Location Discovery using Data-Driven Statistical Error Modeling

Jessica Feng
Computer Science Dept.
University of California, Los Angeles
Los Angeles, USA

Lewis Girod
Computer Science Dept.
University of California, Los Angeles
Los Angeles, USA

Miodrag Potkonjak
Computer Science Dept.
University of California, Los Angeles
Los Angeles, USA

***Abstract*—We have developed statistical error modeling techniques for acoustic signal detection-based ranging measurements in the framework of wireless ad-hoc sensor networks (WASNs). The models are used as the basis for solving the location discovery problem in sensor networks. We first demonstrate that the major difficulty in location discovery is how to treat errors by proving the location discovery in presence of noisy measurements is a NP-complete problem, even in one-dimensional space. Consequently, we formulate the location discovery as an instance of nonlinear function minimization that optimizes each of the empirically derived statistical error models. The minimization problem is then solved using a conjugate gradient-based nonlinear function optimization solver.**

We validate the efficiency of the approach by conducting comprehensive experiments on both deployed and simulated WASNs. The results indicate that the statistical model-based approach significantly improves the location accuracy compared with the approaches using the traditional optimization objectives. In addition, the localized version of our location discovery algorithm is capable of finding competitive solutions using significantly lower communication cost.

Keywords-Statistical error modeling; Location discovery

I. INTRODUCTION

Location discovery (LD) or localization is a highly important task in many sensor network and pervasive computing applications. Numerous problem formulations have been proposed for localization that target different technologies for distance measurements, use different optimization mechanisms, and impose different sets of constraints and objectives [1][2]. These efforts form strong foundations for addressing location discovery in sensor networks.

Interestingly, the characterization of errors in distance measurements has been rarely addressed. As demonstrated by the following small motivational example, the overall accuracy of the location discovery is often strongly correlated to the accuracy of the error model employed.

A. Motivational Example

Consider a WASN shown in Figure 1, where nodes N_1 to N_9 are aware of their exact locations but node N_{10} knows only its measured distances to the other nodes. Table 1 indicates the *real distances*, which are obtained by applying the distance formula given the true positions of the nodes, the *measured (Euclidian) distances* and the normalized distance errors from node N_{10} to all other nodes. The nodes and the distance measurements are randomly selected from a deployed WASN (see Section III.A). The goal is to calculate the location of node N_{10} (x_{10}, y_{10}). We define the discrepancy between nodes N_1 and N_{10} as $\varepsilon_1 = \sqrt{(x_1 - x_{10})^2 + (y_1 - y_{10})^2} - d_{1,10}$, where the measured distance $d_{1,10}$ is 10.02m and the *calculated distance* is $\sqrt{(x_1 - x_{10})^2 + (y_1 - y_{10})^2}$. Similarly, we can define discrepancies between nodes $N_2,\ldots,N_9$ to N_{10}. The discrepancies (Equation (1)) can be used to guide the location discovery. For example, we can formulate LD as determining (x_{10}, y_{10}) to minimize *S*, the sum of absolute values of the discrepancies:

$$S = |\varepsilon_1| + |\varepsilon_2| + |\varepsilon_3| + |\varepsilon_4| + |\varepsilon_5| + |\varepsilon_6| + |\varepsilon_7| + |\varepsilon_8| + |\varepsilon_9| \quad (1)$$

Equation (1) is the L_1 norm and it is the objective of optimization. Other alternative measures that are often used include $L_2 = \sqrt{\varepsilon_1^2 + \varepsilon_2^2 + \ldots + \varepsilon_9^2}$ and $L_\infty = max\{\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_9\}$. The most popular measure is to use the least linear squares minimization approach that targets the Gaussian error model for the distance measurement errors.

Due to the small size of the instance, it can be easily and optimally solved using exhaustive search, which guarantees that the calculated location is within 0.1mm of the optimal in terms of the targeted objective function. The optimization mechanism produces solutions of location error ***1.272m***, ***5.737m***, and ***8.365m*** when L_1, L_2 and L_∞ are used, respectively. If we assume the Gaussian distribution for the measurement errors and utilize the maximum likelihood (ML) approach to maximize the probabilities of errors occurring, the optimization gives a solution with location error ***0.928m***. However, when we use the distance error model derived from experimental data that does not include these 10 measurements shown in Figure 2 and maximize the product of probabilities of individual discrepancies, the optimization produces a solution with location error $\boldsymbol{1.662 \times 10^{-3} m}$. The error is reduced by more than two orders of magnitude. Although the example is small, it strongly suggests the importance of modeling distance measurement errors.

In practice, we have observed that all optimization solvers have great difficulty producing accurate solutions based on

1-4244-0222-0/06/$20.00 (c)2006 IEEE

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the Proceedings IEEE Infocom.

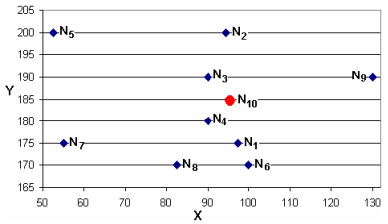


Figure 1. The motivational example topology.

NODE	REAL (R)	MEASURED (M)	ERROR (%) ((R – M)/R)
N_1	10.31	10.02	+2.79%
N_2	15.01	16.59	-10.54%
N_3	7.07	3.02	+57.29%
N_4	7.07	6.67	+5.67%
N_5	45.06	27.65	+38.65%
N_6	15.81	17.34	-9.67%
N_7	41.23	39.84	+3.37%
N_8	19.52	20.22	-3.56%
N_9	35.35	36.46	-3.12%

Table 1. The real, measured and the normalized errors of distance estimates between N_{10} and its neighbors.

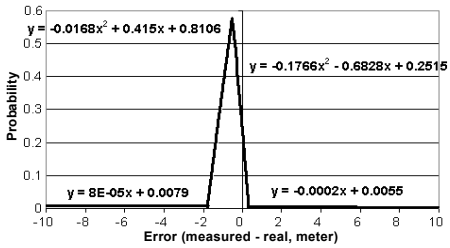


Figure 2. Statistically constructed error model.

measurements that have errors. Therefore, it is clear that measurement errors and error models cause both conceptual and computational difficulties. However, the difficulty of LD can be greatly alleviated when a sound error model is available. Unless the adequate objective function is targeted, regardless of the optimization mechanism, LD will not be effective.

B. New Location Discovery Approah: The Global Flow

In this section we outline the key components of our LD approach. While other sections describe the approach in much more systematic and detailed way, the emphasis in this subsection is on the intuition and reasoning that guide the approach and the process on how the modeling and optimization are conducted.

Our LD approach emphasizes error modeling. The starting point of our approach is traces of collected sets of measured distances. Figure 3 shows 2,000 pairs of measured distances plotted against the corresponding real (correct) distances. The measured distances between a pair of nodes were obtained using the acoustic signal-based ranging method (Section III.A); the real distances were obtained using the distance formula based on the true locations of the nodes. Our analysis shows that the main source of the problem difficulty is errors, since LD is a NP-complete problem even in the 1-d framework (Section III.B). We conducted Chi-Square, Kolmogorov-Smirnov (KS), Anderson-Darling, Cramer-von Mises, and Kupier goodness-of-fit tests [3][4] to evaluate how likely the distance measurements (Figure 3) follow a specific distribution,

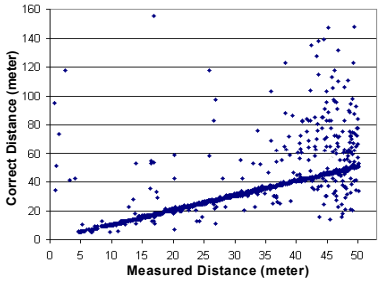


Figure 3. Measured distances vs. Correct Distances.

assuming the data follows one of the following five distributions: Gaussian, beta, gamma, Weibull, and lognormal. The parameters of each distribution were estimated using the maximum likelihood procedure and the Probability Plot Correlation Coefficient (PPCC) method [5]. None of the five considered distributions was able to pass any of the evaluation. The goal of error modeling is not just to answer the question of what is the most likely actual value for a given measured distance (regression), but also to provide the likelihood of any proposed actual distance for a given measured distance (density estimation). We start by developing and evaluating a number of techniques for off-line error modeling that assumes the knowledge of the real distance for each measured distance. We use statistical validation and evaluation techniques to select the most effective procedure (kernel smoothing). Once the error model for individual measurement is available, we analyze the correlation among errors. The statistical-proven independence of the errors provides the justification for the maximum likelihood objective function used by the LD algorithms.

