Sequence-Based Localization in Wireless Sensor Networks

Kiran Yedavalli and Bhaskar Krishnamachari

Abstract—We introduce a novel sequence-based localization technique for wireless sensor networks. We show that the localization space can be divided into distinct regions that can each be uniquely identified by sequences that represent the ranking of distances from the reference nodes to that region. For n reference nodes in the localization space, combinatorially, $O(n^n)$ sequences are possible, but we show that, due to geometric constraints, the actual number of feasible location sequences is much lower: only $O(n^4)$. Using these location sequences, we develop a localization technique that is robust to random errors due to the multipath and shadowing effects of wireless channels. Through extensive systematic simulations and a representative set of real mote experiments, we show that our lightweight localization technique provides comparable or better accuracy than other state-of-the-art radio signal strength-based localization techniques over a range of wireless channel and node deployment conditions.

Index Terms—Wireless sensor networks, localization, location sequence, arrangement of lines.

1 Introduction

Accurate localization is an essential part of many wireless sensor network applications. Over the years, many researchers have proposed many different solutions for this problem (for example, [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], and [11]). In these techniques, there is a trade-off between the accuracy of localization and the complexity of implementation. For instance, least squares estimation techniques (see [1]) require accurate radio frequency (RF) channel parameters such as the radio path loss exponent. Fingerprinting-based techniques (such as [8]) require extensive preconfiguration studies that depend on the features of the localization space. Other techniques require specialized hardware (see [5]) or a complex configuration procedure (see [11]). On the other extreme, really simple techniques such as computing the centroid of nearby beacons (see [7]) provide low accuracy. In this paper, we present a novel sequence-based RF localization technique that is lightweight, works with any hardware, and provides accurate localization without requiring accurate channel parameters or any preconfiguration.

At the heart of our proposed technique is the division of a 2D localization space into distinct regions by the perpendicular bisectors of lines joining pairs of *reference nodes* (nodes with known locations). We show that each distinct region formed in this manner can be uniquely identified by a *location sequence* that represents the distance ranks of reference nodes to that region. We present an algorithm to construct the *location sequence table* that maps all these feasible location sequences to the corresponding regions by using the locations of the reference nodes. This table is used to localize an *unknown node* (that is, the node whose location has to be determined) as follows.

The unknown node first determines its own location sequence based on the measured strength of signals between itself and the reference nodes. It then searches through the location sequence table to determine the "nearest" feasible sequence to its own measured sequence. The centroid of the corresponding region is taken to be its location.

In this paper, we focus only on RF-signal-based localization since radios are used for the essential task of communication and are therefore freely available on all devices in a wireless network. Ideally, the measured distance order of the reference nodes should be identical to the distance order based on true euclidean distances. However, this is not true in the real world, as the RF signals are subjected to multipath fading and noise. These nonideal effects corrupt the location sequence measured by the unknown node. For n reference nodes in the localization space, the possible number of combinations of distance rank sequences is $O(n^n)$. However, we prove in this paper that the actual number of feasible location sequences is much lower due to geometric constraints, that is, only $O(n^4)$. The lower dimensionality of the sequence table enables the correction of errors in the measured sequence. This is one of the reasons that our proposed sequence-based localization (SBL) technique performs well despite channel errors.

The rest of the paper is organized as follows: We formally define location sequences in Section 2 and describe the procedure of localization using them in Section 3. In the same section, we derive the maximum number of feasible location sequences, illustrate the construction of the location sequence table, discuss the effect of RF channel nonidealities on unknown node location sequences, and describe metrics to measure the "distance" between sequences. In Section 4, we describe localization procedures for two different application scenarios and show their robustness to RF channel random errors through examples. In Section 5, we present an exhaustive systematic performance study of

- *K. Yedavalli is with Cisco Systems Inc., 170 West Tasman Drive, San Jose, CA 95134. E-mail: kyedaval@cisco.com.*
- *B. Krishnamachari is with the Department of Electrical Engineering Systems, University of Southern California, 3740 McClintock Avenue, EEB 300, Los Angeles, CA 90089. E-mail: bkrishna@usc.edu.*

Manuscript received 21 Feb. 2006; revised 17 Apr. 2007; accepted 24 Apr. 2007; published online 14 May 2007.
For information on obtaining reprints of this article, please send e-mail to: tmc@computer.org, and reference IEEECS Log Number TMC-0054-0206.
Digital Object Identifier no. 10.1109/TMC.2007.1076.

