated, whereas those from sources 6 and 7 may or may not be, depending on the situation.

3.3. Approach

Taking into consideration the sources of error listed above, we have created a ranging service resilient against common types of errors. Range measurement accuracy is improved by employing a sophisticated signal detection mechanism and by performing statistical filtering and consistency checking on the range estimates. We will now describe in some detail our approach to acoustic ranging.

Clock synchronization. The clocks of source and destination nodes must be tightly synchronized to account for the delays incurred in transmission of the radio message. We synchronize source and destination nodes on an ad hoc basis using the very same radio message used for TDoA ranging. In other words, we do not need to establish clock synchrony *a priori*. Our synchronization method utilizes the MAC-layer time stamping from FTSP [11]. The maximum clock rate difference between a pair of nodes is insignificant, resulting in maximum ranging error of about 0.15cm distances up to 30m.

Signal detection. The MTS310 sensor board comes equipped with a hardware phase-locked loop tone detector. Its output is a binary value indicating whether or not a tone in the 4.0–4.5kHz frequency band is present. We denote the time of detection of a chirp at node j by t^j_{detect}.

Our experiments indicate that the tone detector device is not very reliable. In particular, we have observed that the probability of erroneous detection is strongly affected by environmental conditions at the time of measurement. Fortunately, the probability of detecting a tone in a sequence of measurements $b(t)$, $P[b(t) = 1]$, is much higher if a tone is actually present than if only background noise is. We model the output of this tone detector as a binary time series $b(t)$, where

$$P\left[b(t) = 1|\text{signal}\right] \gg P\left[b(t) = 1|\text{no signal}\right]$$

Based on this model, we improve the confidence of acoustic signal detection by accumulating the binary outputs of the tone detector from multiple ranging attempts in a single buffer. The starting positions of multiple signals between the two nodes are correlated; on the other hand, the random background noise triggering the detector is not. We apply *threshold detection* to make the decision: the sum of the samples must exceed the threshold value to be recorded as a true detection, and this must happen for a sufficient number of nearby samples for a chirp to be recognized.

To make the detector more robust, we encode a pattern in the acoustic signal. We use a sequence of identical chirps interspersed with intervals of silence. When detecting the signal, we look at both the chirp and the interval preceding it, allowing us to identify false detections due to noise or echoes that are not part of the pattern. To counteract the effect of echoes, we include small random delays between elements of the pattern. For identifying the tone itself, we apply threshold detection as described above.

In summary, we compute the detection time series $\mathrm{detect}(t)$ as

$$\mathrm{detect}(t) = \delta_k \left[\sum_{i=0}^{n} \delta_T \left(\sum_{j=0}^{m} b_j(t-1) \right) \right]$$

where m is the number of samples accumulated, T is the threshold for signal detection, n is the total number of signal detections, k is the threshold for pattern identification, and $\delta_\theta(x) = 1$ if $x > \theta$ and 0 otherwise. The beginning of the acoustic signal is determined as the minimum value of t that satisfies $\mathrm{detect}(t) = 1$.

Computing the distance. Finally, we compute the distance d_{ij} between source i and destination j as follows. Denoting the non-deterministic message transmission delays at both sender and receiver as δ_{xmit}, we must introduce an additional constant time interval between the transmission of the radio message and the corresponding acoustic signal that is greater than δ_{xmit}. We denote this combined delay δ_{const}. The time difference of arrival can then be expressed as $t^j_{detect} - t^j_{send} - \delta_{const}$, where t^j_{send} is the time at which node i sent the radio message according to node j's clock. Since $t^j_{send} = t^j_{recv} - \delta_{xmit}$, we compute the distance using information locally available at node j as

$$d_{ij} = V_s \cdot \left(t^j_{detect} - (t^j_{recv} - \delta_{xmit}) - \delta_{const} \right)$$

where V_s is the estimate of the speed of sound.

Statistical filtering and consistency checks. Even with this signal detection algorithm, individual range estimates may still be erroneous, whether due to a low detection threshold, hardware malfunction, or some other cause. We make multiple distance measurements for a pair of nodes and filter the results to yield a more accurate estimate of the distance. Depending on the number of measurements available, we take the median or mode value of the measurements, which limits the effect of outliers. The mode operation is more resistant to the effects of uncorrelated large-magnitude errors than the median, but it requires a larger number of measurements to be applicable. Statistical filtering techniques are quite effective at discounting uncorrelated errors caused by random, non-periodic events.

