2005 IEEE 6th Workshop on Signal Processing Advances in Wireless Communications

ADAPTIVE NEIGHBORHOODS FOR MANIFOLD LEARNING-BASED SENSOR LOCALIZATION

Neal Patwari and Alfred O. Hero III

University of Michigan, Dept. of Electrical Engineering & Computer Science
1301 Beal Avenue, Ann Arbor, MI, USA
E-mail: [npatwari, hero]@eecs.umich.edu

ABSTRACT

Connectivity measurements, *i.e.*, whether or not two sensors can communicate, can be used to calculate localization in networks of inexpensive wireless sensors. We show that a Laplacian Eigenmaps-based algorithm, combined with an adaptive neighbor weighting method, can provide an accurate, low complexity solution. Laplacian Eigenmaps is a manifold learning method which optimizes using eigendecomposition, thus is non-iterative and finds the global optimum. Comparatively, the new localization method is less computationally complex than multi-dimensional scaling (MDS), and we show via simulation that it has lower variance.

1. INTRODUCTION

Emerging applications of wireless sensor networks will depend on automatic and accurate location of thousands of sensors. Device cost will be a key factor. By eliminating the need for additional hardware, such as for GPS, ultrasound, or high accuracy RF time-of-arrival (TOA), we can widen the sensor network application space. In this paper, we compare localization algorithms which use connectivity measurements. If a sensor can successfully demodulate the packets transmitted by another sensor, then the two are considered to be connected. When received signal strength (RSS) is too low, packets can't be demodulated, and sensors will not be connected.

This paper emphasizes that connectivity is a measurement subject to error due to the unpredictable RF channel. Noisy measurements lead to noisy coordinate estimates. Localization algorithms must be chosen to minimize the bias and variance of the coordinate estimates, and to keep computational complexity low, so that sensor localization will scale well with the size of the network. This paper introduces a Laplacian Eigenmaps based localization method which has both lower computational complexity and lower variance than MDS-based methods.

1.1. Estimation Problem Statement: Formally stated, we consider a network of n unknown-location sensors, and m reference (or anchor) sensors which have *a priori* knowledge of their coordinates (for a total of $N = n + m$ sensors). The cooperative localization problem we consider in this paper is the estimation of the unknown-location sensor coordinates $\{\mathbf{z}_i\}$ for $i = 1 \ldots n$, given: (1) Reference coordinates $\{\mathbf{z}_i\}_{i=n+1 \ldots N}$, and (2) Pair-wise connectivity measurements $\{Q_{i,j}\}$. Connectivity measurements are random variables, subject to error, and their statistical model is described in Section 2. In this paper, we do not consider measurements of TOA, RSS, or AOA; furthermore, references are assumed to have exact coordinate knowledge. Extension of these results is future work.

1.2. Relevant Research: Connectivity measurement-based localization, also called *range free* localization, has found considerable application in ad hoc networks and wireless sensor networks, eg., in [1, 2, 3]. In particular, localization via MDS was introduced in [2], which demonstrated that localization can be achieved without resorting to iterative optimization algorithms that don't always converge to the global maxima. The MDS-MAP method in [2] effectively applies the manifold learning technique called Isomap [4] to the connectivity-based sensor localization problem. We compare our new method to the MDS-MAP method in Sections 3.1 and 5.

2. CONNECTIVITY MEASUREMENT MODEL

The key to developing reliable localization systems is to accurately represent the severely degrading effects of the RF channel. We do *not* consider two sensors to be connected solely based on the distance between them – two sensors are connected if the receiving sensor can successfully demodulate packets transmitted by the other sensor. The receiver fails to successfully demodulate packets when the received signal strength (RSS) is too low. Since RSS is a random variable due to the unpredictability of the fading channel, and connectivity is a function of RSS, connectivity is also a random variable.

Specifically, the connectivity measurement of sensors i and j, $Q_{i,j}$, is modeled as a binary quantization of RSS,

$$Q_{i,j} = \begin{cases} 1, & P_{i,j} \geq P_1 \\ 0, & P_{i,j} < P_1 \end{cases} \quad (1)$$

This research was supported in part by National Science Foundation ITR Grant No. CCR-0325571.