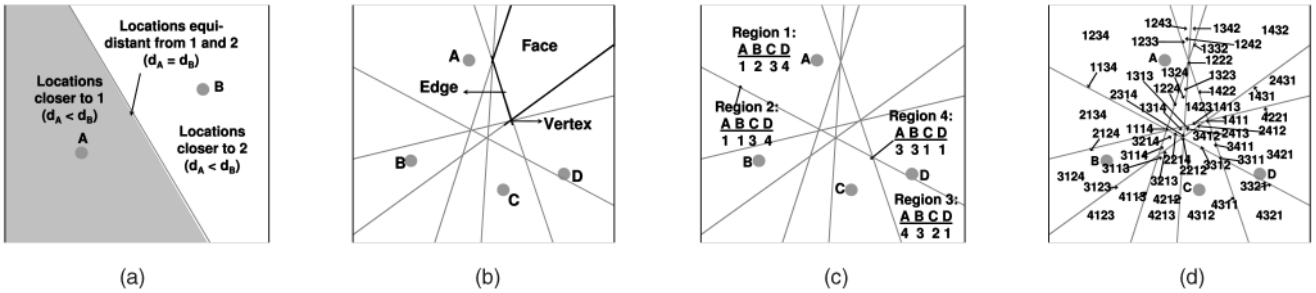

Fig. 1. (a) The perpendicular bisector of the line joining two reference nodes divides the localization space into three distinct regions. (b) Illustration of arrangement of six bisector lines for four reference nodes placed uniformly and randomly in a square localization space. (c) Examples of location sequences for a four-reference-node topology. (d) All feasible location sequences for the topology in (c).

our localization technique, in addition to conducting a comparative study with state-of-the-art localization techniques. We present the evaluation of our technique in real mote experiments in Section 6 and discuss related work in Section 7. We conclude and discuss our future work in Section 8.

2 Location Sequences

In this section, we define location sequences and illustrate them through examples.

Assume that a 2D localization space consists of n reference nodes. Consider any two reference nodes and draw a perpendicular bisector to the line joining their locations. This perpendicular bisector divides the localization space into three different regions that are distinguished by their proximity to either reference nodes, as illustrated in Fig. 1a. Similarly, if perpendicular bisectors are drawn for all $\frac{n(n-1)}{2}$ pairs of reference nodes, they divide the localization space in to many regions of three different types: *vertices*, *edges*, and *faces*, as shown in Fig. 1b. This subdivision of a 2D space into vertices, edges, and faces by a set of lines is an *arrangement* induced by that set [12].

Now, for each region created by the arrangement induced by the set of perpendicular bisectors, determine the ordered sequence of the reference nodes' ranks based on their distances from them. We define this ordered sequence of distance ranks as the *location sequence*.

Proposition 1. *The location sequence of a given region is unique to that region.*

Proof. The proof is by contradiction. Assume that two different regions in the arrangement have the same location sequence. This implies that the distance ranks of reference nodes are the same for both regions. This further implies that there is no bisector line that separates the two regions. The implication applies to all possible combinations of regions such as two faces, two edges, two vertices, a face and an edge, an edge and a vertex, and a face and a vertex in their own different ways. Otherwise, if there was a bisector line of two arbitrary reference nodes that separated the two regions, then it would rank those reference nodes differently for the two regions. However, this is a contradiction as, by definition, two different regions in the arrangement are separated by at least a single bisector line. □

Therefore, each region created by the arrangement has a unique location sequence. Further, we make the following observations:

- All locations inside a region have the same location sequence.
- If each region in the arrangement is represented by its centroid, then there is a one-to-one mapping between a location sequence and the centroid of the region that it represents. For a vertex, the centroid is the vertex itself. For an edge, the centroid is its midpoint, and for a face, the centroid is the centroid of the polygon that bounds it.
- The total number of unique location sequences is equal to the sum of the number of vertices, the number of edges, and the number of faces created by the arrangement in the localization space.

