Report No. UIUCDCS-R-2004-2449 **UILU-ENG-2004-1751**

Resilient Localization for Sensor Networks in Outdoor Environments

by

YoungMin Kwon, Kirill Mechitov, Sameer Sundresh, Wooyoung Kim and Gul Agha

June 2004

Resilient Localization for Sensor Networks in Outdoor Environments

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

Abstract

A process which computes the physical locations of nodes in a wireless sensor network is called *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 forwardi packets, obviating the need for large routing tables. 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.

A number of algorithms have been proposed for self-localization of sensor networks. These algorithms may be characterized as range-based or range-free, depending, respectively, on whether or not they require distance measurements between nodes. Unfortunately, most of the proposed schemes have only been studied under idealized conditions such as isotropic networks, high node density, or very high precision ranging. Since these assumptions are generally very difficult to guarantee in large-scale outdoor sensor network deployments, the real-world accuracy and reliability of these schemes is unclear.

In order to provide a reliable self-localization service to use in environments conducive to ranging errors, we have developed a custom hardware-software solution for acoustic ranging and a family of localization algorithms which are particularly suitable for medium- to large-scale outdoor wireless sensor networks. The ranging and localization services have been implemented and evaluated on the MICA2 mote platform [5] over the past three years. This report describes implementation details of the fully-functional ranging and localization algorithms. Each service's description is accompanied with experimental evaluation results obtained with actual range measurement data from medium-scale (3000m^2) WSN deployments. To further understand performance under different conditions, we have also conducted additional simulation studies of the localization algorithms, with distance measurements randomly generated using parameters collected from the experiments.

The ranging service is based on the time-difference-of-arrival (TDoA) method. Specifically, it uses the difference between the arrival times of synchronized radio and acoustic signals to estimate the distance between two nodes. This technique was chosen because acoustic signals tend to have predictable isotropic propagation in open environments, and enable reasonably accurate distance measurements at significant range. Furthermore, inexpensive speakers and microphones are readily available.

The original sounder-microphone pair of the MICA sensor board yields a detection range of less than 3m on grass. In order to increase the reliable acoustic detection range, we have extended the MICA sensor board with a louder speaker. This hardware extension allows sensor nodes to make range measurements to a sufficient number of neighbors for successful localization even in low-density deployments. To further extend the measurement range and improve reliability, we have incorporated a number of filtering techniques into the ranging service, including a signal processing algorithm to better distinguish signals from noise, *statistical filtering*, and *consistency checking*.

We have designed a centralized localization algorithm which tolerates sparse measurement data, as well as a distributed variant which enables scalable deployment. Specifically, we have developed a localization scheme based on least squares scaling (LSS) [11], a multidimensional scaling (MDS) technique. MDS is a family of analysis techniques which find a hidden structure underlying the proximity data in a set of inputs. Unlike classical MDS, LSS does not require that distance measurements between all pairs of nodes be available. In particular, we have adapted LSS with soft constraints to exploit deployment-specific requirements such as minimum node spacing. Empirical evaluation based on ranging results obtained from field experiments in real deployments 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) [9] is by far the most popular standard for electronic outdoor localization. GPS receivers use the time difference between emission and reception of a signal from a satellite (called *pseudo-random code*) to calculate the distance to the satellite. At least four satellites are used to unambiguously locate a position through three-dimensional trilateration. With selective availability[1] turned off (as of May 1, 2000) [2], GPS error is in a 95% confidence interval of ±6.3m, a fairly large error magnitude for wireless micro-sensors with limited communication range. As such, many GPS-less and hybrid methods have been proposed for wireless sensor networks.

Many hybrid methods employ a two-tiered approach in which a set of *anchor* nodes is used to localize non-anchor nodes. Anchor nodes are assumed to know their own location (*e.g.*, by manual survey and entry, careful deployment, or GPS measurements). Sensor nodes estimate their distances to the anchor nodes and apply triangulation or multilateration to determine their locations. The variations lie in the methods used to estimate distances to the anchors.

The Cricket location support system [15] uses a combination of active anchors and passive ultrasonic receivers to provide locationing services. This system was proposed for mobile nodes in an indoor environment, though it can also be used with static nodes. The pre-installed active anchors broadcast their location information over an RF channel together with an ultrasonic pulse. A mobile node uses the TDoA method to estimate the distance to an anchor.

The GPS-less localization proposed in [3] uses a fixed number of anchors with overlapping regions of coverage placed at known positions. The 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 the selected anchors.

The Ad-hoc Positioning System (APS) is a family of distributed localization algorithms based on trilateration [13]. The idea is to propagate distance information to anchors throughout the network so that non-anchor nodes can estimate distances to an enough number of anchors. Four different techniques are proposed,

1. DV-hop, which maintains minimum hop counts to anchor nodes for each node and computes average distance per hop,

[1] The US Government's intentional degradation of the GPS signals available to the general public.

2. DV-distance, which propagates actual node-to-node distances and maintains minimum path lengths to anchors,

3. Euclidean, which successively computes distances to anchors using the fact that, for a quadrilateral, if the lengths of all the sides and one diagonal are known, the length of the other diagonal is uniquely determined, and

4. DV-coordinate, which first constructs relative coordinate systems for nodes locally and then reconcile them.

The DV-hop and DV-distance techniques work well only for isotropic networks with uniform node density. While the Euclidean and DV-coordinate techniques are more resilient to anisotropic layouts, a high anchor density is required to keep error propagation under control. Another variant of APS relies on sensor nodes equipped with an antenna array to measure the angle-of-arrival of a signal from a landmark [14]. Sensor nodes use triangulation to compute their location from the coordinates of the landmarks.

Another localization method uses anchors pre-encoded with their location, where anchors broadcast not only their location information but also the transmission signal strength [1]. Sensor nodes use a simple power attenuation model to infer the distance based on the difference between the transmission power level encoded in a message and the received signal strength. However, it can be difficult to achieve acceptable precision using this technique on inexpensive mass-produced wireless sensor nodes with small, uncalibrated antennas and radios.

Convex optimization has been proposed as the basis for a constraint-based localization scheme [6]. In this algorithm, measured data are used to constrain the feasible node positions. As the number of constraints increases, the intersection of the feasible sets is reduced. Semidefinite programming (SDP) is used to solve the localization problem where the convex constraints considered are RF communication range, angular constraints via optical devices, *etc.*

Multidimensional Scaling (MDS) has been proposed as the basis for a centralized robust localization algorithm [18]. MDS is a technique to calculate positions of nodes given a set of distances. One problem with this centralized approach is that it requires distances between all pairs of nodes. As a remedy for this impractical requirement, distributed approaches have been proposed [10, 19]. In [19], a local MDS-MAP is applied for each node with neighboring nodes within certain hop range. After this step, each neighborhood is incrementally included into a global coordinate system. Similar steps are taken in [10]. However in this approach, the local map is computed by directly minimizing the discrepancies between measured distances and distances from estimated positions, instead of running a classical MDS algorithm.

3 Long Distance Acoustic Ranging

Wireless sensor nodes with a microphone and a sounder (*e.g.*, a MICA2 mote with a standard MICA sensor board [5, 8]) can exploit acoustic actuation and sensing to measure inter-node distances. We have chosen the acoustic medium for ranging purposes for three reasons. First, acoustic signals tend to be isotropic and to have predictable signal attenuation. Second, acoustic ranging yields reasonable accuracy, while achieving significant measurement range. Finally, acoustic sensors (*i.e.*, microphones) and actuators (*i.e.*, speakers) are inexpensive and commonly available on WSN platforms.

In particular, we use TDoA with radio and acoustic signals which utilizes the fact that the signals travel at known, but different, speeds (340m/s for sounds; almost instantaneous at short distances radio signals). The TDoA method measures the arrival time difference between radio and sound signals to estimate the distance between two nodes. A familiar example is how the distance from a lightning strike is estimated by counting the number of seconds between seeing the lightning and hearing the thunder. Achieving centimeter-level precision over short distances on resource-constrained sensor nodes, however, requires much finer measurement granularity and robust signal detection. In this section, we describe the hardware extension we have made to the MICA sensor board to increase the effective measurement range and the software techniques we have incorporated into the ranging service to obtain the desired accuracy and robustness, followed by evaluation results from field experiments.

3.1 Baseline Acoustic Ranging

A bare-bones acoustic ranging service based on TDoA would consist of the following steps:

1. Source and destination nodes synchronize clocks.
2. The source broadcasts a radio message followed by an acoustic signal (chirp).
3. Destination nodes receive the radio message and detect the chirp.
4. Each receiving node then computes the difference in the arrival times of the signals and subsequently the distance.

There are several points worth noting with respect to acoustic ranging in the context of resource-constrained WSN platforms such as MICA2 motes.

Clock Synchronization

The clocks of source and destination nodes should be tightly synchronized to accurately measure the delay incurred in the transmission of the message over the radio. We synchronize a source and a sink for a short period of time using the very same radio message used for TDoA ranging. In other words, we do not require a separate message to synchronize the clocks nor do we need to establish clock synchrony in advance. This synchronization method utilizes the MAC-layer time stamping of the Flooding Time Synchronization Protocol [12], which eliminates a significant portion of non-determinism in radio communication delays. The maximum clock rate difference between a pair of nodes is on the order of 50 microseconds per second, which translates to maximum ranging error of about 0.15cm for a distance of 30m. We can conclude that time synchronization by itself is not a significant source of error.

Non-deterministic Hardware Delays

Radio waves propagate almost instantaneously over short distances. However, we cannot generally assume that a radio signal arrives at a receiver j (we denote this time t^j_{recv}) as soon as a transmit command is issued at a sender i (at time t^i_{send}). Rather, the arrival time of the first bit of a radio message is determined by non-deterministic non-zero delays at both sender and receiver sides (collectively denoted as δ_{xmit}). Between two nodes that are located near one another, this delay is sometimes larger than the propagation time of an acoustic signal.