There are several reasons why off-line models are important. First, they enable us to learn about the properties of the error distribution functions, which can be used for faster and less expensive on-line model development (Section V). Secondly, in many actual cases when we have beacons, whose exact locations are provided by GPS devices, the distances between the beacons can be measured. Consequently, we can easily construct an on-line model based on the measurements among beacons using the same approaches. Finally, we show how one can iteratively deduce error models by interleaving LD and error modeling. We quantitatively compare the impact of location accuracy based on off-line and on-line error models in Section VII.

We evaluate several optimization mechanisms for localization and select the best one (Section IV.B). Since the problem is NP-complete, there is well-justified need for considering a variety of powerful optimization mechanisms. Some of the best performing approaches are all based on nonlinear function minimization using continuous optimization techniques. We believe that this is a consequence of the nature of the error model that provides strong hints to the continuous optimization methods which direction to pursue. We also have developed a localized LD algorithm and we demonstrate that it often performs better than centralized both in GPS and GPS-less instances (Section VI). The key reason for this unexpected behavior is that no solver can effectively solve systems with too many variables and constraints. It is beneficial in terms of optimization in limiting the size of considered instances.

We analyze the performance of all proposed error models and optimization mechanisms using networks that are composed based on actually deployed network (e.g. Figure 4(a)). In order to properly evaluate the feasibility and the scalability issues, we have developed an integer linear programming (ILP)-based procedure that guarantees the extraction of a network with user specified properties such as the average number of neighbors, minimal and maximal number of neighbors for each node, and the total number of measurements for a network of given size (Appendix).

Note that the developed error model construction techniques are demonstrated on, but certainly not limited to the acoustic signal-based distance measurements. As part of our future work, we continue to investigate whether the derived error model (as opposed to Gaussian) is a good fit across different ranging methods in different environments. In addition, we are also studying how to incorporate the background noise and significant multipath effects generated by an urban setup. However, this paper should serve as a starting point of investigating error characterization using combinations of parametric and non-parametric statistical methods.

II. Related Work

We survey the most closely related literature on location discovery as well as error characterization in the framework of location discovery. One way of classifying the LD algorithms is based on the availability of distance measurements: range-based and range-free. Range-based localization techniques rely on the availability of the measured (or estimated) Euclidean distances between pairs of communicating nodes, while the range-free techniques pose no such requirements.

For range-based techniques, distance is often measured by exploiting time of arrival [6], received signal strength [7][8], time difference of arrival of two different signals (TDOA) [9][10] and angle of arrival (AOA) [11]. Some of the state-of-the-art range-based location discovery techniques and systems for WASN include [12][13][14][15][16][17]. Biswas and Ye [2] propose a semidefinite programming (SDP) relaxation based localization method where the main idea is to convert the non-convex quadratic distance constraints into linear constraints by introducing a relaxation to remove the quadratic term. The L_1 norm of location errors serves as the optimization target. Galstyan et al. [13] treats localization through online distributed learning and integrates it with target tracking given a fraction of anchor nodes while not requiring the moving object to have a-prior knowledge about its own location. Nasipuri and Li [16] propose a localization method that a sensor node can determine its location by noting the times when it receives the different beacon signals, and evaluating its angular bearings and location with respect to the beacon nodes using triangulation. Shang and Zhang [17] present an algorithm that uses the basic connectivity information – which nodes are in the communication ranges of which others – to derive the locations of the unknown nodes.

Range-free localization techniques do not require the availability of the estimated/measured Euclidean distances between pairs of communicating nodes. He et al. [18] present an area-based range-free localization technique – APIT. The locations are estimated by isolating the environment into triangular regions between anchor nodes. A node's presence inside or outside of these triangular regions allows the node to narrow down the area in which it can potentially reside. The diameter of a node's estimated area can be reduced by utilizing different combinations of anchor positions. Doherty et al. [19] form the localization problem as a constraint satisfaction problem where the constraints are induced based exclusively on connectivity. Connectivity between all pairs of communicating nodes is modeled as a set of geometric constraints on the node positions, and then the system is solved globally in a centralized place. Niculescu and Nath in [20] document the Ad Hoc Positioning System (APS), which is a distributed, hop-by-hop positioning algorithm that resembles an extension of both distance vector routing and GPS positioning given a fraction of anchor nodes.

More recently, GPS-less positioning approaches have also emerged [20][21][22] where the resultant locations of the unknown nodes are relative with respect to their neighboring nodes either in terms of the Euclidean distances or hops. For example, Capkun et al. [22] developed a distributed infrastructure-free (mobile) positioning algorithm that uses the measured Euclidean distances between the nodes to build a relative coordinate system in which the node positions are computed in two dimensions. The authors also demonstrated that relative coordinates are sufficient for applications such as Location Aided Routing and Geodesic Packet Forwarding. In addition, there have been studies on the positioning algorithms targeting mobile sensor networks as well [22][23].

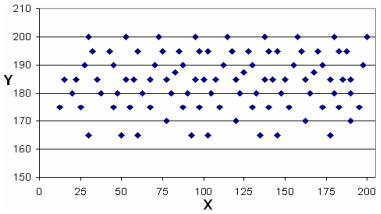
To the best of our knowledge, no comprehensive statistical studies on measurement errors or error modeling have been conducted. Two popular assumptions regarding the measurement/ranging errors include the Gaussian distribution [1][24] and the L_1 norm [12]. When localization is formed in terms of an optimization instance, maximizing the likelihood of Gaussian-based errors or minimizing the L_1 norm of location errors usually serve as the optimization target. For example, Niculescu and Nath [24] derive a Cramer-Rao lower bound for positioning error of multi-hop distance-vector based algorithms based on the assumption of Gaussian error measurements. Savvides et al. [25] conduct comprehensive studies on the position error behavior of multihop localization protocols based on the assumption that the measurement errors are independent Gaussian random variables with zero mean and a known variance.

III. Preliminaries

A. The Distance Measurements

We construct the statistical error models and conduct location discovery on sets of distance measurements that are collected using the acoustic signal detection-based ranging techniques. The number of deployed sensor nodes varies from 79 to 93, with the average being 90. The sensor nodes are custom designed based on an SH4 microprocessor running at 200MHz (Figure 4(b)). The nodes are deployed at the Fort

Leonard Wood Self Healing Minefield Test Facility, which measures 200m x 50m. The nodes are roughly 10.5m apart and the radio signal (communication) range is about 50m. Figure 4(a) shows an example of a deployment topology. Each node is equipped with four independent speakers and microphones that are used as the ranging tool. The distance between two nodes is obtained by timing the arrival of the acoustic signals [26]. Each node in the network takes turns to transmit the acoustic signals; all the nodes that receive the signals record the time of arrival and convert the time of flight to distance in meters. There are 33 sets of distance measurements in total that were collected over the course of few days. Each set consists of one round of acoustic signal transmission by all the nodes. For the sake of simplicity, we demonstrate the algorithms and techniques on a randomly selected set of measurements, and we present the results for ten other randomly selected data sets in Section VII. The details on the experimental setup and the acoustic detection scheme used can be found in [27].



(a)


(b)

Figure 4(a). An example of deployment topology.
Figure 4(b). A SH4 node.

B. Computational Complexity

Recently, it has been proven that localization is a NP-complete problem by transforming the graph embedding problem into the localization problem [28]. It was identified that mirroring and flipping in 3-d space are the cause of computational intractability [15]. In this subsection, we provide an alternative proof of the computational intractability of the localization problem. The impetus for the development of the new proof is provided by our objective to better understand the sources of the complexity, to establish more precise conditions under which localization is NP-complete and to better understand the importance of noisy measurements. Our main conclusion is that localization is NP-complete even when no mirroring and flipping is possible as in the one dimensional localization problem due to errors in measurements. At the same time as we observe, actual distance measurements usually have significant errors and, therefore, there is a need to address error modeling, optimization in presence of error and to develop and employ powerful optimization mechanisms for localization.

We prove the NP-completeness of the 1-d LD problem by polynomial transformation of an instance of the known NP-complete problem – optimal linear arrangement problem [29], into the 1-d localization problems, in polynomial time. For the sake of completeness and readability, we state both problems using the standard Gary-Johnson format:

THE LINEAR ARRANGEMENT PROBLEM

INSTANCE: Graph $G = (V, E)$, positive integer $K \le |V|$.