0-7803-8867-4/05/$20.00©2005 IEEE

where $P_{i,j}$ is the received power (dBm) at sensor i transmitted by sensor j, and P_1 is the receiver threshold (dBm) under which packets cannot be demodulated. This step function is an approximation, but is an accurate model for most digital receivers.

Received power $P_{i,j}$ is strongly affected by shadowing and multipath fading. For a particular environment of deployment, the walls, furniture, buildings, trees and other obstructions in the channel between the two devices cause these deleterious channel effects. Since we can't predetermine the exact layout of every place of deployment, we have to consider these effects to be random. Both empirical and theoretical evidence shows that RSS is well-modeled as a log-normal random variable [5]. Since $P_{i,j}$ is expressed in dB, it is Gaussian distributed, with mean $\bar{P}(d_{i,j})$ and variance σ^2_{dB}. The mean received power $\bar{P}$ is an exponentially decreasing function of the actual transmitter-receiver separation distance $d_{i,j} = \|\mathbf{z}_i - \mathbf{z}_j\|$,

$$\bar{P}(d_{i,j}) = P_0 - 10n_p \log \frac{d_{i,j}}{d_0} \tag{2}$$

where P_0 is the received power (dBm) at a short reference distance d_0, and n_p is the 'path-loss exponent', typically between 2 and 4. Also, values for σ_{dB} are usually between 4 and 12 [5]. The precision possible from connectivity-based localization is proportional to the ratio σ_{dB}/n_p [6].

Because of (2), we can talk about the distance R at which the mean received power is equal to the receiver threshold P_1.

$$R = d_0^{\frac{P_0 - P_1}{10n_p}} \tag{3}$$

We call R the 'mean communication range'. Two devices separated by R have a 50% chance of being connected.

In real networks, connectivity is not symmetric. If a pair of devices don't have the same transmit power, they will be connected more often when the device with higher transmit power is transmitting. However, an asymmetric connectivity graph provides *more* information than a symmetric graph: devices are 'in-range', 'out-of-range', or 'intermediate-range' (when devices are connected in only one direction). Essentially, asymmetric connectivity measurements are equivalent to 3-level quantized received signal strength (QRSS) [6]. Localization in this paper is limited to less informative, symmetric connectivity measurements.

3. LAPLACIAN EIGENMAPS

The Laplacian Eigenmaps method considers the minimization of the cost S_{LE} [7]:

$$S_{LE} = \sum_{i,j} w_{i,j} \|\mathbf{z}_i - \mathbf{z}_j\|^2 \tag{4}$$

subject to the translation and scaling constraints,

$$\sum_i \mathbf{z}_i = 0 \quad \text{and} \quad \sum_i \|\mathbf{z}_i\|^2 = 1. \tag{5}$$

The minimum of cost S_{LE} without any constraints would occur when all the coordinates $\mathbf{z}_i$ were equal. The constraints in (5) remove the translation ambiguity by setting the origin as the center, and counteract the tendency to put all points at the origin by mandating a unit norm average coordinate.

The benefit of the formulation in (4) and (5) is that the globally optimum solution can be found via eigen-decomposition. We define the $N \times N$ weight matrix $W = [[w_{i,j}]]_{i,j}$ and its column sums (or row sums, since W is symmetric) $u_i = \sum_{j=1}^{N} w_{i,j}$. Then the graph Laplacian L is defined as

$$L = \text{diag}[u_1, \ldots, u_N] - W \tag{6}$$

where $\text{diag}[u_1, \ldots, u_N]$ is the diagonal matrix with $\{u_i\}$ on its diagonal. The eigen-decomposition of L is the set of $(\lambda_k, \mathbf{v}_k)$, for eigenvalues λ_k and eigenvectors $\mathbf{v}_k$, $k = 1 \ldots N$. Here, we assume w.l.o.g. that the eigenvectors are sorted in increasing order by magnitude of eigenvalue. As presented in detail by Belkin and Niyogi in [7], the $\mathbf{v}_k$ for $i = 2 \ldots d+1$ provide the optimal lowest-cost, d-dimensional solution to (4). Specifically,

$$\mathbf{z}_i = [\mathbf{v}_2(i), \ldots, \mathbf{v}_{d+1}(i)], \tag{7}$$

where $\mathbf{v}_k(i)$ is the ith element of the kth eigenvector.