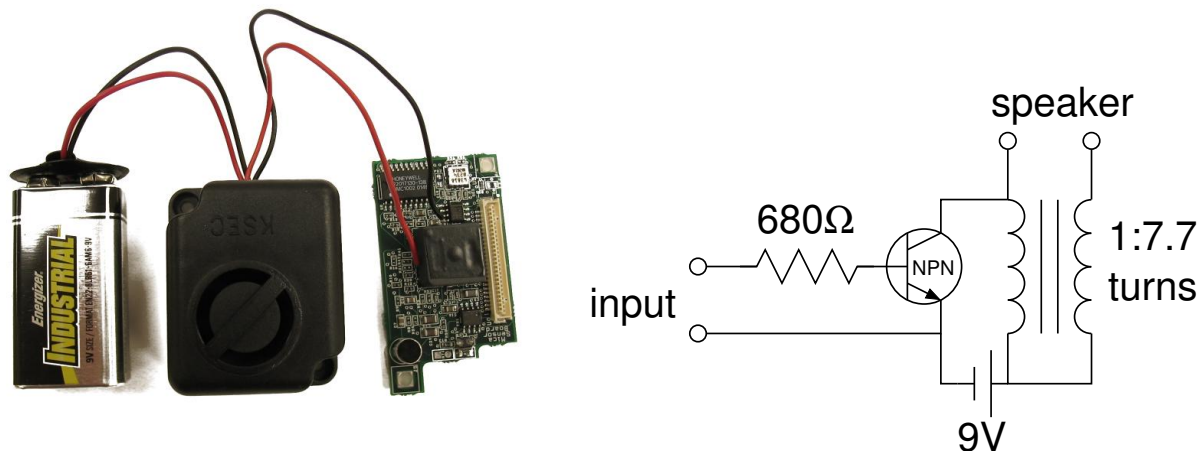
One implication of the delay is that we should not command the buzzer to chirp immediately after sending a radio message. Instead, we introduce a constant delay between transmission of a radio message and the corresponding acoustic chirp. The destination node then takes this delay into account, together with the estimated sensing and actuation delays, when computing the arrival time difference between the radio and acoustic signals. We denote this combined delay δ_{const}. Since the sensing and actuation delays are partially determined by the characteristics of the environment, δ_{const} must be determined through calibration.

Signal Detection

The MICA sensor board comes equipped with a hardware phase-locked loop tone detector, which we use for detecting the chirps. The output of the tone detector 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}. Ideally, this type of detector should be sufficient for acoustic ranging. Unfortunately, our experiments indicate that the output of the tone detector does not correlate with the presence of such a signal with good reliability. A more reliable software detector is necessary.

Computing the Distance

Taking these considerations into account, we compute the distance d_{ij} between source i and destination j as follows. The time difference of arrival can be expressed as $t^j_{detect} - t^j_{send} - \delta_{const}$, where t^j_{send} is the time



(a) Extended sensor board with 9V battery

(b) Circuit diagram

Figure 1: MICA MTS310 sensor board with the loudspeaker extension.

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.

3.2 Hardware Extension

An earlier attempt at acoustic ranging using MICA2 motes and standard MICA sensor boards achieved a measurement range up to 10m on flat, paved surfaces [17]. It relied on a software tone detector which required increased buffer space. The maximum range was limited in large part by the low output signal strength of the sounder, although the small RAM size of the MICA2 mote also limited the theoretical maximum range to about 16m. A more elaborate software tone detector may increase the maximum range in low-attenuation environments; however, in outdoor settings where signal-absorbing obstacles are common, particularly in grassy, uneven and wooded areas, reliable range measurement at longer distances requires an increase in the output power of the acoustic signal.

To increase the maximum measurement range, we have extended the MICA MTS310 sensor board with an inexpensive ($5), off-the-shelf piezo-electric buzzer unit [16] (Figure 1 (a)). The unit provides output power of 105dB; in contrast, the original Ario S14T40A buzzer on MTS310 provides output power of only 88dB, as measured 10cm away from the buzzer. The higher output signal power improves the signal-to-noise ratio (SNR) in high-attenuation environments, allowing receivers 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 and impractical to realize.

The circuit modification, depicted in Figure 1 (b), removes the default siren chip from the commodity speaker unit and connects the inputs to the original buzzer on the MTS310 sensor board. The ground input is connected to the ground terminal of the buzzer, and the positive input to the middle positive terminal. The cavity of the original speaker, which is still powered and used to generate the tone, is closed with electrical tape in order to prevent acoustic feedback.

3.3 Baseline Performance

To empirically evaluate the performance of the baseline ranging service, we deployed a network of 60 MICA2 motes in an urban environment with buildings, pavement, gravel and segments of short grass. Echoes are particularly common in urban environments due to the presence of nearby buildings. Distance measurements

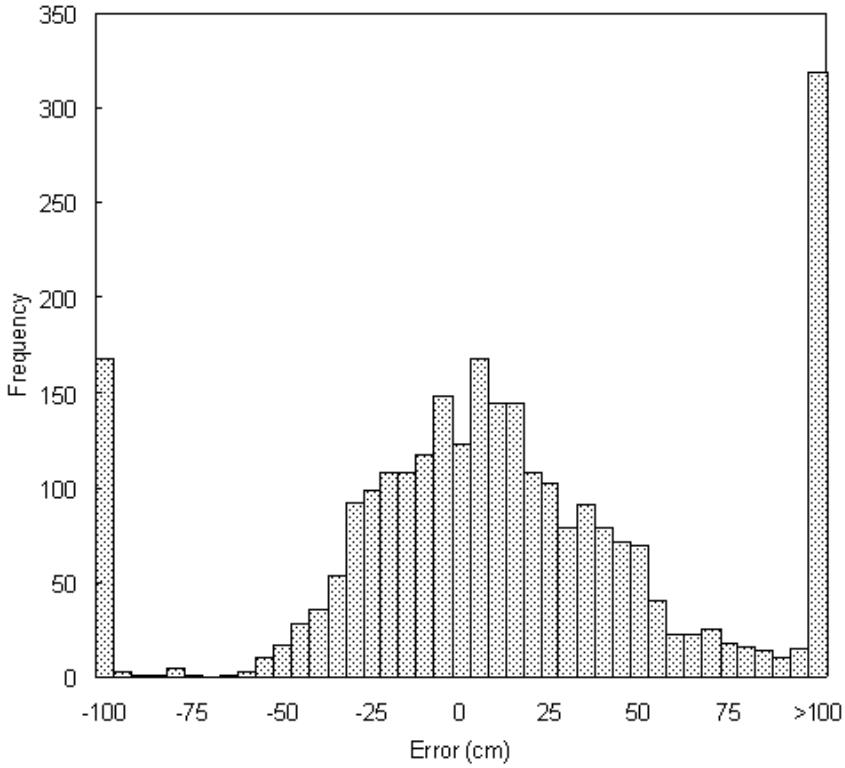


Figure 2: Errors in the distances measured using the baseline acoustic ranging service on a 60-node deployment in the "urban" environment. Inter-node distances up to 30m are recorded.

were made between multiple static sensor locations and a mobile loud speaker device, with distances up to 30m recorded.

Figure 2 shows the errors in the ranging measurements recorded by the nodes. Notice that many of the measurements with >1m errors are underestimates. Making accurate measurements with the TDoA method requires precise detection of the beginning of the chirp. The underestimates were primarily due to a tone detector's picking up noises or echoes from earlier chirps as the acoustic signal. Moreover, we have observed occasional false negatives from the tone detectors which lead to missing the detection of the original signal and picking up later noise or echoes (overestimation). The results clearly indicate that the hardware tone detector alone is not sufficiently reliable for the purpose of determining the beginning of a chirp and thus for estimating distances using TDoA.

3.4 Sources of Error

The rather significant ranging errors encountered by our baseline service came from a number of different sources including the unreliable tone detectors.

1. *Timing effects.* The rate at which the MICA microcontroller samples the tone detector establishes the granularity of ranging measurements. In addition, non-deterministic transmission delays in clock synchronization and radio communication add to the overall magnitude of errors.

2. *Non-deterministic delays in acoustic devices.* It may take some time before an analog sounder reaches its maximum output power level. Likewise, the tone detector may not pick up the signal immediately.

3. *Unit-to-unit variation.* Loudspeakers and microphones vary slightly in output power and sensitivity. As a result, the time for initial detection varies among different speaker-microphone pairs. In extreme

cases, faulty hardware may result in very large errors.

4. *Signal attenuation.* Sensor nodes experience different signal attenuation depending on environmental factors, such as height of grass and presence or absence of trees or bushes, which affect the reliability of signal detection.

5. *Noise.* Noise in frequency bands close to that of the beacon signal frequency as well as wide-band noise may cause false positives.

6. *Echoes.* Due to multi-path effects, some sensors can only hear echoes of the original signal, and not the signal itself. Also, echoes may interfere with the original signal to cancel out the signal or amplify other noises.

7. *Unreliable tone detection.* The tone detector in the MICA sensor board does not always correctly indicate the presence or absence of a signal and thus may trigger false detections.

These sources of error have some distinctive characteristics. We expect errors from sources 1, 2 and 3 follow a mostly Gaussian distribution with a fairly small variance. Errors due to source 4 are likely to be geographically correlated, whereas those from sources 5 and 6 may or may not be correlated, depending on the environment. In the following section, we describe techniques we have incorporated into our refined ranging service to reduce the effects of these errors.

3.5 Refined Approach

Taking into consideration the sources of error listed above, we have improved on the baseline ranging service by adopting a more sophisticated detection mechanism and performing statistical filtering and consistency checking on the results.

Signal Detection

As discussed in Section 3.3, the tone detector device of the MICA sensor board is not very reliable. It sometimes fails to recognize the presence of a signal, particularly at high sampling rates, and other times it returns false positives when no sound other than background noise is present. We have also observed throughout the experiments that the probability of false detection is strongly affected by environmental conditions at the time of measurement (*e.g.*, background noise and signal attenuation). Fortunately, the probability of detecting a tone, $P[b(t) = 1]$, is much higher when a tone is actually present than when only background noise is present. This separation has enabled us to model the output of this tone detector as a binary time series $b(t)$, where $P[b(t) = 1|\text{signal present}] \gg P[b(t) = 1|\text{no signal present}]$.

Based on this model, we improve the confidence of acoustic signal detection by combining multiple sampling results of the tone detector before searching for the beginning of the signal. For our purposes, simple addition suffices, *i.e.*, we add the binary outputs of the tone detector from several ranging attempts in a manner which amplifies tone detections occurring in the same positions in multiple attempts. Since the starting time of a signal at a receiver depends on its distance from the source, we know the starting positions of multiple signals sent between the two nodes are correlated. On the other hand, when random background noise (*e.g.*, birds' chirping, wind noise, footsteps) triggers the detector, the probability that it will be repeated in the same position is very small. Once a sufficient number of sampling results are accumulated, we apply *threshold detection* to make the final judgment: the sum of the binary samples must exceed the threshold value for positive detection to be recorded for that sample, and this must happen for a sufficient number of nearby samples for a chirp to be recognized.