QUESTION: Is there a one-to-one function $f: V \to \{1, 2, \ldots, |V|\}$ such that $\sum_{\{u,v\}\in E} |f(u) - f(v)| \le K$?

THE 1-D LOCALIZATION PROBLEM

INSTANCE: A network of N sensor nodes, measured distances between all pairs of sensors $P=\{d_{ij}, i,j\in N\}$, a subset of P also has bi-directional measurements $Q=\{d_{ji}\}$, $Q\in P$, positive integer M.

QUESTION: Is there a one-to-one function $g: i \to x_i$ where x_i is the estimated location of i, $i=1,\ldots,N$, such that the overall discrepancy between the calculated distances $|g(i) - g(j)|$ and the measured distances d_{ij} and d_{ji} of all pairs of sensor nodes i and j satisfy the following condition:

$$\sum_{\{i,j\}\in P} \left| |g(i) - g(j)| - d_{ij} \right| + \sum_{\{i,j\}\in Q} \left| |g(i) - g(j)| - d_{ji} \right| \le M \ ?$$

Proof: The reduction from the linear arrangement problem to the localization problem is as follows. Let the vertices in G be the sensor nodes, i.e. $V = \{i, i=1,\ldots,N\}$. Let the edges in G be the measurements in Q, i.e., $E= \{d_{ji}, d_{ji}\in Q\}$.

More formally, let the graph $G = (V, E)$ and the positive integer K constitute an arbitrary instance of the linear arrangement problem. The basic units of the instance of the linear arrangement problem are the vertices and the edges of G. The instance of the localization problem is completely specified by:

$i = \{v: v\in V\}$
$d_{ij} = \{\{i,j\}: i,j\in V\}$
$d_{ji} = \{\{j,i\}: \{j,i\}\in E\}$
$M = K + C$, where $C = \frac{1}{4}(K^2+K+2)(K-1)$

It is easy to see that this instance can be constructed in linear time. Note that the measured distances d_{ij} acts as an "enforcer" [29][30], which impose additional restrictions on the ways the sensor nodes must be placed. Specifically, all the d_{ij} have value $K+h$, where h specifies the least distance between any pair of nodes. This enforcer is necessary and sufficient to prevent multiple nodes being placed on an identical location, which corresponds to the condition that each node must have a unique assignment in the linear arrangement problem, that the distance between any two nodes is at least 1 unit. Therefore, each node maps into a unique integer location between 0 and K.

Function f exists if and only if there exists a function g that satisfies the condition of

$$\sum_{\{i,j\}\in P} |g(i) - g(j) - d_{ij}| + \sum_{\{i,j\}\in Q} |g(i) - g(j) - d_{ji}| \le M$$

Suppose g such that

$$\sum_{\{i,j\}\in P} \left| |g(i) - g(j)| - d_{ij} \right| + \sum_{\{i,j\}\in Q} \left| |g(i) - g(j)| - d_{ji} \right| \le M$$

Consequently, there exists an f such that

$$\sum_{\{i,j\}\in P} |f(i) - f(j)| \le K$$

Therefore, function f satisfies the condition

$$\sum_{\{u,v\}\in E} |f(u)-f(v)| \le K \text{ since } P=\{d_{ji}: \{j,i\}\in E\}$$

IV. Off-line Model Construction

In this section we present the acoustic ranging-based distance measurement error models and the objective functions developed using combinations of parametric and non-parametric statistical techniques. In addition to modeling individual distance measurements, we also statistically analyze the error models associated with a particular speaker or microphone (Section III.A), and nodes that are in a particular geographic area. All models are evaluated using resubstitution [31]. We also present several new techniques for evaluating the error cumulative density functions (CDFs). The input to all procedures is a set of pairs of values. In each pair, one value is the distance measurement obtained based on the acoustic signals (the measured distance) and the other is the distance obtained using a high accuracy manual procedure (the real distance). The goal is to find the probability density function (PDF) of errors for any given measurement.

A. Model Construction

We have developed and analyzed the following five families of error models for distance measurements: (i) *independent of distance* (ID); (ii) *normalized distance* (ND); (iii) *kernel smoothing* (KS); (iv) *recursive linear regression* (LR); and (v) *data partitioning* (DP).

For each type of model, we develop a number of variants and statistically test them in order to select the best one for the optimization process in LD. The first model (independent of distance) does not distinguish between different measurements and considers only the positive or the negative error values (i.e. measured – real), shown in Figure 2. Conceptually, this model is attractive because of its simplicity and the use of a single dimension for all different measurement values. In the second family of models (normalized distance), the error values are defined by normalizing the measured distance against the real distance (measured/real). The model is shown in Figure 5.

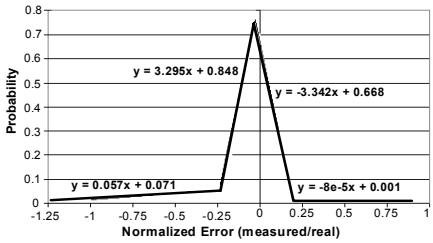


Figure 5. The normalized distance-based error model.

In addition, we also experimented with several more complicated techniques: kernel smoothing and local regression [31]. The basic idea is that when we estimate a continuous density from a dataset, we also seek to smooth the discrete data. The challenge in smoothing is to choose the best bandwidth that balances the desire to reduce the variance of the estimator (which needs lots of data points that we do not have) yet capture significant small-scale features in the underlying distribution (which needs a narrow bandwidth). The kernel smoothing method (KS) convolves the density distribution with a kernel where the user specifies the shape and bandwidth, which supports our primary goal to develop error models that take into account the length of measurements as a prediction parameter. We have experimented with multiple kernel weight and shape functions and selected the 3-d pyramid in our experiments. We use the sliding window kernel smoothing technique [31] to construct the PDF, which is a function of two variables: the measurement errors and the intensity of measurements. Figure 6 shows the model presented in a 2-d plot for easy visualization (the figure only shows the PDFs for five measured distances as an example).

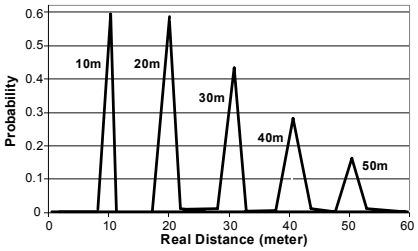
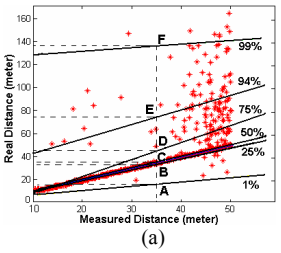
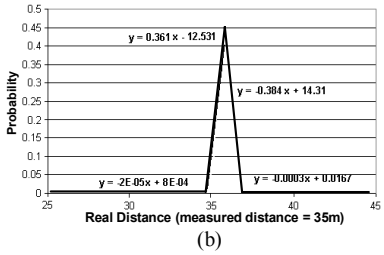


Figure 6. The kernel smoothing-based error model.

The recursive linear regression-based technique (LR) constructs the PDF by utilizing both the measured and the real distances. It is constructed in the following way. First we use the standard linear regression to approximate the real distances as a function of measured distances (the 50% line in Figure 7(a)). The data is naturally partitioned into two fractions by the regression line. We then recursively produce a regression line in both fractions – the 25% and the 75% regression lines respectively. The process is repeated until the specified precision is reached. The precision is set to 1% in our experiments. Given a specific measured distance 35m, the CDF can be constructed by finding the real distance mappings according to the regression lines (shown in Figure 7(a)). Points A to F are the 1%, 25%, 50% 75%, 94% and 99% CDF values respectively. The PDF is then derived from the CDF by subtracting two consecutive terms. Figure 7(b) shows the PDF constructed for the measured distance 35m.



(a)



(b)

Figure 7(a). The linear regression lines.
Figure 7(b) The linear regression-based error model given the measured distance = 35m.