Our ranging service also employs inter-node consistency checks to identify measurements containing errors that may be correlated on a single node (*e.g.*, errors due to faulty hardware or persistent wide-band noise). Bidirectional range estimates between a pair of nodes are discarded if they are inconsistent, and if three nodes have measurements between them, we can use the triangle inequality to identify the inconsistency. A caveat is that these checks cannot identify with certainty which of the measurements is incorrect.

3.4. Experimental Evaluation

We evaluated the performance of the ranging service on a network of 46 MICA2 motes with the modified MTS310 sensor boards. Experiments were performed in a relatively flat, grassy area near an airport, with occasionally loud aircraft engine noise present. The sensors cover a 3300 square meter area, where grass height varies between 10 and 15cm. As for the localization experiments described later in the paper, the sensors were arranged in a 7×7 offset grid pattern with 9.14m and 10.22m grid spacing between the nearest neighbors. However, neither the grid pattern nor the spacing constraints are considered when filtering the ranging experiments described here.

We calibrated the ranging service in the target environment for best performance and determined the appropriate threshold values. A high detection threshold is advantageous in noisy environments to limit false positives; on the other hand, a low threshold is more appropriate in high attenuation environments as it reduces false negatives. For our experiments, the sum of the binary tone detection outputs from 10 chirps must exceed the threshold value of 2 for at least 6 of 32 consecutive samples. Such low thresholds lead to correct detections of weak signals, but they also make the ranging service more vulnerable to false positives. For example, in the time series 021030301503704... a signal would be detected starting at position 5.

The distance measurements in the experiments were affected by several factors, including varied environmental conditions in the deployment area (*e.g.*, grass height, noise) and performance variations between different microphone-speaker pairs. For this reason, we believe the data to be representative of the behavior of the ranging service in a variety of realistic deployments.

Analysis: accuracy. Figure 1 shows the error distribution for the distance measurements across all nodes after filtering and consistency checks. We can identify several distinct features of the error distribution from the figure. There is an approximately zero-mean, Gaussian component of the error with a small variance. This component is most likely due to timing effects, hardware delays and unit-to-unit variation. The fact that the error distribution is virtually zero-mean suggests increasing the number of samples and taking the median or mode is an effective technique for improving accuracy.

Another type of error is present in the measurements. These errors cluster to the right of the mean, caused by overestimation due to late signal detections. The most likely explanation for these errors is that environmental effects (*e.g.*, taller than average grass absorbing the signal more) causes more "misses" in the detection of the early part of the signal. Since these errors are correlated with node positions,

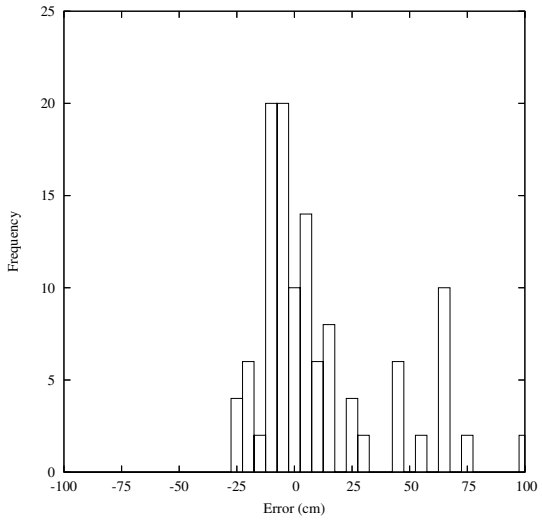

Figure 1. Ranging error histogram.

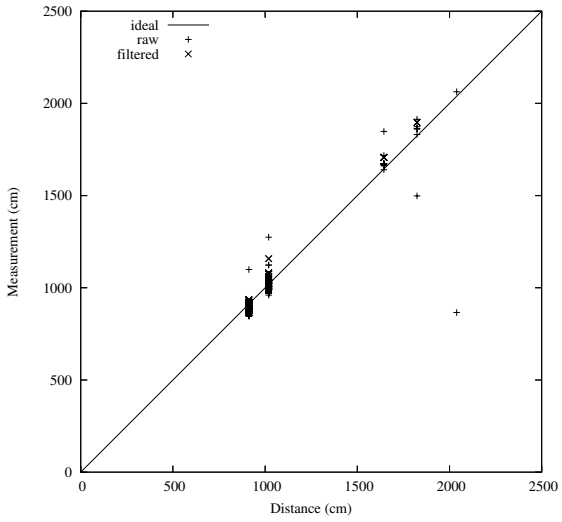

Figure 2. Measurement vs. distance on grass.

internal consistency checking will not be effective even if the number of measurements is increased.