To make signal detection more resistant to echoes and background noise, we encode a pattern in the acoustic signal emitted from the source node. Currently, we use a very simple pattern—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 of the original chirp being detected, we include small random delays, between elements of the pattern. For identifying the tone itself, we apply threshold detection at the receiver and require k positive detections out of n consecutive samples to identify the signal pattern.

We plan to experiment with more complex patterns to examine their effectiveness in reducing the number of incorrect measurements due to echoes and noise. In summary, we compute the detection time series detect(t) as

$$\text{detect}(t) = \text{truefn}\left(\sum_{i=0}^{n} \text{truefn}\left(\sum_{j=0}^{m} b_j(t-i) > T\right) > k\right)$$

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 truefn(x) = 1 if x is true and 0 otherwise. The beginning of the acoustic signal is determined as the minimum value of t that satisfies detect(t) = 1. Figure 3 shows the pseudocode of the signal detection algorithm.

Statistical Filtering

Even with the improved signal detection algorithm, individual range estimates may still be erroneous, albeit less frequently, due to a threshold setting that is too low, hardware malfunction, or some other causes, such as another nearby node chirping out of turn. Assuming that the errors are not correlated, we make multiple distance measurements for a pair of nodes and apply statistical filtering to yield a more stable and accurate estimate of the actual distance. Depending on the number of measurements, 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 outliers than the median, but it needs more measurements to be effective. The statistical filtering is quite effective at discounting uncorrelated errors caused by random, one-time events. Figure 4 shows the performance of the baseline ranging service when employing the median filtering.

Consistency Checking

The ranging service employs 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). These checks involve comparison of data between multiple nodes. For example, bidirectional range estimates between a pair of nodes are discarded if they are inconsistent. If three nodes have measurements to each other, we use the triangle inequality [2] to identify inconsistent one. A caveat is none of these can identify which of the measurements is incorrect with complete certainty. Sometimes it may be beneficial to retain suspicious measurements due to the scarcity of available data.

3.5.1 Deployment Constraints

Some sensor network deployments offer additional information about sensor placement. For example, a deployment may have a requirement of minimum node separation. Rough distance estimates can be made based on node density and network hop count before the ranging service starts. On a regular grid deployment, a set of possible inter-node distances can be deduced from the size and shape of the grid configuration. These data provide additional constraints that consistent ranging measurements should satisfy. We plan to incorporate more advanced filtering and consistency checking based on these constraints in the ranging service.

3.6 Experimental Evaluation

We evaluated the performance of the refined ranging service on a network of 46 MICA2 motes with the extended MICA sensor boards. Experiments were performed in a flat, grassy area near an airport, with occasional loud aircraft engine noise present. The size of the area was 70m by 70m and grass height varied from 10 to 15cm. This type of environment tends to absorb sound more, reducing the maximum measurable range and the probability of detection of the signal in comparison to the urban environment discussed in Section 3.3. Additional experiments were performed as well, with 5-15 sensors on a paved parking lot and a

[2] The estimates of two sides of the triangle add up to less than the third.

```
record-signal(samples[])
samples = 0
for i = 1 to # of chirps
wait for sync message
for j = 1 to # of samples
if tone detected
samples[j] = samples[j] + 1
```

```
detect-signal(samples[],k,m,T)
count = 0
for i = 1 to m
if samples[i] ≥ T
count = count + 1
for i = m to # of samples
if samples[i - m] ≥ T
count = count - 1
if samples[i] ≥ T
count = count + 1
if count ≥ k and samples[i-m+1] ≥ T
return i - m;
return -1
```

Figure 3: The signal recording and detection routines for acoustic ranging. **detect-signal** returns the index of the first successful signal detection. T is the threshold value, and at least k out of m consecutive detections are needed for positive signal identification.

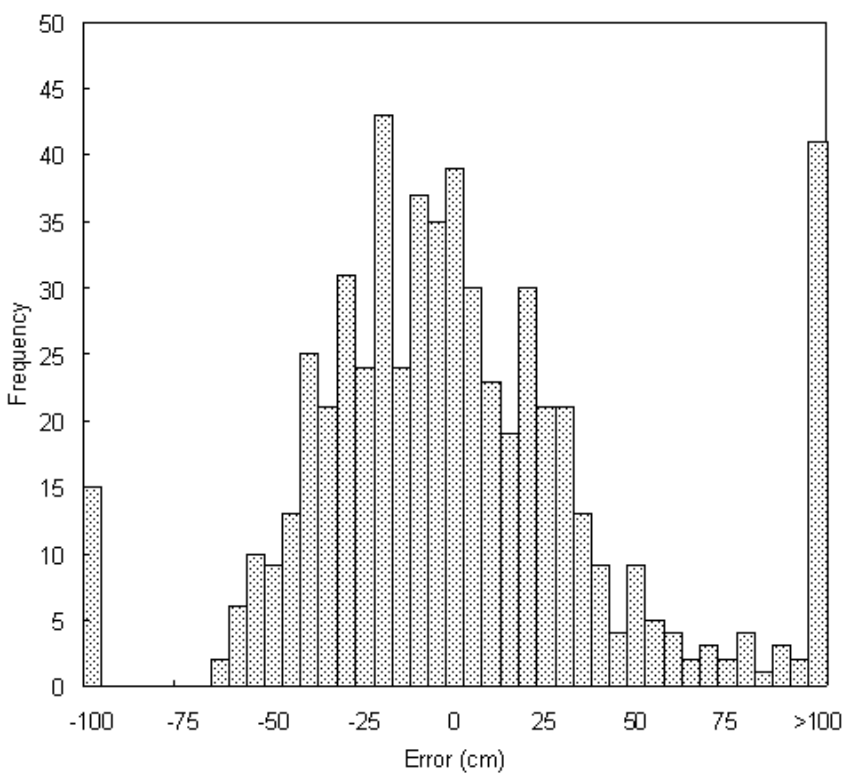


Figure 4: Ranging measurements in the urban environment from the baseline method with median filtering of up to five measurements.

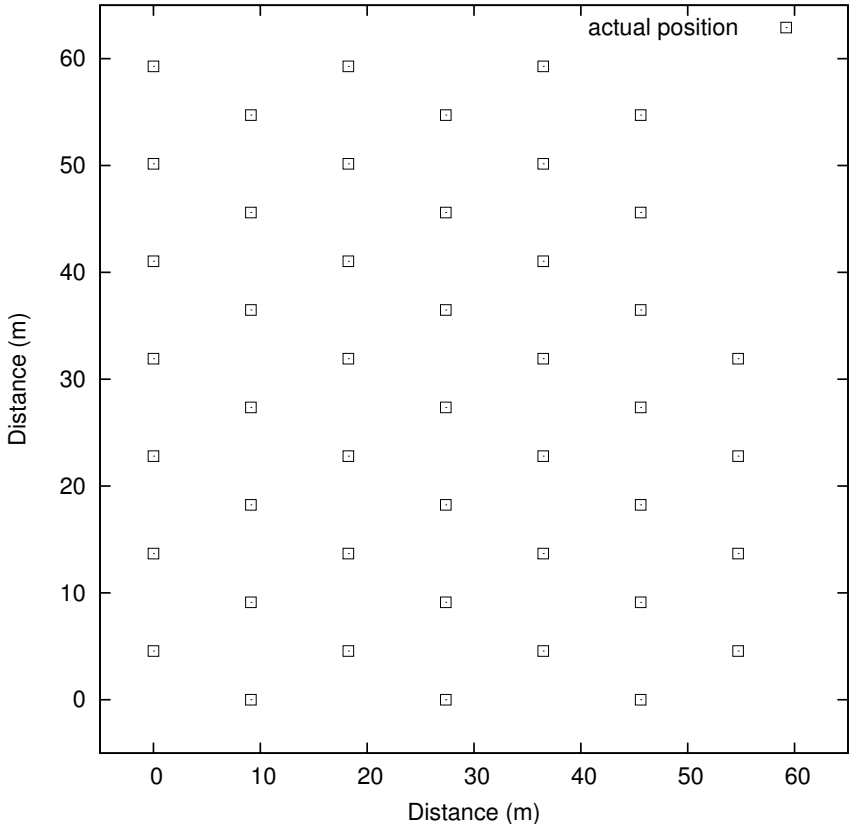


Figure 5: An offset grid deployment pattern with 9m and 10m node spacing.

wooded area with >20cm tall grass and scattered trees. However, the data from these experiments are too scarce to report here in detail.

The sensors were placed in a 7×7 offset grid pattern with 9m and 10m grid spacing between the nearest neighbors (see Figure 5). However, neither the grid pattern nor the spacing constraints were used for filtering the ranging results in the following experiments. A constant 4.3kHz tone is emitted by the buzzer in 8ms chirps. Originally, we used longer chirps (64ms), but this only resulted in a large number of over-estimates. That is because a long chirp has more chances of its later part being detected when its early part is missed. Reducing the length of the chirp removed a large number of overestimation errors due to the late detection. For 8ms chirps, the maximum error due to the over-estimation was approximately 3m. Setting the chirp length lower than 8ms resulted in fewer correct positive detections and shorter range, most likely because the speaker did not have enough time to fully power up. The acoustic signal detector routine sampled the hardware tone detector at 16kHz and accumulated binary detection values of the same offset in multiple chirps in a single buffer.

In threshold detection, a high threshold is advantageous in noisy environments to limit false positives. On the other hand, a low threshold is more appropriate in quieter settings as it reduces false negatives. Prior to the experiments, we calibrated the ranging service in the target environment for best performance and determined the appropriate threshold parameters. For a detection to be recorded, the sum of the binary tone detection outputs from the 10 chirps must exceed the threshold value of 2 for in least 6 of 32 consecutive samples. Such low thresholds are appropriate in high attenuation environments, since they lead to correct detections even of weak signals; however, this also makes the ranging service more vulnerable to false positives.

We performed additional calibration for the offset compensating for the constant delay incurred in sensing and actuation. Since environmental characteristics affect the sensitivity of detection, e.g., the noise floor and signal attenuation, it is best to calibrate the ranging service in an environment similar to the one intended

for deployment. Without such calibration, a constant offset of 10–20cm may be added to every ranging measurement.

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 are representative of the behavior of the ranging service in realistic deployments.