Finding the smallest eigenvalues and eigenvectors of a sparse and symmetric matrix is a problem which has been studied for decades in physics and chemistry [8, 9], and can be solved using distributed algorithms for parallel processing. The computational complexity is $\mathcal{O}(KN^2)$, where K is the average number of neighbors. Note that MDS requires decomposition of a full matrix, which is an $\mathcal{O}(N^3)$ operation.

1.2. Multi-Dimensional Scaling: Classical MDS finds the coordinates $\{\mathbf{z}_i\}$ which minimize the following cost function:

$$S_{MDS} = \sum_{i,j} \left(\delta^2_{i,j} - \|\mathbf{z}_i - \mathbf{z}_j\|^2\right)^2 \tag{8}$$

where $\delta_{i,j}$ is a measured distance between sensors i and j. In MDS-MAP [2], $\delta_{i,j}$ is set to the shortest-path number of hops between sensors i and j. Note that the difference in (8) is not taken between distances, but between *squared* distances, in order to linearize the optimization problem. Using squared distances tends to emphasize the pairs with high $\delta_{i,j}$ and magnify their errors.

More fundamentally, (8) is based on *distance* rather than connectivity. Equation (8) asserts that the distance between $\mathbf{z}_i$ and $\mathbf{z}_j$ should be equal to $\delta_{i,j}$. In comparison, the Laplacian Eigenmaps cost in (4) simply asserts that the distance between sensors i and j is low.

4. WEIGHT SELECTION

The selection of weights $w_{i,j}$ for neighboring sensors is critical to localization performance. In the original Laplacian Eigenmaps method [7], weights are selected by looking at the local geometric structure of neighboring high-

dimensional data points. This paper presents multiple methods of weight selection, and then compares them via simulation in Section 5.

First, in the *Equal Weights* method, we set $w_{i,j} = Q_{i,j}$, *i.e.*, $w_{i,j} = 1$ if i and j are connected and 0 if not. As will be shown in Section 5, this is a poor weight selection method, because sensors with the most neighbors will tend to have too much 'pull', and will bias their neighbors' coordinate estimates too close to their own.

To counteract this tendency, we offer two alternatives which both affect the column sums of W, *i.e.*, $u_i = \sum_{j=1}^{N} w_{i,j}$. Note that u_i is analagous to the 'pull' of sensor i. In both alternative methods, we first set $w_{i,j}$ using the equal weights method. Then,

Equal Sum-of-Weights : Adjust the weights such that the new column sums $\tilde{u}_i = \mu_u$ for all $i = 1 \ldots N$, where μ_u is the average of the original column sums, $\mu_u = \frac{1}{N}\sum_{i=1}^{N} u_i$.

Linear Sum-of-Weights : Adjust the weights such that the new column sums $\tilde{u}_i$ are linearly related to u_i. Specifically, let $\tilde{u}_i = \mu_u + \beta(u_i - \mu_u)/\sigma_u$ where σ_u is the standard deviation of $\{u_i\}_{i=1}^{N}$. In this paper, slope $\beta = 0.1$ is used throughout.

Adjustment of weights to achieve desired column sums is described in Section 4.1. The W output by any neighbor weight selection method is then used to calculate coordinate estimates $\{\hat{\mathbf{z}}_i\}$ via the Laplacian Eigenmaps algorithm in Section 3.

4.1. Symmetric Adjustment of Weights: Matrix W must remain symmetric after any weight adjustment, since it describes a symmetric graph. If we just scaled the weights in column i by $\tilde{u}_i/u_i$, column i would have the desired sum $\tilde{u}_i$, but W would not remain symmetric. In this weight-adjustment algorithm, we iteratively adjust $w_{i,j}$ (or equivalently $w_{j,i}$) until $\tilde{u}_i = u_i$. The inputs to the algorithm are: the original weights $\{w_{i,j}\}$; the desired sum of weights $\{\tilde{u}_i\}$ for $i = 1 \ldots N$; and a convergence threshold ϵ (here $\epsilon = 0.01$). The algorithm outputs the modified weight matrix. The steps are:

1. Calculate $u_i = \sum_{j=1}^{N} w_{i,j}$ for $i = 1 \ldots N$.
2. Define $\gamma_i = \sqrt{\tilde{u}_i/u_i}$, for $i = 1 \ldots N$.
3. Assign $w_{i,j} = w_{j,i} := \gamma_i w_{i,j} \gamma_j \; \forall$ neighbors i, j.
4. If $\forall i, 1 - \epsilon < \gamma_i < 1 + \epsilon$, stop. Else go to 1.