In the unfiltered results, we also see a small number of large-magnitude errors, over 1m and as much as 11m. The causes of the large-magnitude errors can be attributed to non-recurring ambient noise or faulty hardware. It is not likely that such errors are correlated across pairs of nodes, and the vast majority of these errors are eliminated with bidirectional consistency checks.

To better see the effects of filtering on ranging measurements, we plot the ideal, raw, and filtered measurements versus the actual distances (Figure 2). The figure indicates that large-magnitude errors occur more frequently when measuring over a longer distance. Two factors contribute to this effect: *(1)* Since signal power drops significantly at longer distances and the background noise is independent of the distance, the lower SNR increases the probability of false detection. *(2)* It takes longer for a sound signal to reach a more distant node, leaving a bigger window of time for the node to make a false detection. Assuming a constant probability of getting a noise-induced false positive, a more distant node experiences a higher error rate. Fortunately, our filtering mechanism is quite effective at dealing with this problem, eliminating all errors greater than 1.5m.

Analysis: maximum range. To determine the maximum detection range, we have tested the ranging service with a low detection threshold in quiet environments. While the maximum range varies with the features of the environment between each node pair, in our experiments it is 22m on 10–15cm tall grass and over 35m on pavement, on average. Interestingly, higher threshold values needed to make accurate measurements in noisy environments do not significantly reduce the maximum range (15–20m on grass and 30m on pavement). We should note that some speaker-microphone pairs had ranges consistently shorter or longer than the typical values above, likely due to unit-to-unit power and sensitivity variations of the sensor boards.

While environmental conditions usually dictate the maximum detection range in practice, the theoretical maximum range of the service is closely related to the buffer space available in the underlying WSN platform. At distances up to 30m, our ranging service takes up less than 800 bytes of RAM, which leaves plenty of memory available for other applications to run concurrently with the ranging service. This is a significant improvement over the ranging method described in [17], which consumes all available RAM of the MICA2 mote to achieve a theoretical maximum range of less than 16m. To the best of our knowledge, it is also the first fully-functional ranging service for wireless sensor networks offering long range and good precision with a relatively small memory footprint.

4. Resilient WSN Localization with Centralized LSS

In this and the following section we discuss a suite of localization schemes and their experimental evaluation. We develop a localization scheme based on least squares scaling (LSS) [10] with soft constraints.What favors LSS over classical MDS for sensor node localization is that LSSdoes not require distance measurements between all node pairs. Even with some missing measurements, it still yields ac-

ceptable results. Moreover, we may use different weights for measurements depending on their confidence levels and easily incorporate constraints such as minimum node separation into the minimization process to further improve results.

We first describe a centralized version and demonstrate its resilience against measurement errors and omissions. We then extend it to a distributed version asscalable alternative for use in large-scale deployments (Section 5). In figures which show localization results, squares and crosses represent non-anchor nodes' actual and computed locations, respectively. Each localized non-anchor node has its square and cross connected with a line. Lone squares are non-anchor nodes that were not localized.

4.1. Algorithm

Multidimensional scaling is *"any procedure which starts with the 'distance' between a set of points and finds a configuration of points, preferably, in a small number of dimensions, usually 2 or 3"* [10]. Here, a configuration refers to a set of coordinate values. When distances between nodes are available, MDS [19, 18, 10] finds their relative coordinates. In localization using classical MDS, the input distance matrix is transformed to a quadratic matrix of coordinates via double averaging. Then, the singular value decomposition (SVD) is applied to the quadratic matrix to calculate its principal components. The first two principal components are the configuration sought. One critical requirement is that distances between all pairs of nodes be known *a priori*.

An alternative technique is LSS [10], which seeks a configuration $C = \{(x_i, y_i) : i = 1, \ldots, n\}$ from a set of distances $D_{\text{full}} = \{d_{ij} : i, j = 1, \ldots, n\}$ by minimizing the unweighted error function E_u:

$$E_u = \sum_{d_{ij} \in D_{\text{full}}} (\hat{d}_{ij} - d_{ij})^2,$$

where $\hat{d}_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$, and d_{ij} is the measured distance between points (x_i, y_i) and (x_j, y_j). An important property of LSS is that it still works using only a subset of D_{full}. This property allows LSS-based localization to tolerate sparse measurement data.

The error function is the sum of squares of differences between estimated distances and corresponding measured distances. As a result, errors in distance measurement are squared, too. Therefore, weighting distance measurements according to their confidence helps limit the effect of measurement errors on localization results. Statistical entities (*e.g.*, standard deviation) can make a good choice for such weights. We extend the error function E_u to accommodate different weights by defining E_w:

$$E_w = \sum_{d_{ij} \in D} w_{ij} \cdot (\hat{d}_{ij} - d_{ij})^2$$

where $D \subseteq D_{\text{full}}$ is a set of distance measurements from a ranging service.