3.6.1 Analysis: Accuracy

Figure 6 shows the error distribution for three rounds of measurements, with each sensor node emitting one sequence of 10 chirps per round. We identify several distinct features of the error distribution from the figure. There is an approximately zero-mean, bell-shaped component of the error in the range of ±30 cm. This component is most likely due to timing effects, hardware delays and unit-to-unit variation (refer to Section 3.4). The fact that the error distribution is virtually zero-mean suggests increasing the number of samples and taking the median or mode would be a logical way to improve confidence in distance estimates.

We also see several large-magnitude errors in the data outside the ±1m range, some as large as 11m. The causes of the large-magnitude errors can be attributed to non-recurring ambient noise and faulty hardware. It is not likely that such errors are correlated across pairs of nodes. Fortunately, most of these errors are eliminated with the bidirectional consistency check. The error distribution after the filtering is shown in Figure 7. Errors of smaller magnitude, but outside the ±30cm range, are present in bidirectional as well as unidirectional measurements. These errors cluster to the right (*i.e.*, over-estimation 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) caused more misses at the early part of the signal.

For a ranging service to be effective, it should be able to guarantee distance estimates with bounded-magnitude errors. To study how the probability of encountering ranging errors, particularly large-magnitude errors, is related to the distance between two nodes, we plot the ideal, measured, and filtered measurements versus the actual distances (Figure 8). The figure shows that large-magnitude errors occur more frequently when measuring over a longer distance. A couple of factors contribute to this effect. In high-attenuation environments, signal power drops significantly at longer distances. Considering that the background noise is independent of inter-node distance, this results in SNR deterioration and increased false detections. Also, it takes longer for a sound signal to reach a farther node. Therefore, there is a longer window of time during which a farther node can make a false detection. Assuming a constant probability of getting a background noise-induced false positive, a more distant node would experience a higher error rate. We expect that encoding a more sophisticated pattern in the ranging service would alleviate the problem by reducing the chance of false detections.

3.6.2 Analysis: Maximum Range

We have run the ranging service with the lowest detection threshold (*i.e.*, 1) in a quiet environment to determine its maximum detection range on different deployment. In our experiments on an area with 10–15cm tall grass, virtually no detections occurred beyond 20m. On pavement, with other conditions being similar, virtually every chirp was detected up to a range of 35m, and some were detected at as far as 50m away. Interestingly, for higher threshold values which one should use to achieve accurate distance estimates in noisy environments, the maximum range did not drop significantly (15–20m on grass and 30m on pavement). The range at which most chirps were consistently detected (about 80–85%) was 10 and 25m on grass and pavement, respectively. This effect is most likely due to the increase in SNR induced by accumulating multiple measurements.

It should be noted, however, that we have observed that some speaker-microphone pairs have ranges that are consistently much shorter or much longer than the typical values presented above. We suspect this happens due to unit-to-unit variations in loudspeaker power output and microphone sensitivity. The microphones are rated at ±3dB sensitivity, and we have observed variations of up to 5dB on the loudspeakers.

The theoretical maximum range of the service is closed related to the buffer space available in the underlying WSN platform. The buffer space required for our service is determined by the desired maximum measurable distance and the number of samples accumulated per measurement. We allocate 4 bits for each offset of the buffer, which allows samples from up to 15 chirps to be accumulated. For 15 samples at distances

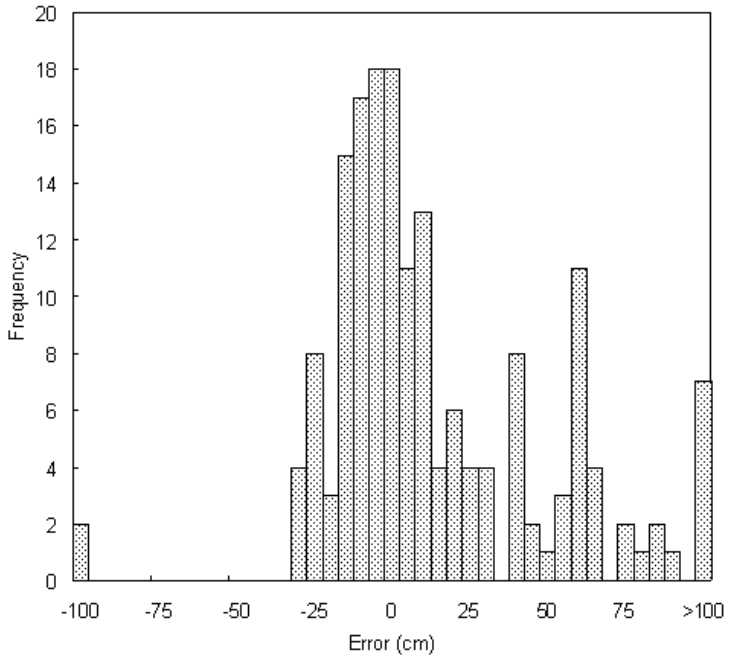


Figure 6: Ranging error histogram after six rounds of ranging measurements, 46-node deployment on grass, distances up to 20m.

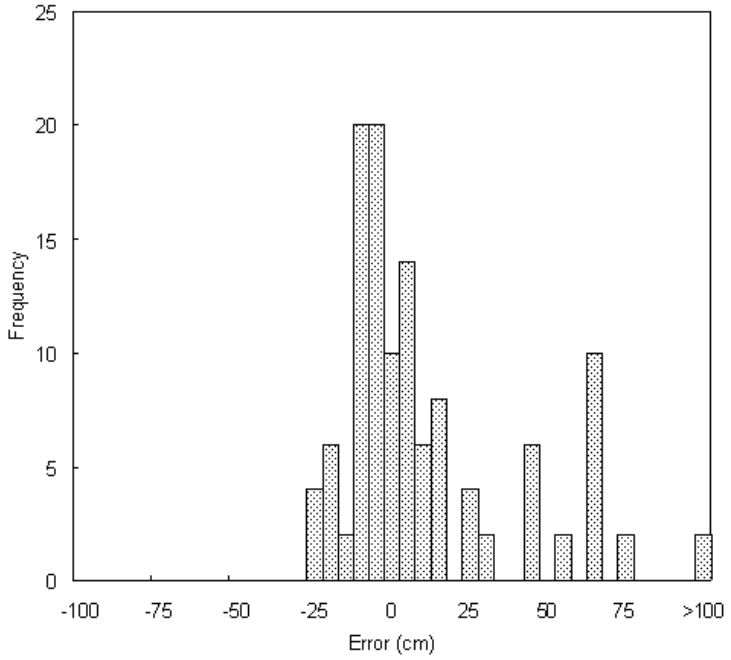


Figure 7: Ranging error histogram for node pairs with bidirectional measurements only.

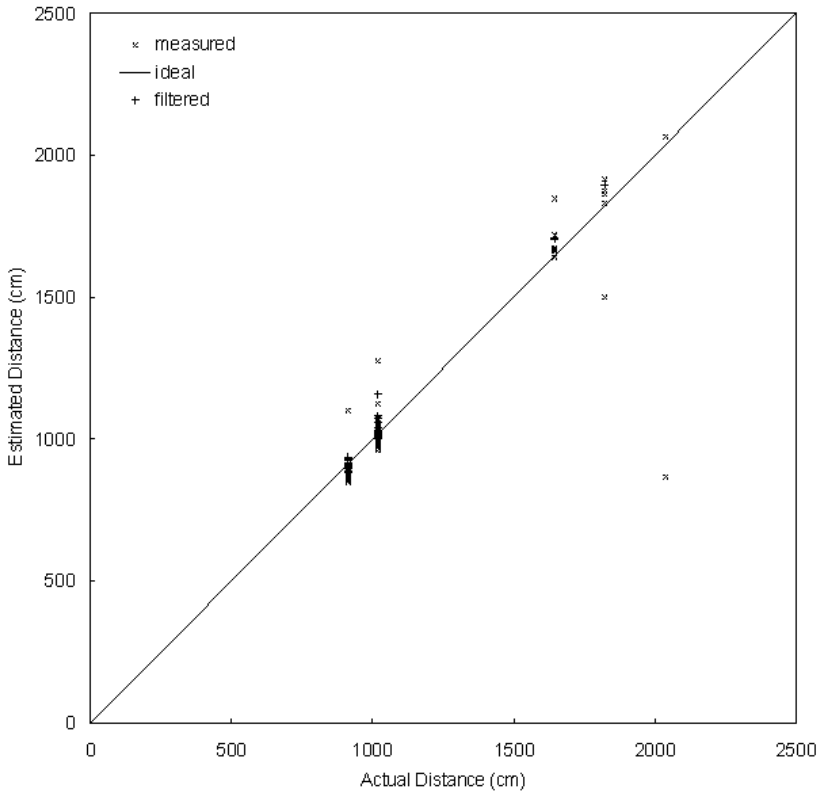


Figure 8: Plot of ideal, measured, and filtered acoustic ranging measurements versus actual distances on a grassy field. Large-magnitude errors are more common at longer distances.

up to 20m, the service uses less than 500 bytes of RAM. It uses the tone detector of the MICA sensor board to reduce the memory footprint at the expense of reliability and accuracy. We offset the loss of accuracy using signal processing and statistical filtering techniques. This is a significant improvement over the ranging service described in [17] which fills all available buffer space in the MICA2 mote platform only to achieve a maximum range of less than 16m. Not only do we extend the maximum measurable range, but also leave enough RAM available for other applications to run concurrently with the ranging service. To the best of our knowledge, this is the first fully-functional ranging service for wireless sensor networks which features long range, high precision, and a relatively small memory footprint.

3.7 Alternative Tone Detection

The acoustic ranging solution described above assumes availability of a fixed-frequency buzzer and a matching hardware tone detector in the WSN platform. However, not all wireless sensor platforms come equipped with the devices. Instead, some provide more general acoustic sensing and actuation features, such as a pulse width modulation (PWM) buzzer and tunable band-pass filter for the microphone (*e.g.*, Crossbow's XSM mote). Emulating MICA's acoustic features on these platforms (*i.e.*, tuning the buzzer and filters to a fixed frequency and performing simple energy detection) may not be sufficient due to the low signal-to-noise ratio when making ranging measurements over long distances. For example, we have found from preliminary tests on the XSM platform that a narrow setting of its hardware bandpass filter combined with energy detection achieves similar accuracy as the MICA hardware tone detector, but a shorter maximum range (10m).