Finally, we explore the data partitioning-based model (DP). The impetus to develop an error model separately for different measurement ranges is provided by the exploratory data analysis. We find that the percentage of outliers in terms of

measurement errors depends on the range of the measured distances. For example, in the majority of our 33 data sets, measurements in the range of 15 to 35m are almost without outliers, while the measurements in the range 40+ m contain more than 80% of all the outliers. The data partitioning is conducted within the framework of dynamic programming which guarantees the optimality under the assumption that the applied regression on each individual segment is optimal. The run time of the algorithm is $O(k \cdot R^2) \cdot O(\text{regression})$, where k is the number of partitions and R is the ratio between the range of the measurements and the minimum size of a partition. Figure 8 shows the PDFs constructed when the data is partitioned into four segments.

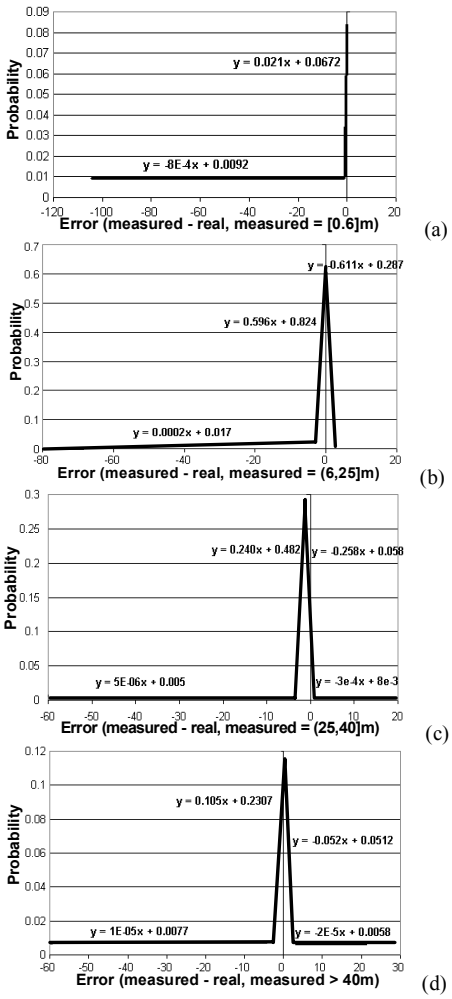


Figure 8. The measurement partitioning-based error model. The error is defined as (measure – real) in meters.

While there are a wide variety of well-proven statistical techniques for the evaluation of regression models, there is very limited literature on the evaluation of density estimation techniques. Therefore, we decide to map the density estimation evaluation problem into the problem of evaluating the regression functions by using the resubstitution paradigm [31]. Resubstitution is the procedure where different K% of the original data is randomly selected as the learning data set to acquire the result, which is then evaluated on the remaining (1–K)% of the testing data. This procedure is repeated R times in order to indicate how frequently different results occur. In our study, K is 60% and R is 200. The key idea in our evaluation is to map each data point in the testing set to its corresponding CDF value, which can be derived from the PDF developed by applying one of the five methods to the learning data set. After each resampling, we plot the testing sets in ascending order where the x-coordinate indicates its ranking normalized against the cardinality of the testing data set, and the y-coordinate shows the product of its CDF value and its ranking. Figure 9(a) shows an example of such plot for the kernel-based model. Note that if the model is perfect, all points will reside on the line $y = x$. Figure 9(b) shows the boxplots of the discrepancy distribution from the line $y = x$ based on the 200 resamplings for all five families of models. A boxplot summarizes a set of data in the following way. The top and bottom lines indicate the maximum and the minimum errors; the top and bottom lines of the rectangle indicate the 75 and 25 percentile values; and the line inside of the rectangle is the median value. We see that the kernel-based and the measurement partitioning-based methods are the best ones in terms of the median discrepancy. Moreover, Table 2 shows the slopes and the R^2 values for all five types of error models when the least linear squares regression is overlaid on the plots such as in Figure 9(a). Again, the results strongly indicate the strength of the kernel-based model. Therefore, we select this method as the basis for constructing the objective function (OF) that serves as the optimization target in LD.

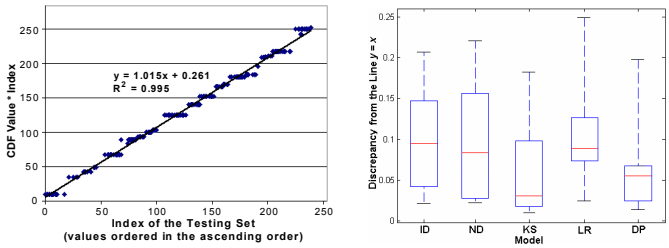


Figure 9(a). The testing set evaluation.
Figure 9(b). The boxplots of discrepancy (from y=x) distribution of the five error models.

MODEL	SLOPE (M)	VARIANCE (R^2)
ID	1.0678	0.9936
ND	0.8287	0.8973
KS	1.0151	0.9955
LR	1.2469	0.8976
DP	0.9193	0.9584

Table 2. The regression line summary of the testing set evaluation for the five error models.

B. *The Objective Function (OF)*

Consider a network of N sensor nodes in a K-d space where each node i has geographic location $(x_{i,1}, x_{i,2}, \ldots, x_{i,K})$, i=1,…,N. d_{ij} indicates the measured distance between a pair of communicating nodes i and j. The individual distance measurement error ε_{ij} associated with i and j is defined in Equation (2). Note that Equation (1) is an instantiation of Equation (2).

$$\varepsilon_{ij} = \left| \sum_{k=1}^{K} \sqrt{(x_{i,k} - x_{j,k})^2} - d_{ij} \right| \quad (2)$$

The objective function is derived by combining the individual discrepancies of all pairs of nodes with distance measurements. More specifically, a function f is defined over the set of discrepancies ε_{ij} for all pairs of i and j, and is subject to minimization (Equation (3)):

$$\text{OF} = f(\varepsilon_{ij}) \quad (3)$$

Commonly used objective functions are metric error norms: L_1, L_2 and L_∞. In the case of the ML-based OF, the OF is the product of probabilities associated with each individual discrepancy (Equation (4)). We denote the function that transforms the discrepancy ε_{ij} into the corresponding probability by following a particular error model as P. The new OF is subject to maximization. In our study, we adopted the kernel-based measurement error model for locations in the 2-d physical space.

$$\text{OF} = \prod_{ij} P(\varepsilon_{ij}) \quad (4)$$

Once the model of individual errors is available, the remaining task is to identify the best possible way to combine them into an overall objective function (OF) that will guide the LD process. One can envision a large number of options. The standard practice is to use L_1, L_2, L_∞, or to apply the maximum likelihood principle. In sensor and ad-hoc wireless network literature, the most common approach is to assume the Gaussian error distribution and follow the ML principle. In addition to these four standard options (L_1, L_2, L_∞, and Gaussian distribution ML), we propose two new maximum likelihood-based OFs.

The ML principle states that we should select the solution which yields errors such that their joint likelihood is optimized. If errors are not correlated, the joint probability is equal to the product of individual probabilities. Otherwise, we have to take into account the joint probabilities and create complex OFs. Therefore, in order to create an accurate OF that is easy to calculate, it is important to identify to what extent the errors between different measurements are correlated.

There are two natural sources of correlation for distance measurements: the use of identical equipment (speaker or microphone) and the impact of the environment (in our case the vicinity of speakers or receivers). We start the analysis by calculating the cumulative distribution function of the measurement errors. The CDF value of a given error indicates the percentage of communication links (measurements) that have smaller error than itself. After that, we examine all the measurements that have the same correlation property (i.e. originated from the same (i) speaker, or (ii) receiver, or (iii) are geographically close).

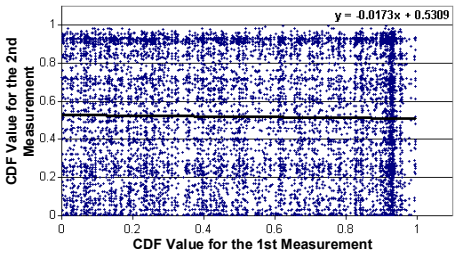


Figure 10. Quality measurements grouped by speakers. The correlation level is very low.

Figure 10 shows the CDF values for all possible pairs of communication links (measurements) grouped by speakers as an example. Each pair of links that originated at the same speaker is characterized by a point in the 2-d space that has x and y coordinates according to its CDF values. The plot shows the absence of correlation, as indicated by the wide spread of data points. Also shown in figure is the best-fit model of the data points, which also hints that there is little correlation among the data points (R^2=0.003, which means that only 0.3% of the data variability is explained by correlation). In addition, we evaluated the correlation significance using the *t-test* [31] (likelihood of accidental presence of the correlation) in all three scenarios across several independent data sets. The probability that the correlation is accidental in all three scenarios across all data sets is always very low (less than 10^{-10}). However, the correlation is also always very low as well (less than 0.01). Therefore, we can conclude that it is not necessary to consider the error correlation during the LD procedure (i.e. error values can be interpreted as independent probabilities in the ML-based optimizations).