In many sensor deployments, a minimum distance between nodes can be known in advance: unless nodes are deployed via a purely random process, it is unlikely that two nodes will be placed very close together. Furthermore, our penalty-based constraint enforcement approach allows some nodes to violate the minimum separation condition, with locally distorted results. LSS allows us to incorporate this minimum spacing constraint into localization as a *soft constraint* [6]. Using the soft constraint, we penalize pairs of nodes which do not have distance measurements from the ranging service and whose assigned coordinates violate the minimum spacing constraint, so that any output solution would become more consistent. This can be visualized as straightening a plane which is incorrectly folded. Note that the set of penalized pairs dynamically changes as minimization progresses. With the soft constraint, the error function which we seek to minimize becomes:

$$E = E_w + \sum_{d_{ij} \notin D} w_D \cdot \left(\min(\hat{d}_{ij}, d_{min}) - d_{min}\right)^2$$

where d_{min} is the minimum node spacing and w_D is the weight for the soft constraint. Note that $w_{ij} = 0$ for pairs of nodes which do not have distance measurements in D.

We use gradient descent to find a configuration that minimizes the error term. We can also use other methods such as simulated annealing. We update coordinates of the nodes at each time step using the rules

$$[\mathbf{x}^{t+1}, \mathbf{y}^{t+1}] = [\mathbf{x}^t, \mathbf{y}^t] - \alpha \cdot \nabla E|_{[\mathbf{x}^t, \mathbf{y}^t]},$$

where $\nabla E = \left[\frac{\partial E}{\partial x_1}, \ldots, \frac{\partial E}{\partial x_n}, \frac{\partial E}{\partial y_1}, \ldots, \frac{\partial E}{\partial y_n}\right]$ is the gradient of E and α is a step size that is either a small constant or a value that minimizes E along the line in the gradient direction at step t. Without the soft constraint,

$$\begin{aligned} \left.\frac{\partial E}{\partial x_i}\right|_{[\mathbf{x}^t, \mathbf{y}^t]} &= \left.\frac{\partial E_u}{\partial x_i}\right|_{[\mathbf{x}^t, \mathbf{y}^t]} \\ &= w_{ij} \cdot \sum_{d_{ij} \in D} \frac{(\hat{d}^t_{ij} - d_{ij}) \cdot (x^t_i - x^t_j)}{\hat{d}^t_{ij}}, \end{aligned}$$

where $\hat{d}^t_{ij} = \sqrt{(x^t_i - x^t_j)^2 + (y^t_i - y^t_j)^2}$. With the soft constraint, if d_{ij} is not defined and $\hat{d}^t_{ij} < d_{min}$,

$$\begin{aligned} \left.\frac{\partial E}{\partial x_i}\right|_{[\mathbf{x}^t, \mathbf{y}^t]} &= \left.\frac{\partial E_u}{\partial x_i}\right|_{[\mathbf{x}^t, \mathbf{y}^t]} \\ &+ w_D \cdot \sum_{d_{ij} \notin D} \frac{(\hat{d}^t_{ij} - d_{min}) \cdot (x^t_i - x^t_j)}{\hat{d}^t_{ij}}. \end{aligned}$$

$\left.\frac{\partial E}{\partial y_i}\right|_{[\mathbf{x}^t, \mathbf{y}^t]}$ can be derived similarly.

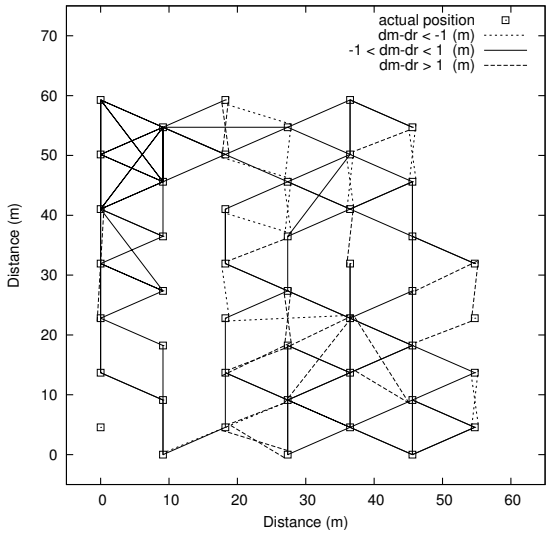

Figure 3. Sensor layout and measurements used for LSS localization experiments. dm: measured, dr: real distance.