The algorithm requires $\mathcal{O}(KN)$ multiplies, where K is the average number of neighbors. We do not address the convergence of this algorithm here, except to note that in simulations, it typically converges in 5-10 iterations.

4.2. Two-Stage Weight Selection: In [10], it was shown that localization estimates can be greatly improved by using a two-stage neighbor selection method. We consider the following two-stage algorithm for weight selection:

1. Using the linear sum-of-weights method to set W, calculate the Laplacian Eigenmaps coordinate estimates $\{\hat{\mathbf{z}}_i\}$.
2. For the 2nd round, let the desired column sums $\tilde{\tilde{u}}_i$ be

$$\tilde{\tilde{u}}_i = \tilde{u}_i \sqrt{K_i/\tilde{K}_i} \tag{9}$$

where $\tilde{K}_i$ is the number of its neighbors j for which $\|\hat{\mathbf{z}}_i - \hat{\mathbf{z}}_j\| < R$, and R is the radius of coverage. Adjust W to meet $\{\tilde{\tilde{u}}_i\}$, and then calculate final coordinate estimates $\{\hat{\mathbf{z}}_i\}_{i=1\ldots N}$.

Intuitively, if few of the neighbors of sensor i are estimated to be within its communication range, then we can guess that sensor i's weights should be increased. The presented choices are by no means optimal, and other iterative algorithms or updates are certainly possible. We simply show that the performance of this ad hoc method does in fact dramatically improve localization performance.

5. SIMULATION RESULTS

In this section we test the localization performance using different estimators in multiple sensor geometries. For each test, we run 200 independent simulation trials in order to determine the mean coordinate estimate $\bar{\mathbf{z}}_i$ for $i = 1 \ldots n$, and the covariance matrix $\mathbf{C}$. In each simulation, the statistical model in Section 2 is used to randomly generate connectivity measurements in the sensor network. Each plot in Fig. 1 shows $\bar{\mathbf{z}}_i$ (▼) and the 1-σ covariance ellipse (—) for each sensor. For comparison, we always plot in gray (or red in the electronic version) the actual sensor coordinate (●) and the Cramér-Rao bound (CRB) for the 1-σ covariance ellipse (- - - -) [6]. For each test, we summarize the mean bias $\bar{b} = \frac{1}{n}\sum_{i=1}^{n} \|\mathbf{z}_i - \bar{\mathbf{z}}_i\|$ and the standard deviation, $\bar{\sigma} = \sqrt{\frac{1}{n}\mathrm{tr}\,\mathbf{C}}$, of the localization estimator. Note all distances are in terms of L, the chosen scale of the network.

5.1. MDS-MAP

We first test MDS-MAP in a 7 by 7 grid network, in which the four corner sensors are reference sensors, and the other 45 are unknown-location sensors. For a communication radius $R = 0.5$, the MDS method has standard deviation of location error $\bar{\sigma} = 0.218$ and a bias of $\bar{b} = 0.087$, as shown in Fig. 1(a). At $R = 0.5$, almost all pairs of sensors are within 1 or 2 hops from each other. At lower radii R, the MDS-MAP achieves very low bias, as shown in Table 5, but the standard deviation of error is largely constant, consistently about twice the lower bound of the CRB.