The order in which the ranks of reference nodes are written in a location sequence is determined by a predefined order of reference node IDs. We illustrate the above ideas through examples. Fig. 1c shows the location sequences of four different regions. In the example, the predefined order of reference node IDs is ABCD. Region 1 is a face, and its location sequence is 1234, since the distance rank of A from it is 1 (A is the closest), and the respective distance ranks of B,C and D are 2,3 and 4 (D is the farthest). Similarly, for Region 3, the location sequence is 4321, as the distance rank of A is the farthest (distance rank 4), D is the closest (distance rank 1), and B is closer than C and A. For Region 4, which is a vertex, the distance ranks of A,B and C,D are equal in pairs as it lies on the intersection of perpendicular bisectors of those pairs of reference nodes. Also, the pair C,D is closer to it than the pair A,B. Therefore, its location sequence is 3311. Similarly, for Region 2, which is an edge, the distance ranks of A and B are the same, and the location sequence is 1134. Fig. 1d shows all feasible location sequences for the topology of the reference nodes in Fig. 1c.

Next, we describe how location sequences can be used for localization.

3 Localization Using Location Sequences

The procedure for localization of unknown nodes using location sequences is given as follows:

1. Determine all feasible location sequences in the localization space and list them in a *location sequence table*.

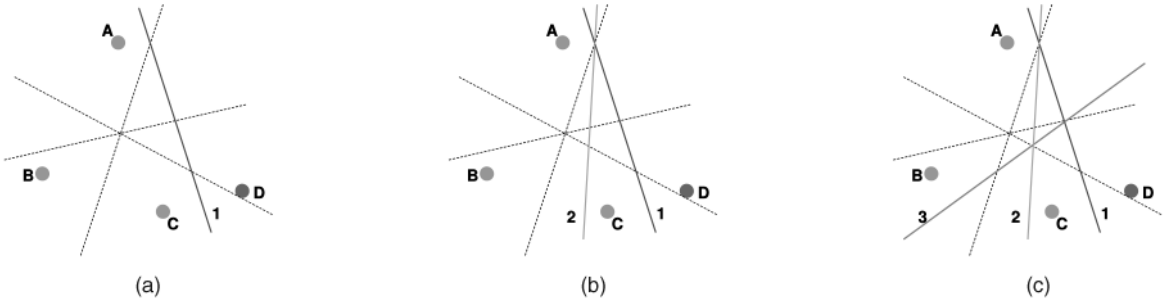

Fig. 2. Addition of the fourth reference node D adds three new bisector lines to the localization space. (a) The first of the three new bisector lines, that is, line 1, which is the perpendicular bisector of CD, creates three new vertices (equal to the number of preexisting lines in the localization space), four new faces, and seven new edges at most. (b) The second line, line 2, which is the perpendicular bisector of BD, has to pass through the intersection point of the bisectors of CD and BC because $\{BD, CD, BC\}$ form a triangle, and the perpendicular bisectors of the three sides of a triangle intersect at a single point. Therefore, line 2 creates two new vertices, four new faces, and six new edges at most. (c) Similarly, line 3, which is the perpendicular bisector of AD, has to pass through the intersection points of the perpendicular bisectors of AB, BD, AC, and CD, as $\{AD, AB, BD\}$ and $\{AD, AC, CD\}$ are two triangles with a common side AD. Therefore, line 3 creates one new vertex, four new faces, and five new edges at most.

2. Determine the location sequence of the unknown node location by using received signal strength (RSS) measurements of localization packets exchanged between itself and the reference nodes. The RSS-based location sequence will be a corrupted version of the original location sequence.
3. Search in the location sequence table for the "nearest" location sequence to the unknown node location sequence. The centroid mapped to by that sequence is the location estimate of the unknown node.

The above procedure opens itself to the following questions: How many feasible location sequences are there in a 2D localization space? How can we get them? How do random errors in RSS measurements affect the unknown node location sequence? What is the meaning of "nearest" location sequence and how do we measure distances between location sequences?

In the rest of this section, we answer the above questions. We begin by determining the maximum number of feasible location sequences in the localization space.

3.1 Maximum Number of Location Sequences

For n reference nodes in the localization space, the number of possible combination sequences of distance ranks is $O(n^n)$. However, we show that the actual number of feasible location sequences is much lower, which is in the order of $O(n^4)$ at worst.