We constructed two OFs, the first one is based on the kernel smoothing-based error model within the ML framework. The second OF incorporates an additional heuristic factor: nodes that are closer to beacons receive weight factor proportional to the inverse of their distance to the three closest beacons.

Figure 11 shows the correlation between the kernel-based OF values and the resultant location errors. Table 3 summaries the results of the statistical analysis of the four widely used OFs and two of our new OFs. One conclusion is that the ML-based OF is superior to the norm-based OFs. By far the worst is L_∞ because it focuses only on the single largest error value. Although the Gaussian OF performs reasonably well, it is still inferior when compared to the two new OFs. The third column indicates the consistency between high quality solutions and objective functions. Although the OF constructed with heuristics performs slightly better in terms of consistency, we decide to use the kernel OF because of its generic nature.

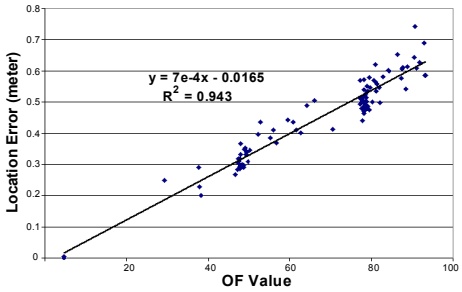


Figure 11. The correlation between the OF values and the resultant location errors of the kernel-based error model.

OF	VARIENCE (R^2)	CONSISTENCY
L_1	0.469	0.720
L_2	0.417	0.742
L_∞	0.090	0.566
GAUSSIAN	0.885	0.86
KERNEL	0.943	0.915
KERNEL+HEURISTIC	0.927	0.938

Table 3. Evaluation of the six objective functions.

In the case of centralized algorithm, we assume that the distance measurements are collected at nodes that receive acoustic signals from their geographical neighbors and are gathered at a centralized location. If the LD problem is GPS-based, we assume that a small fraction of nodes have their locations. The goal is to calculate the locations of all unknown nodes by optimizing the OF derived in the previous section.

Once all the measurements are aggregated at a single point, we evaluated several optimization mechanisms. Overall, our conclusion is that the nonlinear function minimization (Broyden-Fletcher-Goldfarb-Shanno (BFGS) variant of Davidon-Fletcher-Power minimization procedure [32] to be exact) is the best performing method in terms of both location accuracy and run time. The experimental results using the BFGS method are presented in Section VII.

The starting point for GPS-less localization is the identical objective function as in the case when beacons are present. The only change is that in this case all nodes are unknown. We use three steps to match the relative locations produced by the optimization solver against the correct locations of the nodes: (i) *flipping*; (ii) *translation*; and (iii) *rotation*. The details about these operators can be found in [15]. Solutions obtained with and without flipping with respect to the x-axis are always attempted and the better matching solution is preserved.

V. On-line Model Construction with Simultaneous Location Discovery

It is easy to envision many situations when error models for the distance measurements are not available a-priori, including deployment in environments with unknown characteristics, the presence of moving obstacles, employment of new and different models of speakers and microphones, and applications of different technologies for distance measurements. Both in principle and often in practice we can develop the error models in many of these situations by (i) using distance measurements among a relatively small number of nodes relying on GPS (e.g. beacons); or (ii) extrapolating from models developed at similar environments with the same distance measurement equipment. Nevertheless, the importance of the on-line in-field techniques for error model construction is clear.

In this section we present four methods for simultaneous on-line error model construction and LD: (i) parameter fitting; (ii) monolithic approximation; (iii) iterative approximation; and (iv) iterative shape and space approximation. The methods provide trade-offs between the amount of required/assumed information, and the solution quality and the computational complexity. We compare the error models constructed off-line and on-line in Section V.E; and we analyze the impact of location accuracy by adopting off-line and on-line error models in Section VII.

A. Parameter Fitting

Given a set of distance measurements as the input, our goal is to simultaneously determine the known locations and construct an error model. Unfortunately, it is easy to see that it is not possible to solve the LD problem unless a set of restrictions/properties is imposed or assumed on the error model. If there are no such restrictions or assumptions, the solver can always produce an arbitrary solution that follows an arbitrary error distribution perfectly. Therefore, restrictions or assumptions of errors must be imposed and our goal is to find a minimum set of intuitive assumptions that will be applicable to a variety of distance ranging technologies and environments. One example is that the error function has to be unimodular, i.e., there exists an error ε_i such that for any two errors ε_k and ε_j, $\varepsilon_i > \varepsilon_k > \varepsilon_j$ implies $P(\varepsilon_i) > P(\varepsilon_k) > P(\varepsilon_j)$; and there exists error ε_i such that for any two errors ε_k and ε_j, $\varepsilon_i < \varepsilon_k < \varepsilon_j$ implies that $P(\varepsilon_i) < P(\varepsilon_k) < P(\varepsilon_j)$, where P is the probability of the error. From a practical point of view, one can view ε_i as the bias of the imposed random noise of an unspecified distribution. We statistically examined 33 data sets and they all satisfy the unimodular property.

Motivated by the similarities of our data sets, we first addressed the easy and less general but commonly practiced case, where the shape of the error distribution function (error model) is known and we just have to determine the parameters. We start by identifying the shape that is a accurate approximation for error distributions for all 33 sets of data. By approximating the actual distributions of the data sets using least-squares, we have selected two linear functions and two polynomials. Figure 12 shows the selected shape and the 10 parameters used for its characterization. Note that although the number of variables in the formulation of the nonlinear function subject to minimization does not increase significantly, the topology of the solution space becomes much more nonlinear.

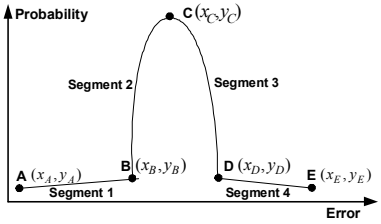


Figure 12. The selected shape of the error model and the corresponding 10 parameters. Segments 1 and 4 are linear, segments 2 and 3 are quadratic. The parameters include: the coordinates of five points except the x coordinates of points A and E (the min. and max. of the error values are known); two additional parameters for the two quadratic segments 2 and 4.

B. *Monolithic Approximation*

The second approach was monolithic piece-wise linear approximation where we assumed that the targeted unimodular model distribution can be approximated with at most L ($L = 20$ in our study) piecewise linear segments that satisfy the unimodularity constraints. The approach is subject to very mild assumptions and conceptually it is easy to introduce a new term to the OF. However, our experiments show that solving such an instance of nonlinear function is excessively difficult for the optimization solver and unsuccessful for any instances with more than 20 nodes. In all 33 data sets, the eventual location error was at least an order of magnitude larger than in any other on-line technique. Therefore, we abandoned this line of research and focused on two iterative techniques with the same goal under the identical set of assumptions.

C. *Iterative Approximation*

This technique starts with the error model approximation that has a triangular form, which is characterized by four parameters as shown in Figure 13. After each iteration, we add one additional parameter and use the previous solution as a starting point. Whenever we add a new parameter, we also allow any modification of all already-existing parameters. The procedure terminates when no improvement large than ε in the OF is observed after two consecutive additions of parameters. In our experiments, ε is set to 0.1%.

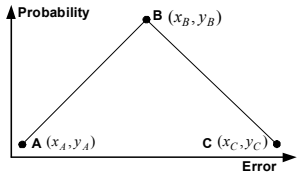


Figure 13. The initial approximation of the error model. The four parameters are the x and y coordinates of points A, B and C, except the x coordinates of A and C (which are the known min. and max. error values).

D. *Iterative Shape and SpaceApproximation*

The last technique – iterative shape and space approximation, tries to further enhance the advantages of the iterative learning technique while simultaneously reducing the run time. Again, the idea is very simple: we first divide all nodes into k partially overlapping subsets using our ILP instance selection formulation (Appendix) in such a way that the number of measurements is maximized within each subset (in our experiments each subset had at least 25 nodes and at least 8 of which were overlapping nodes). We applied the iterative learning procedure only on the first subset of data, and then refine the error model in the round-robin manner. The approach takes advantage of checking and refining the partially developed model on a small set of data where the solver is able to produce high quality solutions much faster.