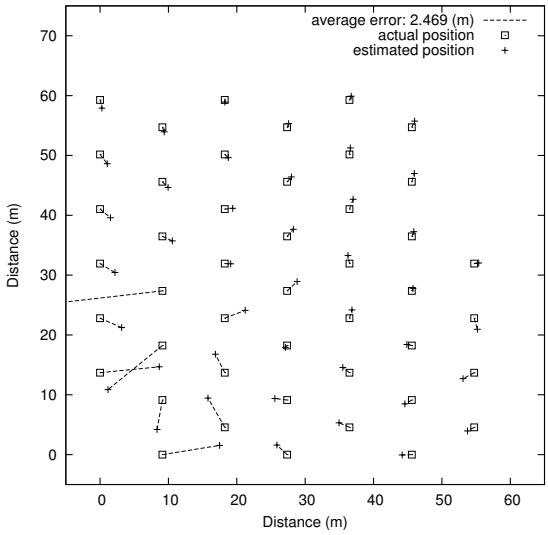

Figure 4. Centralized LSS localization with a minimum spacing constraint.

To avoid local minima, the gradient descent starts each round of minimization with a seed position obtained by perturbing the best result so far. This is analogous to the random movement of the simulated annealing algorithm [20]. This process is repeated until a reasonable minimum is reached or some maximum computation time limit expires.

4.2. Experimental Evaluation

To evaluate the resilience of centralized LSS localization, we performed an experiment with 45 MICA2 motes equipped with standard sensor boards and our custom loudspeakers in a $60\mathrm{m} \times 55\mathrm{m}$ grassy area, using the grid layout of Figure 3. The minimum inter-node spacing was 9.14m. To allow a margin of error, we used an 8.5m soft constraint, with w_{ij} = 1 and w_D = 10. As we increased w_D, the rate of convergence similarly increased.

Figure 4 compares actual and estimated node positions. The computed coordinate system was translated and rotated to line up with the actual node coordinates. Most errors are found in the bottom left quadrant; the overall average localization error is 2.47m. Without the largest 5 errors, the average improves to 1.5m. As seen in Figure 3, lack of measured distances allows the two nodes in the (0∼10,10∼20) area to be swapped. Adding a simulated measurement of 10.2 between nodes (9.14, 18.28) and (18.28, 13.71) corrected the position estimates. The lower three nodes are rotated because of large underestimation (*i.e.,* $<$ 1m) of the distance between nodes (9.14,0) and (18.28,9.14) and large overestimation of the distance between (18.28,4.57) and (27.42,0). Lack of measurements between the second and third column creates a hole, attracting the estimated coordinates of the nodes in the upper left and lower right quadrants, while the concentration of overestimated distances in the middle pushes the coordinates in the upper right quadrant outward. Note also that the minimum constraint prevents nodes in the (50∼60,0∼40) area from being flipped inwards.

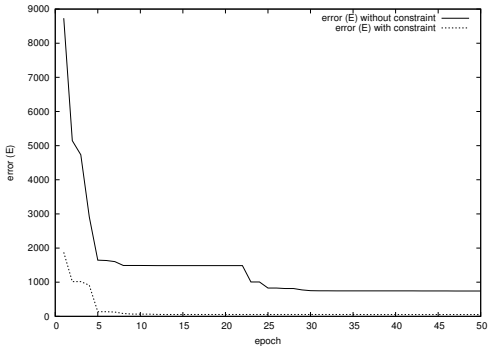

Figure 5. Gradient descent error vs. time.

To examine the significance of the soft constraint, we conducted the same localization experiment without it. The estimated positions did not converge anywhere near the actual positions after several hours of minimization.

Figure 5 shows how soft constraints improve error minimization. The error function has more terms with the soft constraint than without, which are all squared, hence posi-

tive. Thus, although the error function without constraint terms has a smaller global minimum, the soft constraint greatly reduces the time to reach a global minimum.

5. Distributed LSS Localization

The centralized LSS localization algorithm is resilient against sparse distance measurements and large measurement errors. Unfortunately, it does not scale well as networks grow in size. As more nodes are added, the number of terms in the error function increases, as does the number of local minima. In this section, we extend the centralized algorithm to a more scalable distributed version by performing localization at each node on its n-hop neighborhood.

5.1. Algorithm

For the distributed LSS localization algorithm, we assume there is a local broadcast mechanism so a node can send data to its n-hop neighbors. The distributed localization algorithm consists of three steps: neighborhood localization, calculation of transforms between the local coordinate systems of adjacent nodes, and alignment of local coordinate systems to a global coordinate system.