As stated previously, the number of feasible location sequences is equal to the sum of the number of vertices, edges, and faces created by the arrangement induced by the perpendicular bisectors of reference nodes. Therefore, its upper bound can be obtained by determining the maximum number of such vertices, edges, and faces, given the locations of the reference nodes. In [12], the authors show that the maximum number of vertices, edges, and faces for an arrangement induced by n lines is $\frac{n(n-1)}{2}$, n^2, and $\frac{n^2}{2}+\frac{n}{2}+1$, respectively. Using these results, for $\frac{n(n-1)}{2}$ perpendicular bisectors of n reference nodes,

1. the number of vertices is at most $\frac{n^4}{8}-\frac{n^3}{4}-\frac{n^2}{8}+\frac{n}{4}$,
2. the number of edges is at most $\frac{n^4}{4}-\frac{n^3}{2}+\frac{n^2}{4}$, and
3. the number of faces is at most $\frac{n^4}{8}-\frac{n^3}{4}+\frac{3n^2}{8}-\frac{n}{4}+1$.

Owing to the properties of perpendicular bisectors, it is possible to derive tighter upper bounds on the number of vertices, edges, and faces.

Theorem 1. *Let L be the set of bisector lines for n reference nodes $|L|=\frac{n(n-1)}{2}$. Let $A(L)$ be the arrangement induced by L. Then,*

1. *the number of vertices of $A(L)$ is at most $\frac{n^4}{8}-\frac{7n^3}{12}+\frac{7n^2}{8}-\frac{5n}{12}$,*
2. *the number of edges of $A(L)$ is at most $\frac{n^4}{4}-n^3+\frac{7n^2}{4}-n$, and*
3. *the number of faces of $A(L)$ is at most $\frac{n^4}{8}-\frac{5n^3}{12}+\frac{7n^2}{8}-\frac{7n}{12}+1$.*

Proof. We make use of the property that the perpendicular bisectors of the sides of a triangle intersect at a single point. Assume that $(i-1)$ reference nodes have already been added, implying that the localization space already has $\frac{(i-1)(i-2)}{2}$ bisector lines. When the ith reference node is added, $(i-1)$ new bisector lines are added to the localization space.

Vertices. The first of the $(i-1)$ bisector lines intersects the already present lines in at most $\frac{(i-1)(i-2)}{2}$ new vertices. The second new line is the perpendicular bisector of a side of the triangle in which the first new line is also a perpendicular bisector. Therefore, the second new line has to pass through at least one of the vertices created by the first new line, thus creating at most $\frac{(i-1)(i-2)}{2}-1$ new vertices. Similarly, the third new line creates at most $\frac{(i-1)(i-2)}{2}-2$ new vertices. This is illustrated in Fig. 2 for $n=4$. Finally, the $(i-1)$th new line creates at most $\frac{(i-1)(i-2)}{2}-(i-2)$ new vertices. Therefore, the total number of new vertices added by the ith reference node is at most

$$\frac{(i-1)(i-2)}{2}+\frac{(i-1)(i-2)}{2}-1+\frac{(i-1)(i-2)}{2}-2+\cdots+\frac{(i-1)(i-2)}{2}-(i-2)=\frac{(i-1)(i-2)^2}{2}. \quad (1)$$

The maximum number of vertices for $n = 3$ is 1. Therefore, for n reference nodes, the maximum number of vertices is

$$1 + \sum_{i=4}^{n} \frac{(i-1)(i-2)^2}{2} = \frac{n^4}{8} - \frac{7n^3}{12} + \frac{7n^2}{8} - \frac{5n}{12}. \quad (2)$$

Edges. As explained previously, the first new line intersects the already present lines in at most $\frac{(i-1)(i-2)}{2}$ vertices and creates at most $\frac{(i-1)(i-2)}{2} + 1$ new edges on the new line and at most $\frac{(i-1)(i-2)}{2}$ new edges on the old lines, which add up to $\frac{(i-1)(i-2)}{2} \cdot 2 + 1$ new edges at most. Since the second new line passes through at least one of the vertices created by the first new line, it creates at most $\frac{(i-1)(i-2)}{2} + 1$ new edges on the second new line, and it creates at most $\frac{(i-1)(i-2)}{2} - 1$ new edges on the old lines, including the first new line. This adds up to at most $\frac{(i-1)(i-2)}{2} \cdot 2$ new edges in the localization space. This trend is again illustrated in Fig. 2 for four reference nodes in the localization space. Finally, the $(i-1)$th new line adds $\frac{(i-1)(i-2)}{2} \cdot 2 - (i-3)$ new edges to the localization space. Therefore, the total number of new edges added by the ith reference node is at most