To support the acoustic ranging solution on platforms where a hardware tone detector is not available,

we have designed a digital signal processing filter[3] based on the Discrete Fourier Transform (DFT)[4] [4]. While the standard DFT algorithm calculates the amplitude of each frequency component of a signal, for tone detection purposes we only need to know the amplitude of the beacon signal's frequency component. In order to reduce changes of false positive detection, it is useful to automatically isolate the amplitude of noise and subtract it from the DFT output; a positive result indicates detection of a tone. We evaluate DFT for all frequency components and average the results to calculate this amplitude. The pseudo code of the tone detection algorithm is in Figure 9. Figure 10 depicts the tone detection filter's outputs for a clean and a noisy signal containing periodic constant-frequency chirps. In the latter case, three of the four chirps are correctly detected, with no false positives.

Because the ranging service with the software tone detector needs to store a sum of raw sampled values rather than a sum of 1-bit output values from a hardware tone detector, its memory requirement is considerably larger than that of its hardware counterpart. To achieve a maximum range of 20m, a 2kB buffer is required with a sampling rate of 16kHz. Depending on the environment and the characteristics of the acoustic signal, a slower sampling rate or a smaller number of bits recorded per sample may increase the maximum range when given a fixed amount of buffer space.

4 Resilient Localization for Wireless Sensor Networks

In this section we discuss a suite of localization schemes and their experimental evaluation. We start off the discussion by presenting a localization algorithm using multilateration, by far the most popular scheme (Section 2). Localization results with the technique demonstrate sub-par performance when only a small number of measurements are available or distance measurements contain errors of large magnitude. In response to these limitations, we develop a localization scheme based on least squares scaling (LSS) [11] with soft constraints, a multidimensional scaling (MDS) technique. First, we describe a centralized version and demonstrate its resilience against measurement errors and missing range measurements. We then extend it to a distributed version, to provide a scalable alternative for use in large-scale deployments. In figures which show localization results, unless noted otherwise, circles represent anchor nodes, and 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. Squares without a connected cross are non-anchor nodes that are not localized.

4.1 Multilateration Localization

In order to localize a node, multilateration (or trilateration) requires distance measurements to at least three non-collinear nodes which know their own locations (called *anchors*). Draw an imaginary circle at each anchor a of radius d_a, and the circles should pass through the node being localized; this is the basis of trilateration. In theory, o find the location of a node in the plane, three anchors should suffice. In practice, however, errors in distance measurements keep the range circles at the anchors from intersecting at a single point. As such, the system of second-order equations determined by all measurements generally does not have an exact solution. A more robust approach is to estimate positions through error minimization using as many anchors as available (hence, multilateration). One such scheme is based on least squares estimation.

4.1.1 Algorithm

Let A be a set of anchor nodes to which a node has distance measurements. The position of the node (x, y) is estimated as

$$\operatorname{argmin}\left(\sum_{a\in A} w(c_a)\cdot\left(\sqrt{(x-x_a)^2+(y-y_a)^2}-d_a\right)^2\right)$$

where (x_a, y_a) is anchor a's position, d_a is the measured distance to a, and $w(c_a)$ is the weight based on the degree of confidence in a's position and the quality of d_a.

[3]We hasten to note that detection using the filter has not been fully tested and more experiments are needed to empirically assess the filter's performance and fine-tune this approach.

[4]In practice, FFT would be used instead of simple DFT.

```
init():
  samples[36] = 0
  n = k = re_4 = im_4 = re_6 = im_6 = 0

filter(sample):
  sample -= samples[n], samples[n] += sample
  cases of n (mod 4):     cases of k:
    0 → re_4 += sample     0 → re_6 += 2·sample
    1 → im_4 += sample     1 → re_6 += sample, im_6 += sample
    2 → re_4 -= sample     2 → re_6 -= sample, im_6 += sample
    3 → im_4 -= sample     3 → re_6 -= 2·sample
                           4 → re_6 -= sample, im_6 -= sample
                           5 → re_6 += sample, im_6 -= sample
  n = (n+1) (mod 36), k = (k+1) (mod 6)
  return [(re_4^2 + im_4^2), (re_6^2 + 3·im_6^2)/2]
```

Figure 9: The XSM signal detection routine. Two different beacon frequency bands around 1/4 and 1/6 of the sampling rate are amplified. These frequencies are chosen to minimize the need for numerical calculations when multiplying the samples by the complex roots of unity in the DFT coefficient calculation. Tone detection may compare the amplitude of these frequency bands against each other or a constant threshold.

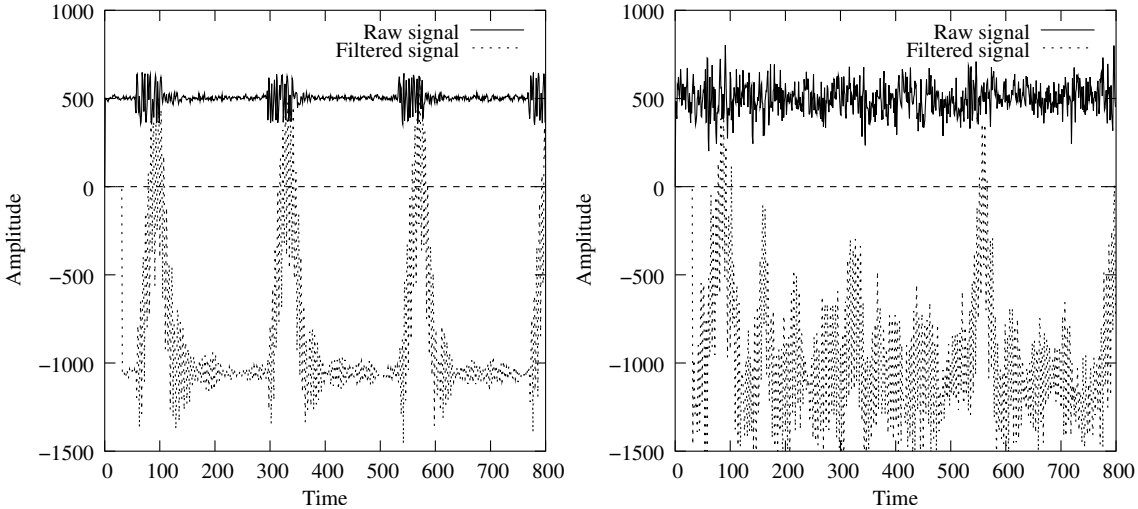


Figure 10: Clean (left) and noisy (right) signals before and after applying the tone detection filter.

With a slight modification we can devise a scheme which progressively localizes non-anchors so as to increase the number of anchors used per non-anchor. Non-anchors with distance measurements to sufficiently many anchors are localized first. Once localized, they become anchor nodes and are used to localize the remaining non-anchors. In this progressive localization, different anchors may be assigned different weights depending on the quality of range measurements (d_a) as well as on associated localization errors (E) if they were originally non-anchors and subsequently localized. For the experiments reported below, we did not use this proposed modification; rather, we used the original set of anchors only and assigned a constant weight of 1 to all anchors.

4.1.2 Intersection Consistency Checking

Like other least squares estimation techniques, multilateration via least squares is sensitive to noisy measurements. We improve resistance to measurement errors by cross-checking consistency of distance measurements and throwing out inconsistent data.

The consistency check is based on the following simple observation. Recall that errors in distance measurements to anchors would not allow the corresponding range circles to intersect at a single point. Rather, some intersection points of the circles would form a cluster around the node being localized. For an anchor with a sufficiently accurate distance measurement, its range circle should have intersection points close to other consistent anchors' intersection points.

Exploiting this observation, we compute intersection points of all pairs of circles and drop from consideration those anchors which have no intersection points close to other intersection points (*e.g.*, beyond 1m range). The remaining anchor nodes, *i.e.*, those with consistent distance estimates, are used for localization. Note that we may take the mode of the intersection points of the remaining anchors instead of minimizing the error if the number of anchors is large enough.

Figure 11 illustrates the intersection consistency check. The circles represent measured distances from anchor nodes (the small circles) to a non-anchor node (the square). The range circles are centered at the anchor positions and each has the radius of the measured distance to the non-anchor node. Consider anchors (-170, 700) and (950, 600). The two nodes are located almost collinear with respect to the non-anchor node. Thus, even a small amount of error in either of the measured distances would cause large displacement of intersection points. Since the localization algorithm is based on error minimization, such a large displacement results in a large localization error. The intersection consistency check filters these anchors out. For example, anchor (-170, 700) has no intersection points near the cluster formed around the non-anchor node and hence is discarded.

4.1.3 Experimental Evaluation

We performed a small scale experiment to evaluate the performance of multilateration localization in real deployments. The experiment used 15 MICA2 nodes (of which 5 are anchors) with the MICA sensor boards in a $25\times25\text{m}^2$ parking lot. The acoustic ranging service described in Section 3 was used to measure distances between nodes. At the time of the experiment, only 5 MICA sensor boards were fitted with loudspeakers. All of these were used for anchor nodes, resulting in one-way measurement data. The median operation was used as a statistical filter to discard outliers. This experiment was performed before we had incorporated the sound pattern into the ranging service. As a result, individual range measurements carried larger error magnitudes.

Figure 12 shows the localization results. The average localization error (*i.e.*, the average of the distances between actual node positions and the corresponding estimated positions) is 0.9m. Although this result seems reasonable, the small deployment size renders the judgment of multilateration-based localization inconclusive. Furthermore, the large percentage of anchors is disheartening in terms of scalability.

In order to confirm the performance, we conducted another experiment with a medium-scale deployment. For this experiment, we randomly chose 13 nodes as anchors from a total of 46 nodes deployed on the regular grid shown in Figure 5, and localized the remaining 33 non-anchors. Figure 14 shows the experimental results. To our disappointment, the results were dismal: only 7 nodes (20%) were localized, because the others did not have distance measurements to even a minimum of three anchors (Figure 13). The average number of anchors per node was 1.47. The 7 localized nodes had an average localization error of 0.7m.