5.2. Laplacian Eigenmaps One-Stage

Equal Weights: Next, we test Laplacian Eigenmaps using the equal weights method (as described in Section 4). The simulation results show a heavily biased estimator. For $R = 0.5$, the results are shown in Fig. 1(b), in which the mean

Location Estimator	MDS-MAP	LE Eql. Wts.	LE Eql. $\sum$-Wts.	LE Lin. $\sum$-Wts.	LE 2-Stage Linear Sum-of-Weights			
Geometry	7 by 7 Grid				Grid+$Z/4$	Grid+$Z/2$	Unif. Rand	
$\mathbf{R = 0.3}$	$\bar{b} = 0.026$	$\bar{b} = 0.106$	$\bar{b} = 0.056$	$\bar{b} = 0.048$	$\bar{b} = 0.039$	$\bar{b} = 0.046$	$\bar{b} = 0.062$	$\bar{b} = 0.069$
	$\bar{\sigma} = 0.205$	$\bar{\sigma} = 0.191$	$\bar{\sigma} = 0.153$	$\bar{\sigma} = 0.153$	$\bar{\sigma} = 0.133$	$\bar{\sigma} = 0.142$	$\bar{\sigma} = 0.155$	$\bar{\sigma} = 0.126$
$\mathbf{R = 0.4}$	$\bar{b} = 0.022$	$\bar{b} = 0.154$	$\bar{b} = 0.059$	$\bar{b} = 0.035$	$\bar{b} = 0.033$	$\bar{b} = 0.037$	$\bar{b} = 0.055$	$\bar{b} = 0.048$
	$\bar{\sigma} = 0.205$	$\bar{\sigma} = 0.188$	$\bar{\sigma} = 0.143$	$\bar{\sigma} = 0.144$	$\bar{\sigma} = 0.136$	$\bar{\sigma} = 0.139$	$\bar{\sigma} = 0.141$	$\bar{\sigma} = 0.127$
$\mathbf{R = 0.5}$	$\bar{b} = 0.087$	$\bar{b} = 0.186$	$\bar{b} = 0.040$	$\bar{b} = 0.036$	$\bar{b} = 0.026$	$\bar{b} = 0.027$	$\bar{b} = 0.040$	$\bar{b} = 0.031$
	$\bar{\sigma} = 0.218$	$\bar{\sigma} = 0.189$	$\bar{\sigma} = 0.149$	$\bar{\sigma} = 0.146$	$\bar{\sigma} = 0.144$	$\bar{\sigma} = 0.147$	$\bar{\sigma} = 0.149$	$\bar{\sigma} = 0.140$
Figure	1(a)	1(b)	1(c)	1(d)	1(e)	1(f)	1(g)	1(h)

Table 1. Simulated performance of MDS-MAP and Laplacian Eigenmaps (LE) using equal weights, equal sum-of-weights, linear sum-of-weights, and two-stage linear sum-of-weights. The sensor geometries are the 7 by 7 grid, grid plus noise ($c = 2$ and $c = 4$), and uniform random.