$$\frac{(i-1)(i-2)}{2} \cdot 2 + 1 + \frac{(i-1)(i-2)}{2} \cdot 2 + \frac{(i-1)(i-2)}{2} \cdot 2 - 1 + \cdots + \frac{(i-1)(i-2)}{2} \cdot 2 - (i-3) = i^3 - \frac{9i^2}{2} + \frac{15i}{2} - 4. \quad (3)$$

The maximum number of edges for $n = 3$ is 6. Therefore, for n reference nodes, the maximum number of edges is

$$6 + \sum_{i=4}^{n} \left[i^3 - \frac{9i^2}{2} + \frac{15i}{2} - 4 \right] = \frac{n^4}{4} - n^3 + \frac{7n^2}{4} - n. \quad (4)$$

Faces. The number of new faces created by a new line is equal to the number of edges on the new line. Therefore, the number of new faces created by the first new line among the $(i-1)$ new lines is at most $\frac{(i-1)(i-2)}{2} + 1$. Since the second new line has to pass through one of the intersection points of the first line, it would also create $\frac{(i-1)(i-2)}{2} + 1$ new faces and this trend continues for all the $(i-1)$ new lines as illustrated in Fig. 2. Therefore, the total number of new faces added by the ith reference node is at most

$$(i-1)\left(\frac{(i-1)(i-2)}{2} + 1 \right). \quad (5)$$

The localization space has one face when $n = 1$. Therefore, for n reference nodes, the maximum number of faces in the localization space is given by

$$1 + \sum_{i=2}^{n} (i-1)\left(\frac{(i-1)(i-2)}{2} + 1 \right) = \frac{n^4}{8} - \frac{5n^3}{12} + \frac{7n^2}{8} - \frac{7n}{12} + 1. \quad (6)$$

□

Corollary 1. *The maximum number of unique location sequences due to n reference nodes is $\frac{n^4}{2} - 2n^3 + \frac{7n^2}{2} - 2n + 1$.*

Proof. The maximum number of unique location sequences is the sum of the maximum number of vertices, edges, and faces due to n reference nodes, as derived in Theorem 1:

$$\left(\frac{n^4}{8} - \frac{7n^3}{12} + \frac{7n^2}{8} - \frac{5n}{12} \right) + \left(\frac{n^4}{4} - n^3 + \frac{7n^2}{4} - n \right) + \left(\frac{n^4}{8} - \frac{5n^3}{12} + \frac{7n^2}{8} - \frac{7n}{12} + 1 \right) = \frac{n^4}{2} - 2n^3 + \frac{7n^2}{2} - 2n + 1. \quad (7)$$

□

Next, we illustrate how we can obtain all these feasible location sequences in the localization space and store them in the location sequence table.

3.2 Location Sequence Table Construction

Below, we present the pseudocode for an algorithm that constructs the location sequence table, given the locations of the reference nodes and the boundaries of the localization space:[1]

Algorithm 1: CONSTRUCTLOCATIONSEQUENCETABLE.
Input:
1) Location coordinates of reference nodes $\{(ax_i, ay_i) | i = 0 \rightarrow n-1\}$.
2) Boundaries of the localization space B.
Output: Location Sequence Table.

```
0   L = {l_i | i = 0 → (n(n−1)/2 − 1)} ←
BISECTORLINES({(ax_i, ay_i) | i = 0 → n − 1}, B)
1   (FL, EL, VL) ← CONSTRUCTARRANGEMENT(L)
▷Get vertex sequences.
2   for i ← 0 to (|VL| − 1)
3     Centroid[i] ← VL[i]
4     Sequence[i] ← GETSEQUENCE(Centroid[i])
5   end for
▷Get edge sequences.
6   for i ← |VL| to (|VL| + |EL| − 1)
7     Centroid[i] ← GETEDGECENTROID(EL[i])
8     Sequence[i] ← GETSEQUENCE(Centroid[i])
9   end for
▷Get face sequences.
10  for i ← (|VL| + |EL|) to (|VL| + |EL| + |FL| − 1)
11    Centroid[i] ← GETFACECENTROID(FL[i])
12    Sequence[i] ← GETSEQUENCE(Centroid[i])
13  end for
▷Return the location sequence table
14  return {Sequence, Centroid}
```

- BISECTORLINES takes in the locations of the reference nodes and the boundaries of the localization space as input and returns the set L of all pairwise perpendicular bisector lines within the boundaries of the localization space. Each line is represented by the

1. C++ code files that construct the arrangement of lines and the location sequence table are available for download at http://anrg.usc.edu/downloads.html.

intersection points on the left and right boundaries of the localization space.