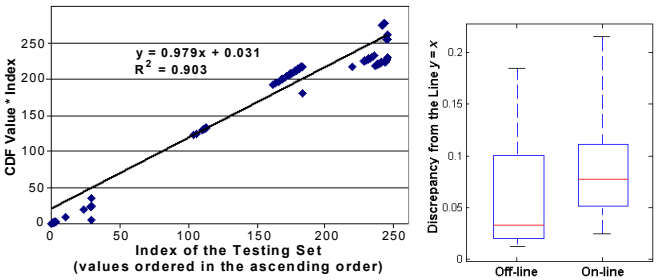


Figure 14(a). The testing set evaluation.
Figure 14(b). The boxplots of the discrepancy (from $y = x$) distribution of the model constructed off-line and on-line.

MODEL	SLOPE (M)	VARIANCE (R^2)
OFF-LINE	1.015	0.995
ON-LINE	0.979	0.903

Table 4. Comparison of the error models constructed off-line vs. on-line.

```
1.  Create token, wait for a randomly generated period of time
2.  if (no packet arrives from neighbors before waiting time expires)
3.      send all measurements and neighbor locations (if known) to the closest neighbor
4.  else {
5.      disable the ability to start LD procedure by disabling the token
6.      aggregate all its measurement and neighborhood information
7.      if (number of visited nodes >= VISIT_LIMIT) {
8.          invoke optimization solver
9.          broadcast the resultant locations }
10.     else {
11.             determine the next node
12.             send packet along with the token }
13. }
```

Figure 15. Localized LD algorithm pseudo-code.

E. *Model Evaluation and Analysis*

For the sake of brevity, we only present the results on error models constructed using parameter fitting. The results indicate that the iterative improvement method is of almost identical accuracy with somewhat larger run time. The same method discussed in Section IV.A (Figure 9(a)) is used to evaluate the on-line error model (Figure 14(a)). Figure 14(b) shows the discrepancy boxplots of the on-line model when compared to the off-line kernel-based model. Table 4 compares the slope and the variances for both models.

VI. LOCALIZED ALGORITHM FOR LOCATION DISCOVERY

We now present the localized algorithm for LD in presence of noisy measurements. There are several advantages of localized over centralized algorithms for location discovery. Some of them are well known and often advocated, e.g. lower communication and computation cost, enhanced fault

tolerance, and scalability. We found through experimentation that the localized algorithm provides a surprising advantage once the distance error is considered: improved location accuracy. Our intuition is that it is much easier for any solver to find a high quality solution to a smaller system of equations with fewer variables or to optimize an objective function that has fewer terms. Consequently, we found that it is advantageous to limit the number of nodes that are simultaneously considered for location discovery and reiterate the procedure. This is also the basis of our localized algorithm. The best suitable number of nodes is a function of the complexity of the OF used and the average number of neighbors. It ranged between 40 and 70 in our experiments given the optimization tool we have adopted.

In practice, a true localized algorithm is not only subject to local optimization, but also localized/limited measurement information. Therefore, we derived the following algorithm upon which all of our experimental results regarding the localized algorithm are based. Our assumption is that the communication range of each node is larger than the distance measurement range, as is the case with the majority of today's technologies. Figure 15 presents the pseudo-code for the localized LD algorithm.

Each node starts its own LD procedure at a random point in time unless it receives information from a neighbor. It creates a token and sends it to the closest neighbor along with the information about the distance measurements that the node has collected (lines 1-3). The procedure continues as the next node disables its ability to start the LD procedure (lines 4-6). It then sends the token and its information about the measured distances and locations of other nodes (if known) to the third node that is closest in terms of the sum of the measured distances to nodes visited by the token (lines 10-13). The data collection procedure terminates when the number of visited nodes is larger than the specified threshold (lines 7-9). In our experiments, the threshold is set to 40 nodes. The locations of all nodes with three or more neighbors are calculated. The information about locations of all the nodes is then broadcast back. If there are nodes with locations that are not calculated, they restart the mechanism for initiating the LD process at some other random moments.

VII. Experimental Results

In this section we experimentally evaluate the centralized off-line LD algorithm, the on-line LD algorithm with simultaneous error model construction, as well as the localized algorithm. All three algorithms are evaluated in situations with and without GPS devices. We conduct analysis of the LD algorithms with respect to (i) performance across different data sets; (ii) the average number of neighbors; (iii) the quality of distance measurements; and (iv) scalability. We also compare the performance of the LD algorithms with a sample of previously published algorithms. Finally, we analyze the communication cost of the centralized and the localized LD algorithms. All experiments are conducted using the acoustic signal-based distance measurements collected by the deployed sensor networks (Section III.A).

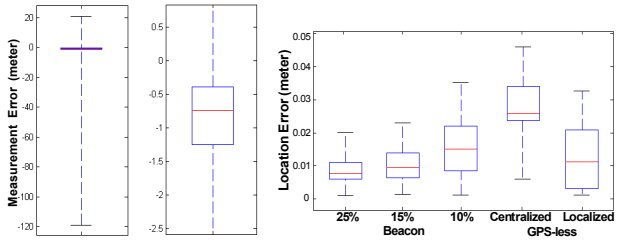


Figure 16(a) The measurement error (measured – real) boxplot.
Figure 16(b) The measurement error boxplot zoom view.
Figure16(c). The location error boxplots of the off-line GPS-base LD and localized GPS-less LD.

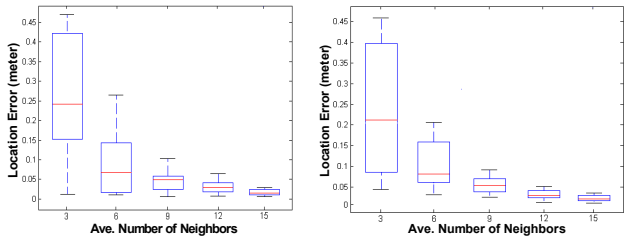


Figure 17(a). The location error boxplots given different average connectivity for off-line GPS-based LD.
Figure 17(b). The location error boxplots given different average connectivity for localized GPS-less LD.

A good way to evaluate the overall effectiveness of both the objective function and the LD algorithm is to compare the input error (the distance measurement errors) and the resultant location errors. Figures 16(a) and 16(b) present the boxplots of the distance measurement errors. The median and average of the measurement error are 6.73m and 0.74m, respectively. Figure 16(c) presents the boxplots of the location error distribution in five optimization scenarios with models constructed off-line: (i) 25% beacons; (ii) 15% beacons; (iii) 10% beacons; (iv) centralized off-line GPS-less; and (v) localized off-line GPS-less. We can see from the plot that increasing the percentage of beacons has diminishing returns, as indicated by the small improvement from having 25% of beacons compared to only 15%. The plot also shows that even in the least competitive scenario (centralized GPS-less), the maximum error is smaller than 0.05m. Another interesting observation is that the localized GPS-less LD often outperforms the centralized case with 10% beacons. In addition, as the collateral cost for switching from off-line to on-line model construction, the run time increased by a factor of almost 2 while the location accuracy of the approach deteriorated by approximately 1/3 in our experiments when compared to centralized off-line GPS-less LD. Its corresponding boxplot is partially out of the current scale range (0 – 0.05m). Therefore, we excluded it from the plot for a better visualization of the reminding five boxplots.

It is widely assumed that a high degree of connectivity in LD graph results in smaller location errors. Figures 17(a) and 17(b) show the boxplots of the resultant location errors given different average number of neighbors for centralized GPS and

localized GPS-less LD algorithms. We see that while it is important to have more than minimally required three neighbors, once the number of neighbors per node is more than 10, one can expect very little further improvement. More interestingly, the quality of the neighboring measurements matters much more than the number of neighbors. For example, Figure 18 indicates that lower median and average location errors are achieved when the number of neighbors is only 5 but all the measurements are in the range [15m, 35m] (where the measurements are the most accurate) than having 15 neighbors. There are at least two major ramifications: (i) it is often advantageous to conduct LD by considering a subset of measurements, both in terms of optimization complexity and accuracy; and (ii) more accurate locations are often calculated by considering only measurements of certain range.