Step 1. Neighborhood Localization Each node collects distance measurements to and amongst its neighbors, *i.e.,* those nodes within direct measurement range. Given the measurements, each node uses the LSS localization algorithm to find a configuration of itself and its neighbors in a local relative coordinate system.

Step 2. Calculating Transforms and Pairwise Transformation The next step is to find a transform between the local relative coordinate systems for each pair of neighboring nodes. Let (u_i, v_i) and (x_i, y_i) represent coordinates of a point in a source and a target coordinate system, respectively. A rigid transform between the two coordinate systems is a composition of translation, rotation, and reflection. This can be written in a 3×3 matrix using the homogeneous coordinate system as follows,

$$[x_n, y_n, 1] = [u_n, v_n, 1] \cdot \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ f\sin\theta & f\cos\theta & 0 \\ t_x & t_y & 1 \end{bmatrix},$$

where (t_x,t_y) is a translation vector, θ is a rotation angle, and $f \in \{1, -1\}$ is a reflection factor. Calculating a transform corresponds to finding t_x, t_y, θ and f.

We find the transform $\mathbf{T}$ between the coordinate systems of two nodes a and b using their shared neighbors. Let C be the set of shared neighbors of nodes a and b which have coordinates in both local coordinate systems. A straight forward method is to use minimization. We find two solutions (θ, t_x, t_y, f) = argmin E_f for f=1, -1, where

$$E_f = \sum_{n \in C} (x_n - \hat{x}_{n,f})^2 + (y_n - \hat{y}_{n,f})^2,$$

$$[\hat{x}_{n,f}, \hat{y}_{n,f}, 1] = [u_n, v_n, 1] \cdot \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ f\sin\theta & f\cos\theta & 0 \\ t_x & t_y & 1 \end{bmatrix}.$$

We then take the solution with the smaller error as the transform. Although this procedure returns fairly accurate results, it is too computationally intensive to implement on resource-constrained WSN platforms such as the MICA2.

Thus we have developed an alternate method to find a transform which is slightly less accurate, but computationally tractable for many WSN platforms. The idea is to view translation between two nodes' coordinate systems as translation between the centers of mass of C in the two coordinate systems. Let the center of mass of C in the source's coordinate system, (μ_u, μ_v), be $(\sum_{n \in C} u_n/|C|$, $\sum_{n \in C} v_n/|C|)$. C's center of mass in the target's coordinate system can be defined similarly. Then the simplified transformation is a sequence of three steps: translation by $(-\mu_u, -\mu_v)$, rotation by angle θ with possible reflection, and translation by (μ_x, μ_y). The rotation angle θ is chosen to minimize the expression

$$\sum_{n \in C} \left\| \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \cdot \begin{bmatrix} u_n - \mu_u \\ v_n - \mu_v \end{bmatrix} - \begin{bmatrix} x_n - \mu_x \\ y_n - \mu_y \end{bmatrix} \right\|^2$$

Considering the condition that the derivative of the above formula is zero, the rotation angle θ should satisfy the equation

$$[C_{xu} + C_{yv}, C_{xv} - C_{yu}] \cdot \begin{bmatrix} \sin\theta \\ \cos\theta \end{bmatrix} = 0$$

where covariances C_{xu}, C_{yv}, C_{xv} and C_{yu} are defined as $C_{\alpha\beta} = \sum_{n \in C} (\alpha_n - \mu_\alpha) \cdot (\beta_n - \mu_\beta)/|C|$. Note that both θ and $\theta + \pi$ satisfy the equation; we choose the solutions which minimizes the error E. Putting them together, the transform $\mathbf{T}_f$ is:

$$\begin{aligned} \mathbf{T}_f &= \mathbf{T}_{-\mu_u, -\mu_v} \cdot \mathbf{R}_{f,\theta} \cdot \mathbf{T}_{\mu_x, \mu_y}, \\ \mathbf{T}_{\alpha,\beta} &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \alpha & \beta & 1 \end{bmatrix}, \\ \mathbf{R}_{f,\theta} &= \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ f\sin\theta & f\cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}, \end{aligned}$$

where μ_u, μ_v, μ_x, μ_y and θ are computed values. We choose whichever of $\mathbf{T}_1$ or $\mathbf{T}_{-1}$ yields smaller E.