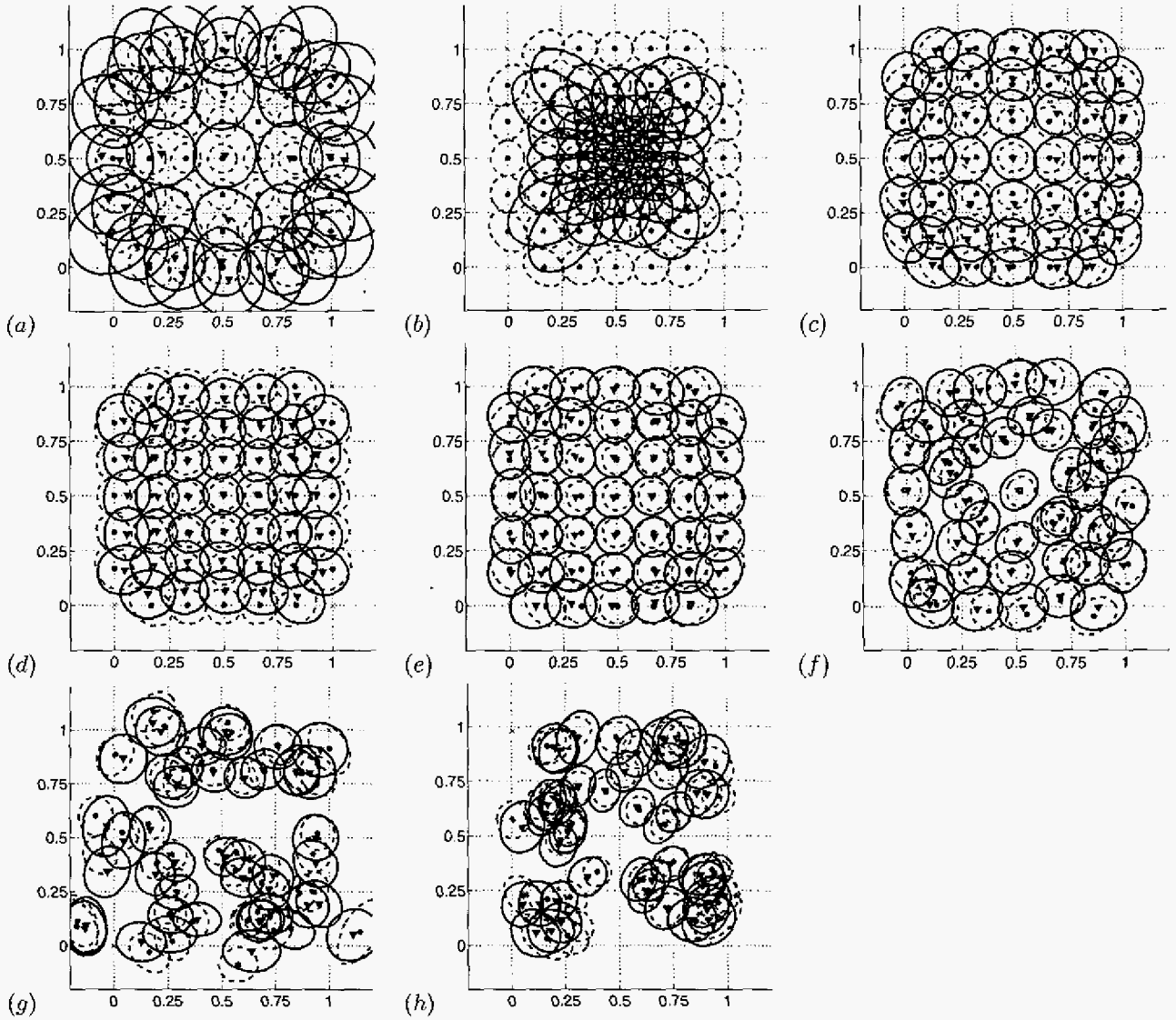

Fig. 1. Estimator mean (▼) and 1-σ uncertainty ellipse (—) for each blindfolded sensor compared to the true location (•) and CRB on the 1-σ uncertainty ellipse (- - - -), when reference sensors are located at each ×. All cases are $R = 0.5$ tests described in Section 5 and in Table 1.

bias $\bar{b} = 0.186$ and the standard deviation of location error $\bar{\sigma} = 0.189$. At $R = 0.3$ and $R = 0.4$, the biases $\bar{b}$ listed in Table 5 are lower but still very high.

Equal Sum-of-Weights: The performance of Laplacian Eigenmaps, when weights are determined by the equal sum-of-weights method, is dramatically better than the equal weights method, as shown in Table 5. For $R = 0.5$, the results shown in Fig. 1(c) show that the edge nodes seem to have weights too high compared to the interior nodes, the opposite bias pattern compared to Fig. 1(b).

Linear Sum-of-Weights: The Laplacian Eigenmaps with adjusted sum-of-weights reduces the bias compared to equal sum-of-weights. As shown in Table 5 and in Fig. 1(d), the bias has been reduced, especially at $R = 0.4$, even though the values of $\bar{\sigma}$ are largely unchanged.

5.3. Laplacian Eigenmaps Two-Stage

Using the two-stage weight adjustment described in Section 4.2, bias is further reduced. Furthermore, as shown in Table 5, the variance for $R = 0.3$ and 0.4 is dramatically lower than the one-stage linear sum-of-weights method. These variances in the grid geometry are about 30-35% higher than the Cramér-Rao lower bound, so even an efficient estimator would not reduce $\bar{\sigma}$ dramatically further.

However, we certainly don't expect that sensors will be arranged in a perfect grid. The true test of sensor localization is performance sensor placement is random, which is presented next. Each test shows the performance of the Laplacian Eigenmaps two-stage weight selection method.