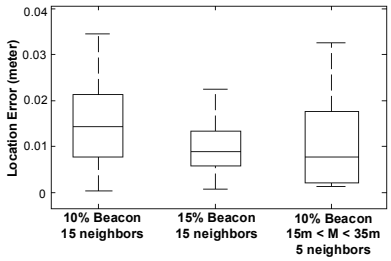


Figure 18. The location error boxplots of 10%, 15% beacons with an average connectivity of 15; and 10% beacons with an average connectivity of 5 (15m < measurements < 35m).

In the appendix we explain how we create different sizes of the location discovery problem instances using the original set of the measurements (Figure 3). Each node in the generated instance has a user-specified number of neighbors; the generated distance measurements follow the same error distribution as the original data set. All the scalability analysis is conducted on the instances created by this ILP instance generation, and we use the localized GPS-less LD approach for this study. From Figure 19(a), we observe that initially the median location error increases by a factor of 2, but it stabilizes with any further size increase. In addition, we observe that the location error distribution expands to a larger range as the network size grows, especially in the case of 1000 and 2000 nodes. This is an expected consequence of the presence of large number of nodes. Simply put, the interpretation is that some nodes have higher probability of getting 'lucky' and vice versa when the network size increases. It is interesting to observe that no instances larger than 300 nodes are solved well using the centralized algorithms: obviously we reached the limit that can be addressed using the BFSR optimization software. In Figure 19(b) we plotted the median location errors verses the network size. Also shown in the plot is the best fit of median location errors. Note that the trend is sublinear (logarithmic).

In addition to network size, we also analyze the scalability in terms of dimensions. Figure 20 shows the location error boxplots when the localization is conducted in 1-d, 2-d and 3-d spaces. It is interesting to note that in 3-d all percentiles of the location error increased by almost 45%.

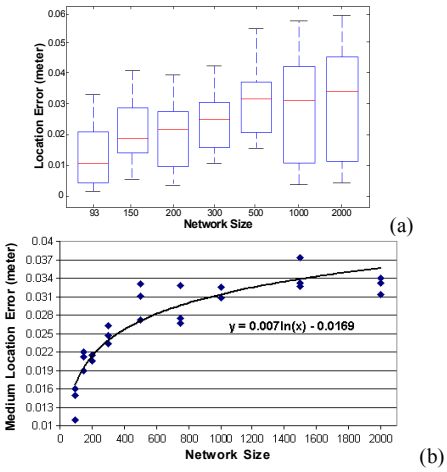


Figure 19(a). The scalability study – location error boxplots given different network sizes.
Figure 19(b). Best fitted model of the scalability in terms of network size.

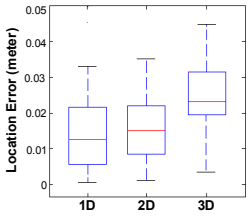


Figure 20. The scalability study – location error boxplots when LD is conducted in different dimensions.

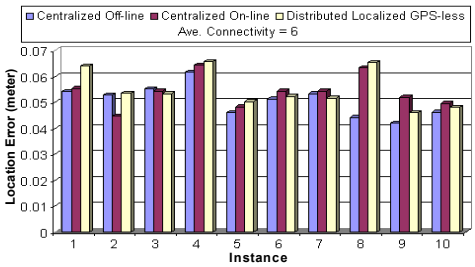


Figure 21. The median location error comparison of the centralized off-line LD, the centralized on-line LD, and the localized GPS-less LD across 10 independent data sets.

We also examine the consistency of performance of the developed LD algorithms by applying them on all 33 instances of our data set. Figure 21 shows the results for 10 randomly selected instances where the number of neighbors is on average six per node. Centralized GPS-based off-line and on-line, and the localized GPS-less on-line algorithms are evaluated. We see that the on-line algorithm, although often somewhat inferior, is essentially performing at a similar level to the off-line algorithms.

Table 5 presents the communication cost comparison between the centralized and the localized scenarios. In both cases, the calculation is done in terms of number of bytes-hops that are transmitted/received for the purpose of LD (Section VI). The precision is set to 16 bits. The localized algorithm becomes more scalable as the network size expands, indicated by the ratio between the two in the last column. The run time for instances of size 100 is usually 1-2min on a Pentium III 1.2GHz processor in the centralized scenario.

NETWORK SIZE	CENTRALIZED	LOCALIZED	RATIO (C/L)
93	56KB-hops	16KB-hops	3.50
150	114KB-hops	25KB-hops	4.56
200	175KB-hops	34KB-hops	5.14
300	324KB-hops	51KB-hops	6.35
500	698KB-hops	86KB-hops	8.11
1000	1.91MB-hops	172KB-hops	11.37
2000	5.44MB-hops	345KB-hops	16.14

Table 5. The communication cost comparison.

	AVE. CONNECTIVITY	
	12.1	9.0
ROBUST	93.75%	–
N-HOP	43.25%	58.3%
APS	40.36%	43.25%
CENTRALIZED OFF-LINE	0.089%	0.15%
LOCALIZED GPS-LESS	0.082%	0.13%

Table 6. Comparison of the normalized location errors.

Finally, we compare our LD algorithms with three previously published algorithms: (i) APS [20]; (ii) N-hop multilateration [10]; and (iii) Robust positioning [33]. Langendoen and Reijers [2] present a comprehensive performance comparison of these three approaches on a single (simulation) platform – OMNeT++ discrete event simulator [34]. The authors induced random noise that follows the Gaussian distribution in the simulation. A total of 225 sensor nodes were randomly generated in the simulations; 5% of the nodes were randomly set to be beacons. The average location errors were normalized against the measurement range. For example, 30% location error means the real and the estimated positions differ by 30% of the maximum measurement range. Under these conditions, the authors considered two different average connectivity values (average number of neighbors): 12.1 and 9.0. In order to create a similar experimental setup for the best possible comparison, we have also generated a network of same number of nodes, beacons, and connectivity, using the ILP instance generator. The measurements follow the error distribution of the original data set (Figure 3). Table 6 shows the average location error comparison of all three techniques with our centralized GPS-based on-line and localized GPS-less approaches. An average reduction in location error of approximately 1/3 is usually achieved when we apply our LD algorithms as compared to the authors' simulated data with the Gaussian distribution.

Note that the propagation characteristics of acoustic signals are related to a number of environmental factors such as the pressure (altitude), vapor in the air and the temperature as well as the terrain features. We investigate the compensation for temperature in [35]. In addition, Detailed explanations of the five model construction techniques and more comprehensive experimental results can be found in [35].

VIII. CONCLUSION

We have developed a new location discovery approach that uses statistical models to drive the objective function; and uses minimizations of a continuous nonlinear function as the optimization mechanism. The approach is evaluated using data collected from deployed sensor networks and we compared the performance with several other location discovery methods. The analysis indicates the importance of error models and that the nature of errors can be captured well by parametric (closed formula) models. We experimentally demonstrate that it is equally important to identify what to optimize (objective function) as how to optimize (optimization algorithms). The approach is generic and often performs better in a localized manner and in a GPS-free framework than in a centralized beacon-based setup. Finally, we conclude that it is often more valuable to have measurements in certain low-error ranges than to have a large degree of connectivity.

REFERENCES

[1] J. Hightower, and G. Borriello, "Location system for ubiquitous computing," IEEE Computer, vol 34, issue 8, pp. 57-66, 2001.

[2] K. Langendoen, and N. Reijers, "Distributed localization in wireless sensor networks: a quantitative comparison," Tech. Rep. PDS-2002-003, Technical University, Delft, 2002.

[3] M.A. Stephens, "EDF statistics for goodness of fit and some comparisons," Journal of the American Statistical Association, Vol. 69, pp. 730-737, 1974.

[4] G.W. Snedecor, and W.G. Cochran, Statistical methods, 8th Edition, Iowa State University Press, 1989.

[5] J.J. Filliben, "The probability plot correlation coefficient test for normality," Technometrics, pp. 111-117, 1975.

[6] B.H. Wellenhoff, H. Lichtenegger, and J. Collins, Global Positioning System: Theory and Practice, 4th Edition, Springer Verlag, 1997.

[7] P. Bahl, and V.H. Padmanabhan, "RADAR: an in-building RF-based user location and tracking system," INFOCOM, pp. 775-784, 2000.

[8] N. Patwari, and A.O. Hero III, "Using proximity and quantized RSS for sensor localization in wireless networks," WSNA, pp. 20-29, 2003.