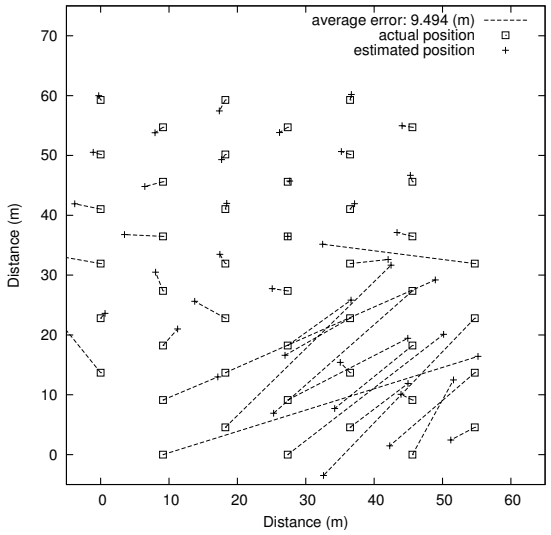

Figure 6. Distributed LSS, same measurements as Figure 4.

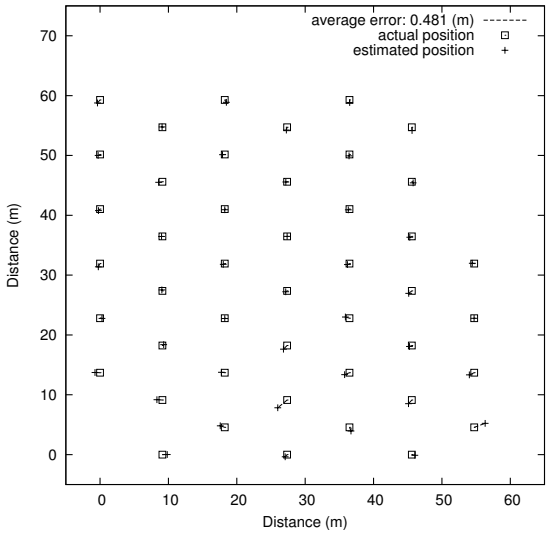

Figure 7. Distributed LSS extended with 185 additional simulated distances.

Step 3. Alignment As the last step, all the coordinate systems in the network are successively aligned. After alignment, each node computes its position in the global coordinate system. Starting from the root node, a node broadcasts a vector representation of the origin of the global coordinate system and two orthonormal axis vectors that span the local coordinate system. When a node receives the three vectors in the sender's local coordinate system (*i.e.*, $\mathbf{o}$, $\mathbf{x}$, and $\mathbf{y}$), it finds the corresponding transform $\mathbf{T}$ and computes their representations in its local coordinate system (*i.e.*, $\hat{\mathbf{o}}$, $\hat{\mathbf{x}}$, and $\hat{\mathbf{y}}$). Finally, the node forwards the transformed vectors to its neighbors. In the mean time, it computes its position in the global coordinate system as $((\mathbf{p} - \hat{\mathbf{o}}) \cdot \hat{\mathbf{x}}, (\mathbf{p} - \hat{\mathbf{o}}) \cdot \hat{\mathbf{y}})$ where $\mathbf{p}$ is the vector representation of its position in its local coordinate system. Eventually, all nodes in the network compute their positions in the root's coordinate system.

This algorithm requires two local data exchanges per node and one round of flooding. It is more scalable than the centralized approach because minimization, the most time-consuming component of LSS localization, is performed concurrently on small neighborhoods.

5.2. Evaluation

Figure 6 shows experimental results for distributed LSS localization using the same set of measurements as in Section 4. The node at (27, 36) is the root node for the global coordinates system. The figure shows that approximately one half of the nodes have very large localization errors. We examined the results and found that the bad transform of a pair of nodes caused large localization errors which were amplified and propagated. We suspect the cause was lack of distance measurements: only 101 distinct distance measurements between pairs were available for the 45 nodes. Assuming that maximum measurable distance is 22m there are 286 possible distance measurements within the grid layout of Figure 3. As such, we actually measured only 35% of the feasibly measurable distances.

To verify the cause of poor performance, we repeated the localization procedure by adding simulated data. The measured data set contained distances between 101 distinct pairs of nodes, all of which were less than 22m apart. We added 185 simulated measurement so that all pairs of nodes less than 22m apart were connected by either a real or simulated distance (*i.e.,* we added 19% of the 990 possible node pairs). Measurement errors were simulated in these additional pairs by taking the real distance and adding a perturbation drawn from the Gaussian distribution $N(\mu = 0; \sigma = 0.33\text{m})$. Figure 7 shows the results. As expected, we achieved fairly good localization results; all nodes were localized, with an average localization error of 0.48m. For a run with only 128 simulated distances (80% of all pairs within 22m, and 13% of all possible pairs) the average localization error was 2.12m.