Grid Plus Noise: First, we add a Gaussian random vector to each unknown grid coordinate, *i.e.*, for $i = 1 \ldots n$, $\mathbf{z}_i = \dot{\mathbf{z}}_i + \mathbf{Z}_i/c$, where $\dot{\mathbf{z}}_i$ is the original coordinate on the 7 by 7 grid, and $\{\mathbf{Z}_i\}$ are independent Gaussian-distributed with mean zero and covariance $(1/6)^2\mathbf{I}_2$, and $c = 2$ or 4. Essentially, the standard deviation of the random addition is either one-fourth or one-half of the distance between grid nodes. We generate two geometries from this model for $c = 2$ and $c = 4$, and show simulation results in Table 5 and in Fig. 1(f-g). For $R = 0.4$ and 0.5, the bias and standard deviation of location error increase only slowly. However, for $R = 0.3$, the bias and variance do increase considerably. We note that sensors actually located outside of the unit square $[0, 1]^2$ have noticeably higher bias and variance.

Uniform Random: Next, for $i = 1 \ldots, N$, $\mathbf{z}_i$ are independently chosen from a uniform distribution over the unit square area, $[0, 1]^2$. The sensors closest to each corner are selected as the 4 references, so in this test, even the references are randomly deployed. The resulting $\bar{\sigma}$ are lower than in the 7 by 7 grid or the grid plus noise geometries. We note that the CRB for $\bar{\sigma}$ is also about 15% lower for this deployment compared to the 7 by 7 grid, so it is legitimate to expect lower $\bar{\sigma}$. Essentially, sensors very close together can provide increased information about their relative location. However, the biases $\bar{b}$ are higher than the 7 by 7 grid, especially for $R = 0.3$.

6. DISCUSSION AND CONCLUSION

For random deployments, a low communication radius like $R = 0.3$ may cause some sensors to have very few neighbors, and localization performance will suffer. System designers should plan for the tendency of sensors outside of the convex hull of the reference nodes to experience higher localization errors.

Using a realistic statistical model for connectivity, simulations show the potential of the Laplacian Eigenmaps method to be a robust, low-bias and low-variance sensor location estimator. It does not suffer from local optima and it has low computational complexity compared to MDS-based estimators. The presented two-stage weight-selection method is used to achieve low bias and standard deviation within 35% of the lower bound. However, general analysis of weight selection methods has not been attempted. Finally, distributed algorithms have not yet been presented for the proposed methods. These issues remain open for future research.

7. REFERENCES

[1] Dragos Niculescu and Badri Nath, "Ad hoc positioning system," in *IEEE Globecom 2001*, April 2001, vol. 5, pp. 2926 –2931.

[2] Yi Shang, Wheeler Ruml, Ying Zhang, and Markus P. J. Fromherz, "Localization from mere connectivity," in *Mobihoc '03*, June 2003, pp. 201–212.

[3] Niveditha Sundaram and Parameswaran Ramanathan, "Connectivity based location estimation scheme for wireless ad hoc networks," in *IEEE Globecom 2002*, Nov. 2002, vol. 1, pp. 143–147.

[4] Joshua B. Tenenbaum, Vin de Silva, and John C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, pp. 2319–2323, Dec 2000.

[5] Theodore S. Rappaport, *Wireless Communications: Principles and Practice*, Prentice-Hall Inc., New Jersey, 1996.

[6] Neal Patwari and Alfred O. Hero III, "Using proximity and quantized RSS for sensor localization in wireless networks," in *IEEE/ACM 2nd Workshop on Wireless Sensor Nets. & Applications*, Sept. 2003.

[7] Mikhail Belkin and Partha Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Computation*, vol. 15, no. 6, pp. 1373–1396, June 2003.

[8] Earnest R. Davidson, "The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real-symmetric matrices," *J. Comput. Phys.*, vol. 17, no. 1, pp. 87–94, Jan. 1975.

[9] Luca Bergamaschi, Giorgio Pini, and Flavio Sartoretto, "Computational experience with sequential and parallel,preconditioned jacobidavidson for large, sparse symmetric matrices," *J. Computational Physics*, vol. 188, no. 1, pp. 318–331, June 2003.

[10] Jose A. Costa, Neal Patwari, and Alfred O. Hero III, "Distributed multidimensional scaling with adaptive weighting for node localization in sensor networks," *IEEE/ACM Trans. Sensor Networks*, submitted May 2004, (revised Jan. 2005).