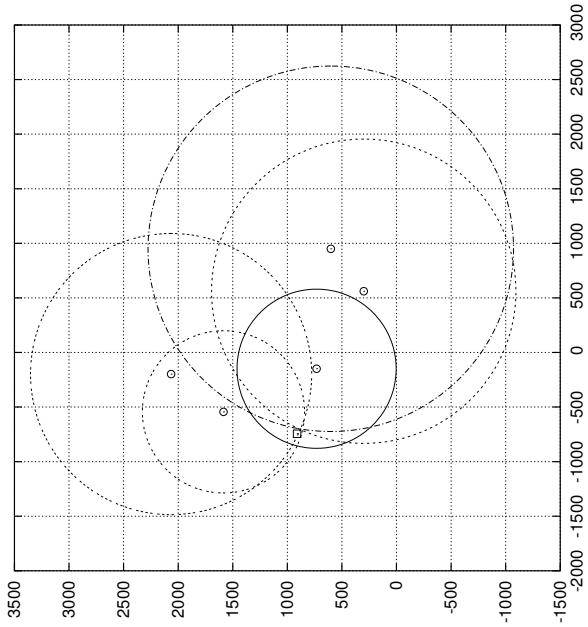


Figure 11: Distance measurements to collinear anchors.

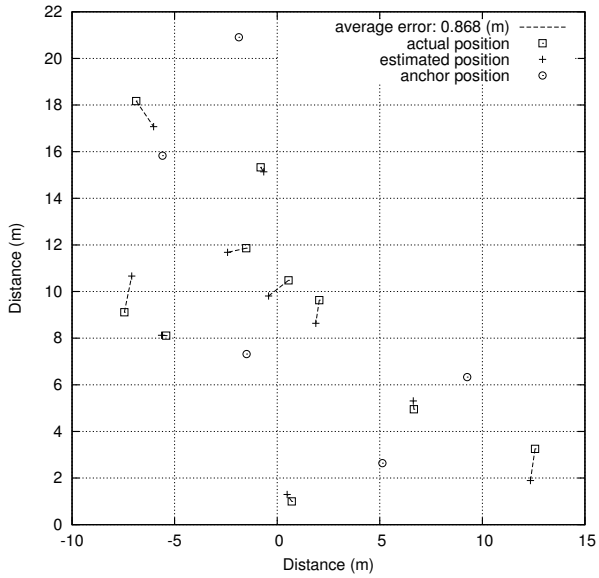


Figure 12: Multilateration localization results with 15 nodes (5 anchors) in a 25×25m area.

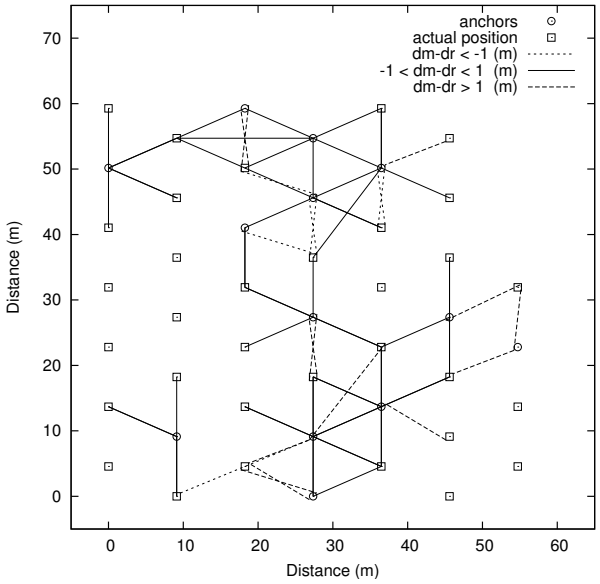


Figure 13: Distance measurements used for the experiment in Figure 14. The node at (0, 4.5) failed to report its existence. Consequently, it does not appear in any localization results.

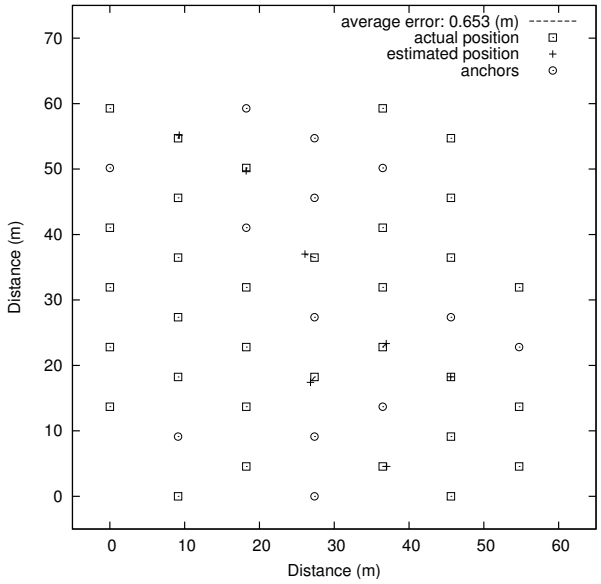


Figure 14: Multilateration localization results with actual distance measurements from the grid sensor deployment in Figure 5. The boxes with no corresponding cross are nodes that were not localized.

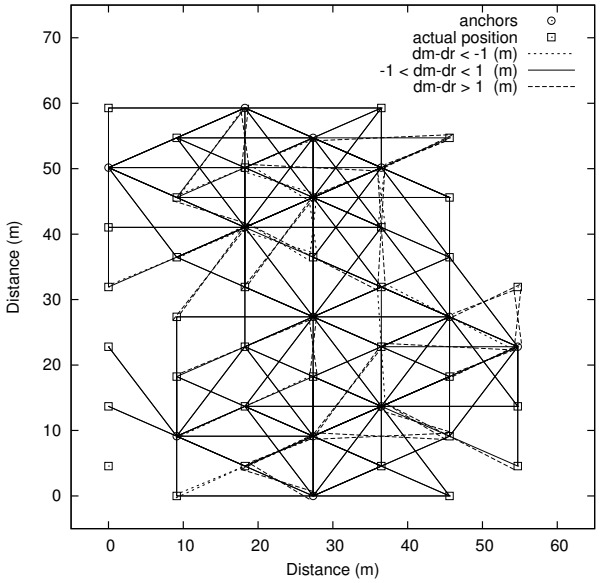


Figure 15: Distance measurements extended with "simulated" distances. Distances were generated by adding noise from $\mathrm{N}(\mu = 0; \sigma = 0.33\mathrm{m})$ to actual distances calculated from the grid geometry.

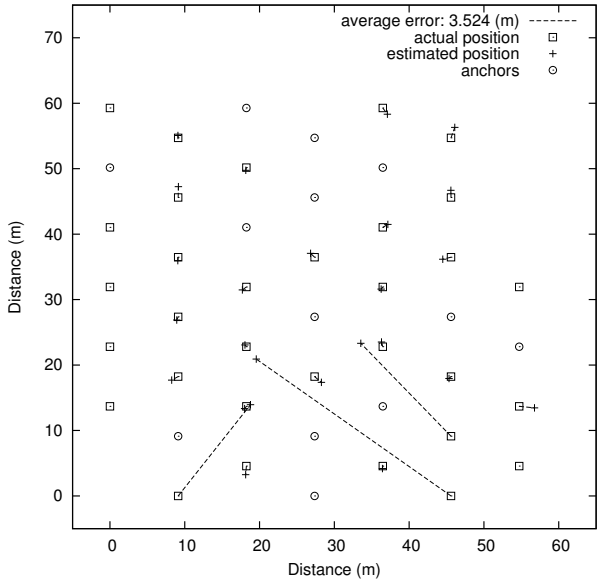


Figure 16: Multilateration localization results with the extended distance measurements in Figure 15.

To see how multilateration-based localization performs when distance measurements to sufficiently many anchors are available, we added simulated distances to the set of actual measurements, shown in Figure 13, and repeated the localization process. Each pair of nodes which did not have a distance measured by the ranging service was assigned a distance generated by adding a random noise term drawn from $N(\mu = 0; \sigma = 0.33\text{m})$ to the actual distance calculated from the known grid geometry. Based on experimental results, we assumed the ranging service had a maximum measurable range of 22m, so no distances are assigned to node pairs separated by more than 22m. This augmentation increased the average number of anchors per node to 3.84.

Figure 16 shows the localization results. As expected, we see a drastic improvement in the percentage of localized nodes; about 80% of the nodes are localized. The average position error for the 28 localized nodes is 3.5m, most of which is contributed by three incorrectly localized nodes near the bottom of the graph. Of those nodes, the leftmost one is due to an underestimated distance measurement,[5] while the other two are victims of the gradient descent falling into a local minimum (Figure 15). Without these three nodes, the error improves to 0.9m. Notice that the nodes that are not localized appear on the periphery of the area. In particular, the entire leftmost column of nodes are not localized, except for an anchor node. This can be attributed to the lack of anchors on the boundary of the grid, especially in the far left section.

4.1.4 Multilateration with Low Anchor Density

The multilateration-based localization algorithm works best when a network has the following properties.

1. High node density
2. High ratio of anchor to non-anchor nodes
3. Uniform anchor distribution

Unfortunately, these are difficult to attain in real WSN deployments. When a network has too low a node density or has only a few anchors, no ranging service can generate distance measurements to a sufficient number of anchors. Lack of relevant distance measurements is the most common cause of break-down of the naive multilateration localization technique (Figure 14). Our experimental results suggest that we need a more resilient localization method that can withstand a sparse collection of measurements with large errors, as found in sensor networks deployed in real environments.

4.2 Centralized LSS Localization

In what follows we develop a resilient localization service based on least squares scaling (LSS) with soft constraints. We first present a centralized version which requires all measurement data be collected at a node. Then we extend it to a distributed version in Section 4.3. What favors LSS over classical MDS for localization of sensor nodes is that LSS-based localization does not require distance measurements between all pairs of nodes. Even with some missing distance measurements, it still yields acceptable localization results. Moreover, we may use different weights for distance measurements depending on their confidence levels and easily incorporate constraints such as minimum node separation into the minimization process to further improve localization results.

4.2.1 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" [11]. Here, a configuration refers to a set of coordinate values. When distances between nodes are available, MDS [19, 18, 11] 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, singular value decomposition (SVD) is applied to the quadratic matrix to get 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*.

[5] Intersection consistency checking was omitted in this localization simulation.