6. Conclusions

We have designed and implemented a fully functional localization system for wireless sensor networks, in particular for the MICA2 mote platform. By increasing the out-

put signal power with a louder speaker and applying advanced filtering and consistency checking techniques, we have significantly extended the ranging service's measurement range in comparison to previous work, even in environments conducive to measurement errors (up to 22m maximum, 11m reliable on grass and 30m maximum, 25m reliable on pavement). This represents a threefold improvement over previous work, while maintaining a distance-invariant median measurement error of about 1% of maximum range. The addition of a minimum node spacing soft constraint to the LSS localization algorithm is successful in overcoming the detrimental effects of sparse, noisy ranging measurements. Demonstrating good performance in realistic conditions, the system renders feasible deployment of large-scale autonomous WSNs in outdoor environments without prior surveying or manual configuration.

Several avenues of research remain to be explored. It may be possible to further improve the quality of ranging estimates, particularly through consistency checking, if additional sensing modalities are available to use in conjunction with acoustics. We are currently conducting research to improve localization performance and accuracy by exploiting additional information, such as the deployment pattern or node density [21]. Finally, the distributed localization algorithm needs to be improved to the point where its results approach the accuracy of the centralized algorithm before we can reliably apply this methodology to self-localization of very large sensor networks.

Acknowledgments

This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Award No. F33615-01-C-1907 and by the ONR Grant N00014-02-1-0715.

References

[1] P. Bergamo and G. Mazzini. Localization in sensor networks with fading and mobility. In *Proceedings of the 13th IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC 2002)*, Lisboa, Portugal, September 2002.

[2] I. G. E. Board. http://www.igeb.gov/sa/.

[3] N. Bulusu, J. Heidemann, and D. Estrin. GPS-less low cost outdoor localization for very small devices. *IEEE Personal Communications Magazine*, 7(5):28–34, October 2000.

[4] Crossbow Technology, Inc. http://www.xbow.com/.

[5] L. Doherty, K. S. J. Pister, and L. E. Ghaoui. Convex position estimation in wireless sensor networks. In *Proceedings of the 20th Conference of the IEEE Communications Society (IEEE INFOCOM)*, pages 1655–1663, 2001.

[6] R. Fletcher. *Practical Methods of Optimization*. Wiley, 1987.

[7] J. L. Hill. *System Architecture for Wireless Sensor Networks*. PhD thesis, University of California, Berkeley, 2003.

[8] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins. *Global Positioning System: Theory and Practice*. Springer-Verlag, 4th edition, 1997.

[9] X. Ji and H. Zha. Sensor positioning in wireless ad-hoc sensor networks with multidimensional scaling. In *Proceedings of the 23rd Conference of the IEEE Communications Society (IEEE INFOCOM)*, 2004. (to apear).

[10] P. M. Lee. Multivariate Analysis: Lecture Notes. Chapter 8: Multidimensional Scaling. http://www.york.ac.uk/depts/maths/teaching/pml/mva/.

[11] M. Maroti, B. Kusy, G. Simon, and A. Ledeczi. The Flooding Time Synchronization Protocol. Technical Report ISIS-04-501, Institute for Software Integrated Systems, Vanderbilt University, 2004.

[12] D. Moore, J. Leonard, and D. R. S. Teller. Robust distributed network localization with noisy range measurements. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*, 2004.

[13] D. Niculescu and B. Nath. Ad hoc positioning system (APS). In *GLOBECOM*, November 2001.

[14] D. Niculescu and B. Nath. Ad-hoc positioning system using AoA. In *Proceedings of the IEEE/INFOCOM 2003*, San Francisco, CA, April 2003.

[15] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan. The Cricket location-support system. In *Mobile Computing and Networking*, pages 32–43, 2000.

[16] RadioShack Corporation. 102dB Siren Sound Piezo Buzzer, Part No. 273-079. http://www.radioshack.com/.

[17] J. Sallai, G. Balogh, M. Maroti, and A. Ledeczi. Acoustic ranging in resource constrained sensor networks. Technical Report ISIS-04-504, Institute for Software Integrated Systems, 2004.

[18] Y. Shang and W. Ruml. Improved MDS-based localization. In *IEEE INFOCOM*, 2004.

[19] Y. Shang, W. Ruml, Y. Zhang, and M. P. Fromherz. Localization from mere connectivity. In *Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking & Computing*, pages 201–212, Annapolis, Maryland, USA, 2003. ACM Press. http://doi.acm.org/10.1145/778415.778439.

[20] Stuart Russell and Peter Norvig. *Artificial Intelligence A Modern Approach*. Prentice Hall, 1995.

[21] S. Sundresh, Y. Kwon, K. Mechitov, W. Kim, and G. Agha. Localization of sparse sensor networks using layout information. Technical Report UIUCDCS-R-2005-2525, Dept. of Computer Science, University of Illinois at U-C, 2005.