[9] N.B. Priyantha, A. Miu, H. Balakrishnan, and S. Teller, "The cricket compass for context-aware mobile applications," MOBICOM, pp. 1-14, 2001.

[10] A. Savvides, C. Han, and M.B. Strivastava, "Dynamic fine-grained localization in ad-hoc networks of sensors," MOBICOM, pp. 166-179, 2001.

[11] D. Niculescu, and B. Nath, "Ad hoc positioning system (APS) using AoA," INFOCOM, pp. 1734-1743, 2003.

[12] P. Biswas, "Semidefinite programming for ad hoc wireless sensor network localization," IPSN, pp. 46-54, 2004.

[13] A. Galstyan, B. Krishnamachari, K. Lerman, and S. Pattern, "Distributed online localization in sensor networks using a moving target," IPSN, pp. 61-70, 2004.

[14] A. Haeberlen, E. Flannery, A.M, Ladd, A. Rudys, D.S. Wallach, and L.E. Kavraki, "Practical robust localization over large-scale 802.11 wireless networks," MOBICOM, pp. 70-84, 2004.

[15] D. Moore, J. Leonard, D. Rus and S. Teller, "Robust distributed network localization with noisy range measurements," Sensys, pp. 50-61. 2004.

[16] A. Nasipuri, and K. Li, "A directionality based location discovery scheme for wireless sensor networks, WSNA, pp. 105-111. 2002.

[17] Y. Shang, W. Ruml, and Y. Zhang, "Localization from mere connectivity," MOBIHOC, pp. 201-212, 2003.

[18] T. He, C. Huang, B.M. Blum, J.A. Stankovic, and T. Abdelzaher, "Range-free localization schemes for large scale sensor networks," MOBICOM, pp. 81-95, 2003.

[19] L. Doherty, K.S.J. Pister, and L. El Ghaoui, "Convex position estimation in wireless sensor networks," INFOCOM, pp. 1655-1663, 2001.

[20] D. Niculescu, and B. Nath, "Ad hoc positioning system (APS)," GLOBECOM, 2001.

[21] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-less low cost outdoor localization for very small devices," IEEE Personal Communications, vol 7. no 5, pp. 28-34, 2000.

[22] S. Capkun, M. Hamdi, and J.P. Hubaux, "GPS-free positioning in mobile Ad Hoc Networks," Cluster Computing Journal, vol 5. no 2, pp. 157-167, 2002.

[23] L. Hu, and D. Evens, "Localization for mobile sensor networks. MOBICOM, pp. 45-57, 2004.
[24] D. Niculescu, and B. Nath, "Error characteristics of ad hoc positioning systems (APS)," MOBIHOC, pp. 20-30, 2004.
[25] A. Savvides, W. Garber, S. Adlakha, R. Moses, and M.B. Srivastava, "On the error characteristics of multihop node localization in ad-hoc sensor networks," IPSN, pp. 317-332, 2003.
[26] L. Girod, "Development and characterization of an acoustic rangefinder," Tech. Rep. USC-CS-00-728, 2002.
[27] W. Merrill, L. Girod, J. Elson, K. Sohrabi, F. Newberg, and W. Kaiser, "Autonomous position location in distributed, embedded, wireless systems," IEEE CAS Workshop, 2002.
[28] J.B. Saxe, "Embeddability of weighted graphs in k-space is strongly NP-hard," 17th Allerton Conf. Commun. Control Comput, pp. 480-489, 1979.
[29] M.R. Garey, and D.S. Johnson, Computers and Intractability, A Guide to the Theory of NP-Completeness, W.H. Freeman and Company, New York, 2002.
[30] T.G. Szymanski, "Assembling code for machines with span-dependent instructions," Comm. ACM 21, pp. 300-308, 1978.
[31] T. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer-Verlag. New York, 2001.
[32] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling, Numerical Recipes. Cambridge, 1986.
[33] C. Savarese, K. Langendoen, and J. Rabaey, "Robust positioning algorithms for distributed ad-hoc wireless sensor networks," WSNA, pp. 112-121, 2002.
[34] A. Varga, The OMNeT++ Discrete Event Simulation System. European Simulation Multiconference, 2001.
[35] J. Feng, L. Girod, and M. Potkonjak, "Location discovery using data-driven statistical error modeling," To appear in Tech. Report, University of California, Los Angeles, 2006.

APPENDIX

ILP Formulation for Instance Selection

In this appendix we present the boolean ILP formulation which selects a portion of the original data set in order to satisfy a set of neighboring requirements, while the measurement errors still follow the same distribution as the original instance. Note that the formulation for instance generation, which is the basis for all the scalability studies, is very similar to the instance selection we are presenting. We formulate this problem as ILP because its optimality and its ability of solving large instances. All of our experiments regarding instance selection and generation are done by the commercial ILP solver CPLEX [35]. The input of the formulation is a set of constants that denote the existence of edges between all N nodes:

$$E_{ij} = \begin{cases} 1, & \text{if there exists edge between nodes } i \text{ and j} \\ 0, & \text{otherwise} \end{cases}$$

The outputs are the subset of nodes selected and the size of the subset. More specifically, the goal is to select a subset S of nodes from the original network in such a way that (i) each node in S has at least C_1 but not more than C_2 neighbors in S; (ii) the average number of neighbors in the subset is equal to C_3. The ILP formulation has two types of variables and four types of constraints:

$$x_i = \begin{cases} 1, & \text{node } i \text{ is chosen} \\ 0, & \text{otherwise} \end{cases}$$

$$L_{ij} = \begin{cases} 1, & \text{edge between nodes } i \text{ and } j \text{ is chosen} \\ 0, & \text{otherwise} \end{cases}$$

Four required types of constraints are:

1. All variables must have value either 1 or 0.

$$x_i \geq 0; \quad x_i \leq 1; \quad i = 1,\ldots,N$$
$$\forall\ E_{ij} = 1, \quad L_{ij} \geq 0; \quad L_{ij} \leq 1; \quad i = 1,\ldots,N;\ j = 1,\ldots,N$$

2. Each node in S must have at least C_1 and at most C_2 neighbors that also belong to S.

$$\sum_{j=1}^{N} \left(L_{ij} \wedge \left(E_{ij} \left(x_i \wedge y_i\right)\right)\right) - C_1 x_i \geq 0;$$

$$\sum_{j=1}^{N} \left(L_{ij} \wedge \left(E_{ij} \left(x_i \wedge y_i\right)\right)\right) - C_2 x_i \leq 0; \quad i = 1,\ldots,N$$

3. The nodes in S have an average number of neighbors close to C_3.

$$\sum_{i=1}^{N}\sum_{j=1}^{N} L_{ij} - C_3 \sum_{i=1}^{N} x_i \leq \varepsilon;$$

where ε is a small user specified discrepancy constant.

4. If an edge between nodes i and j is selected, then nodes i and j must be selected as well

$$L_{ij} - (x_i \wedge y_i) \leq 0; \quad i = 1,\ldots,N;\ j = 1,\ldots,N$$

In the second type of constraint, the term $L_{ij} \wedge E_{ij}(x_i \wedge y_i)$ specifies whether the edge E_{ij} is chosen (value 1) or not (value 0). The summation of this term over all j is therefore the number of edges (or neighbors) chosen for node i. The condition of not fewer than C_1 number of chosen neighbors is enforced by constraining the difference between the summation and ($C_1 x_i$) to be greater than or equal to zero. Similarly, constraining the difference between the summation and ($C_2 x_i$) requires that the number of chosen neighbors does not exceed C_2. The last constraint ensures that in the situation in which an edge E_{ij} is chosen, the corresponding node i and j must belong to the subset.

The objective function is to maximize the number of nodes that satisfy these conditions (i.e. the size of the selected subset):

$$\max: \sum_{i=1}^{N} x_i$$

The logical '$\wedge$' (and) operator is implemented in the following way. Consider a and b as the two operands and c is the result (i.e. $c = a \wedge b$). There are four types of constraints:

1. $a \geq 0; \quad a \leq 1; \quad b \geq 0; \quad b \leq 1; \quad c \geq 0; \quad c \leq 1;$

2. $c - a \leq 0$;
3. $c - b \leq 0$;
4. $c - (a + b) \geq -1$;

The first type of constraint forces every variable to be boolean. The second and the third type of constraint enforce c to be 0 when either a or b, or both have value 0. The forth constraint enforces c to be 1 when both a and b have value 1.