An alternative technique is LSS [11], 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:

$$E_u = \sum_{d_{ij} \in D_{\text{full}}} \left(\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} - d_{ij} \right)^2$$

where d_{ij} is the 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:

$$E_w = \sum_{d_{ij} \in D} w_{ij} \cdot \left(\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} - d_{ij} \right)^2$$

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. LSS allows us to incorporate this minimum spacing constraint into localization as a *soft constraint* [7]. 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:

$$E = E_w + \sum_{d_{ij} \notin D} w_D \cdot \left(min \left(\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}, d_{min} \right) - d_{min} \right)^2$$

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 update coordinates of the nodes at each time step using the rules

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

where α is a step size and $\nabla E = [\frac{\partial E}{\partial x_1}, \ldots, \frac{\partial E}{\partial x_n}, \frac{\partial E}{\partial y_1}, \ldots, \frac{\partial E}{\partial y_n}]$ is the gradient of E. Without the soft constraint,

$$\left.\frac{\partial E}{\partial x_i}\right|_{[\mathbf{x}^t, \mathbf{y}^t]} = \left.\frac{\partial E_{unconstrained}}{\partial x_i}\right|_{[\mathbf{x}^t, \mathbf{y}^t]} = w_{ij} \cdot \sum_{d_{ij} \in D} (d_{ij}^{comp} - d_{ij}) \cdot (x_i^t - x_j^t)/d_{ij}^{comp}$$

where $d_{ij}^{comp} = \sqrt{(x_i^t - x_j^t)^2 + (y_i^t - y_j^t)^2}$. With the soft constraint, if d_{ij} is not defined and $d_{ij}^{comp} < d_{min}$,

$$\left.\frac{\partial E}{\partial x_i}\right|_{[\mathbf{x}^t, \mathbf{y}^t]} = \left.\frac{\partial E_{unconstrained}}{\partial x_i}\right|_{[\mathbf{x}^t, \mathbf{y}^t]} + w_D \cdot \sum_{d_{ij} \notin D} (d_{ij}^{comp} - d_{min}) \cdot (x_i^t - x_j^t)/d_{ij}^{comp}.$$

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

To avoid local minima, the gradient descent starts each round of minimization with seed positions obtained by perturbing the best results so far. This process is repeated until a reasonable minimum is reached or the maximum computation time limit expires.

4.2.2 Experimental Evaluation

To evaluate the resilience of centralized LSS localization, we performed an experiment with 47 MICA2 motes equipped with standard MICA sensor boards and our custom loudspeaker devices in a $64 \times 64m^2$ grassy area, using the grid layout shown in Figure 5. The minimum inter-node spacing of 9.14m in the grid layout was used as a soft constraint, with $w_{ij} = 1$ and $w_D = 10$.

Figure 18 compares actual and estimated positions of the nodes. The computed coordinates were translated, rotated and flipped to achieve a best-fit match with the actual node coordinates. Most of the errors are found in the bottom left quadrant of the plot; the overall average localization error is 2.2m. Without the largest 5 errors, the average improves to 1.5m. As seen in Figure 17, lack of measured distances causes the two nodes in the (0:10,10:20) area to be flipped. We verified that adding a reasonable measured distance between nodes (9, 18) and (18, 13.5) (we used 10.2m) corrected the flipping. The lower three nodes are rotated because of large underestimation (*i.e.,* $<$ 1m) of the distance between nodes (9,0) and (18,9) and large overestimation of the distance between nodes (18,4.5) and (27,0). Lack of measurements between the second and third column creates a hole, attracting the computed coordinates of the nodes in the upper left and lower right quadrant, while the concentration of overestimated distances in the middle pushes the computed coordinates of the nodes in the upper right quadrant outward.

To examine the effectiveness of the minimum distance soft constraint, we conducted the same localization experiment without the soft constraint. Figure 19 shows localization results from the experiment. The estimated positions did not converge anywhere near the actual positions, even after running the minimization for many hours. This demonstrates the importance of the soft constraint.

We further performed simulations to compare multilateration to LSS localization using a random node deployment. We selected 59 plausible node positions in a map of a few city blocks in a small town and randomly designated 18 of them as anchors. Then, we selected 945 pairs of nodes whose Euclidean distances were less than 22m and perturbed the distances with errors from a Gaussian distribution $N(\mu = 0; \sigma = 0.33\text{m})$. For LSS localization, we ignored the 18 anchors and instead regarded them as non-anchors.

The results were as follows: 35 nodes were localized with average localization error of 1.0m. Figures 21 and 22 show the results of the LSS localization with and without the minimum distance soft constraint, respectively. For the former, we penalized pairs of nodes with unknown distance when they were assigned coordinates which made them closer than 9m. All the nodes were localized with average error of 0.5m. The results are much better than multilateration, considering that some nodes were not localized at all with mulitlateration, while no anchors were used with LSS localization. Without the soft constraint, most of the nodes in the lower half were not properly localized (Figure 22).

Figure 23 shows how use of the soft constraint helps with error minimization. The error function has more terms with the soft constraint than without, which are all positive (because they are squared terms). Thus, we know that the the error function without constraint terms should have a smaller global minimum. As is evident from the figure, the soft constraint greatly reduces the time to reach a global minimum.

4.3 Distributed Localization

The centralized LSS localization algorithm is resilient against sparse distance measurements and large measurement errors. Unfortunately, it does not scale well as network deployments grow in size. As more nodes are added, the number of terms in the error function increases, as does the number of local minima the computation may fall into. In this section, we extend the centralized LSS localization algorithm to a more scalable distributed version.

4.3.1 Algorithm

The distributed version consists of three steps: local localization, calculating a transform between the local coordinate systems for each pair of neighboring nodes, and alignment of local coordinate systems to a global coordinate system.

Step 1. Local Localization Each node collects distance measurements to its neighbors as well as amongst them. A node's neighbors are nodes to which it has direct distance measurements from the ranging service.

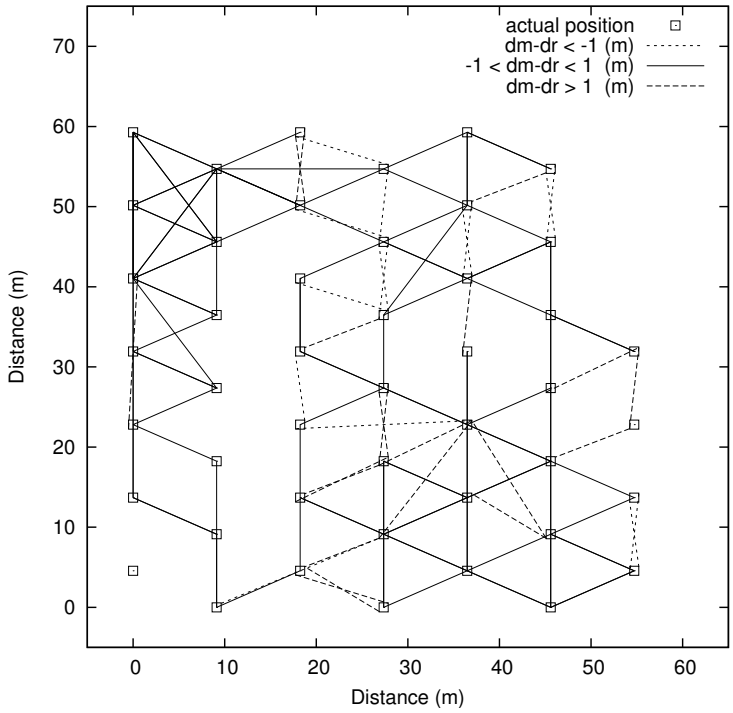


Figure 17: Distance measurements used in the LSS localization experiments are the same as the ones plotted in Figure 13, except that the latter shows the distances to the anchors only.

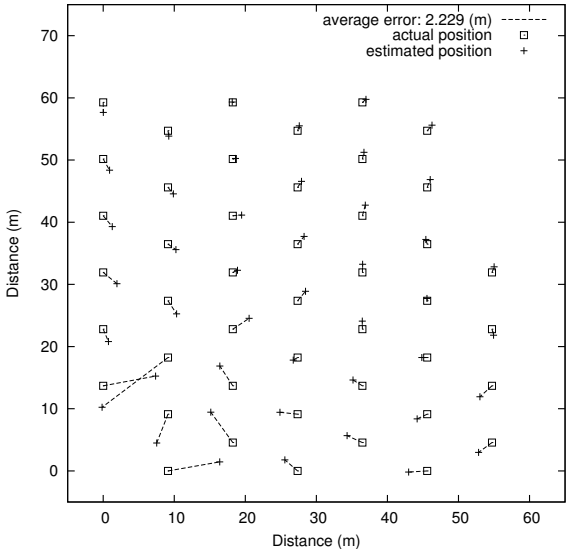


Figure 18: Centralized LSS localization results with the minimum spacing constraint. The average localization error is 2.2m.

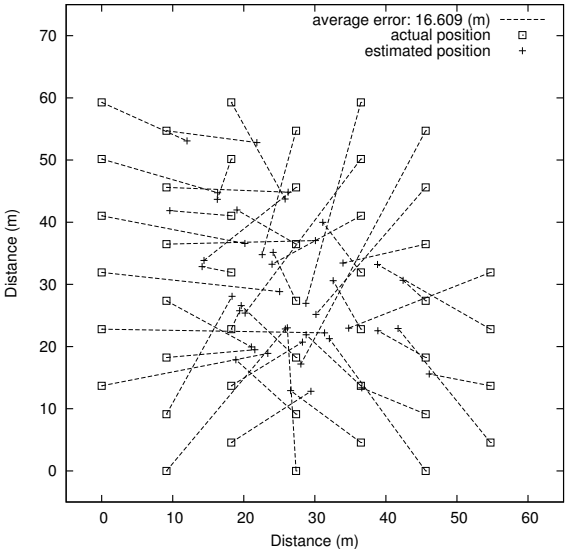


Figure 19: Centralized LSS localization results without minimum inter-node distance constraint after a full day of minimization. The computed coordinates failed to converge to the corresponding actual coordinates.

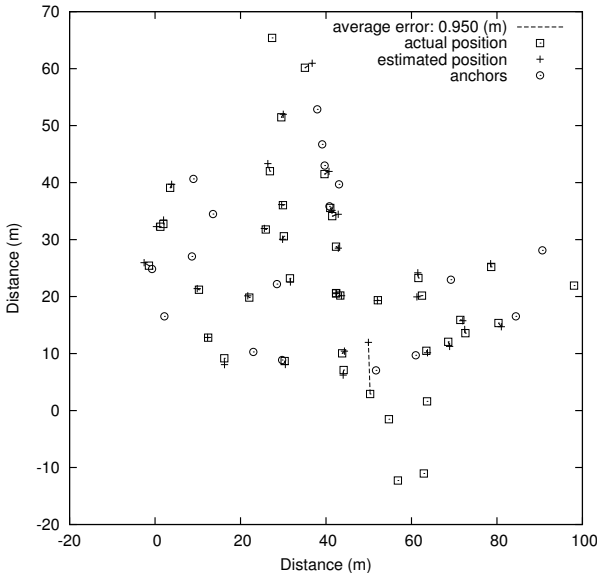


Figure 20: Multilateration localization simulation results with a random sensor deployment.

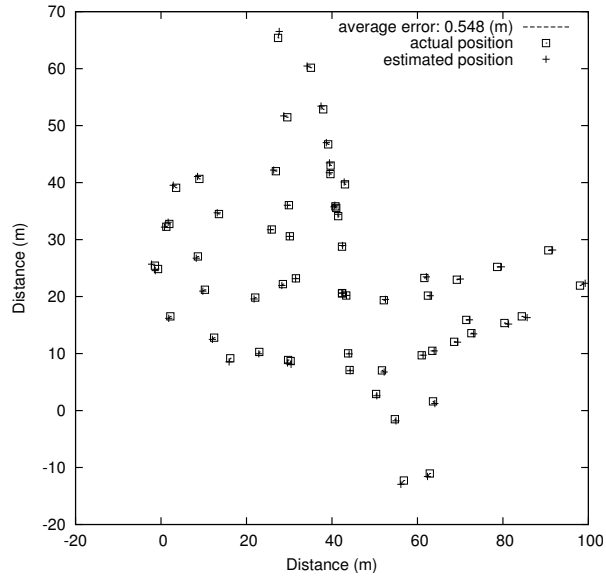


Figure 21: Centralized LSS localization simulation results with the minimum distance constraint.

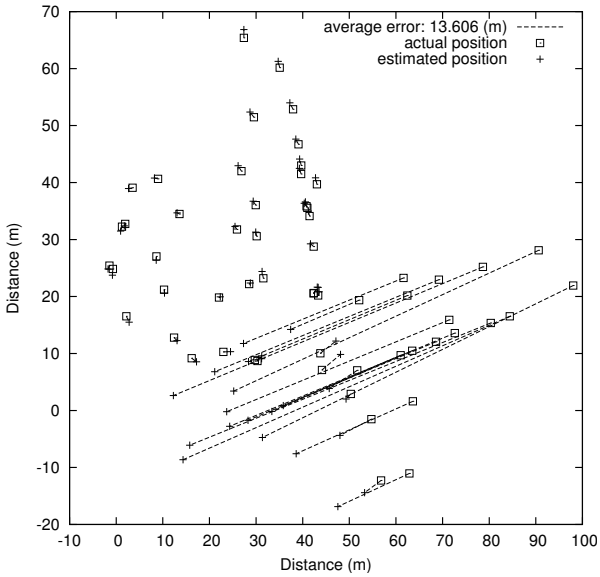


Figure 22: Centralized LSS localization simulation results without the minimum distance constraint.

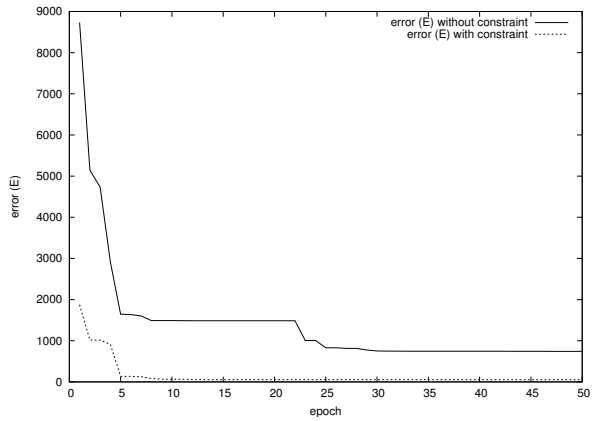


Figure 23: Changes of errors over time in the minimizations of Figure 21 (with constraint) and Figure 22 (without constraint).

Given the measurements, each node uses the LSS localization 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 T between 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 $(\theta, t_x, t_y, f) = \text{argmin}\, E_f$, f=1, -1, where

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

and

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

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 mote.

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:

$$\sum_{n \in C} (\cos\theta\ (u_n - \mu_u) - \sin\theta\ (v_n - \mu_v) - (x_n - \mu_x))^2 + (\sin\theta\ (u_n - \mu_u) + \cos\theta\ (v_n - \mu_v) - (y_n - \mu_y))^2$$

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 T_f is:

$$T_f = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -\mu_u & -\mu_v & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ f\sin\theta & f\cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \mu_x & \mu_y & 1 \end{bmatrix}.$$

We choose whichever of T_1 or T_{-1} yields smaller E.

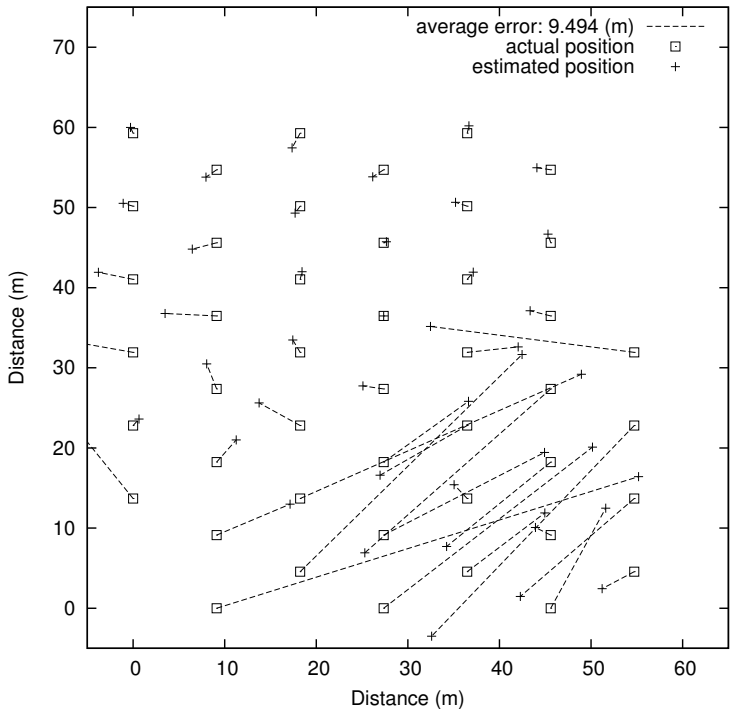


Figure 24: Distributed LSS localization results. Node positions and measured distances are the same as those used in Figure 18. The root node is at (27, 36).

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 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 because minimization, which is the most time-consuming component of the LSS localization, is performed with a smaller number of distance measurements and concurrently at individual nodes.

4.3.2 Experimental Evaluation

Figure 24 shows experimental results for the distributed LSS localization using the same set of measurements as used in the experiment of Figure 18. The node at (27, 36) is the root node. The figure shows about 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 247 total distance measurements between pairs were available for the 47 nodes.

To verify the cause of poor performance, we repeated the localization procedure by adding simulated data to provide an extended set of distance measurements. We selected 370 pairs of nodes whose separation was less than 22m and perturbed their actual distances by errors drawn from the Gaussian distribution $N(\mu = 0; \sigma = 0.33\text{m})$. Figure 25 shows the results. As expected, we achieved fairly good localization results; all nodes were localized, with an average localization error of 0.5m.

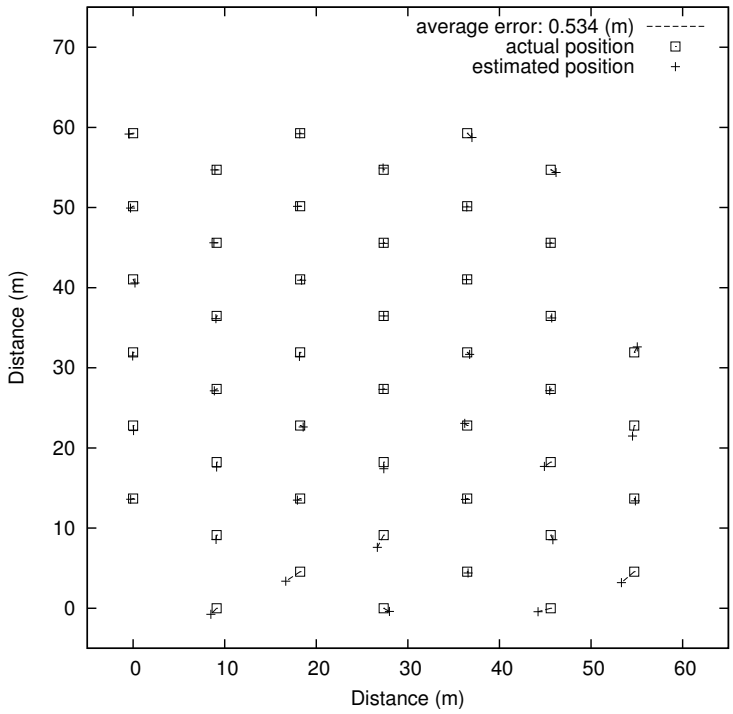


Figure 25: Distributed LSS localization results with extended distance measurements. Measurements in Figure 24 were used along with additional 370 "simulated" distances. Average localization error is 0.5m.

5 Conclusion

We have designed and implemented a fully functional localization system for wireless sensor networks, in particular for the MICA2 mote platform. By increasing the output signal power with a louder speaker and 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 20m maximum, 10m 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 is successful in overcoming the adverse 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. The software tone detection algorithm needs to be extensively tested and additional improvements and optimization made before it can be used with confidence in a ranging service for WSN platforms without a hardware tone detector. For faster and more accurate localization, further research is necessary to fully exploit other information, such as the deployment pattern or node density. 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.

References

[1] Pierpaolo Bergamo and Gianluca 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] Interagency GPS Executive Board. http://www.igeb.gov/sa/.

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

[4] Thomas H. Cormen, Charles E. Leiserson, and Ronald L. Rivest. *Introduction to Algorithms*. MIT Press, 1990.

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

[6] Lance Doherty, Kristofer S. J. Pister, and Laurent El 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.

[7] Roger Fletcher. *Practical Methods of Optimization*. Wiley, 1987.

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

[9] Bernhard Hofmann-Wellenhof, Herbert Lichtenegger, and James Collins. *Global Positioning System: Theory and Practice*. Springer-Verlag, 4th edition, 1997.

[10] Xiang Ji and Hongyuan 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).

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

[12] 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.

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

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

[15] Nissanka B. Priyantha, Anit Chakraborty, and Hari 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] Yi Shang and Wheeler Ruml. Improved MDS-based localization. In *IEEE INFOCOM*, 2004.

[19] Yi Shang, Wheeler Ruml, Ying Zhang, and Markus P.J. 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.