- CONSTRUCTARRANGEMENT constructs the arrangement, given a set of lines as input, and returns a doubly connected edge list (EL) that consists of a vertex list (VL), an *EL*, and a face list (FL). Please refer to [12] (Section 8.3) for a detailed description of this algorithm.
- *VL* contains pointers to all vertices of the arrangement induced by the set L.
- *EL* contains pointers to all edges of the arrangement induced by the set L.
- *FL* contains pointers to all faces of the arrangement induced by the set L.
- GETEDGECENTROID takes in an edge pointer as the input and returns the centroid of the edge. The centroid of an edge (c_x, c_y) is its midpoint, given by

$$(c_x, c_y) \leftarrow \left(\frac{o_x + d_x}{2}, \frac{o_y + d_y}{2}\right), \tag{8}$$

where (o_x, o_y) and (d_x, d_y) are the origin and destination vertices of the edge.

- GETFACECENTROID takes in a face pointer as the input and returns the centroid of the face. The centroid of a face (c_x, c_y), given its vertices $\{(x_i, y_i) | 0 \leq i \leq p-1\}$, is calculated as follows:

$$c_x \leftarrow \frac{1}{6A}\sum_{i=0}^{p-1}(x_i + x_{i+1})(x_i y_{i+1} - x_{i+1} y_i), \tag{9}$$

$$c_y \leftarrow \frac{1}{6A}\sum_{i=0}^{p-1}(y_i + y_{i+1})(x_i y_{i+1} - x_{i+1} y_i), \tag{10}$$

where p is the number of vertices that bound a given face and A is its area given by

$$A \leftarrow \frac{1}{2}\sum_{i=0}^{p-1}(x_i y_{i+1} - x_{i+1} y_i)\ ;\ (x_p, y_p) = (x_0, y_0). \tag{11}$$

- GETSEQUENCE takes in the coordinates of a point in the localization space and returns the location sequence for that point with respect to the locations of the reference nodes.

Theorem 2. *Algorithm 1 takes $O(n^5 \log(n))$ worst case time and $O(n^5)$ worst case space to construct the location sequence table.*

Proof. The function BISECTORLINES in line 0 takes $O(n^2)$ time and space. The algorithm CONSTRUCTARRANGEMENT that constructs the arrangement of lines takes $O(n^4)$ time, which is optimal, as proven in [12, Theorems 8.5 and 8.6]. Since this algorithm returns the *VL*, the *EL*, and the *FL*, it requires $O(n^4)$ space to store all the three lists. The functions GETFACECENTROID and GETEDGECENTROID in lines 3 and 7, respectively, take $O(1)$ time and space each. The function GETSEQUENCE involves sorting n reference nodes based on their distances from the centroid of the region in consideration. This takes $O(n \log n)$ time and $O(n)$ space. Since the number of faces, edges, and vertices is $O(n^4)$, the worst case time requirement for lines 2-13 in the above algorithm is $O(n^5 \log(n))$ and the worst case space requirement is $O(n^5)$. Therefore, in total, Algorithm 1 takes $O(n^5 \log(n))$ worst case time and $O(n^5)$ worst case space to construct the location sequence table. □

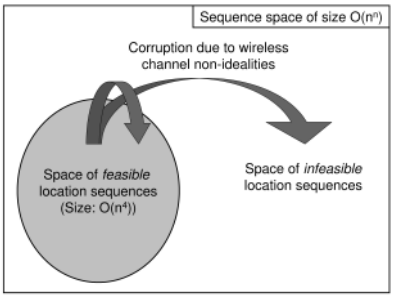

Fig. 3. RF channel nonidealities could corrupt a location sequence from the feasible space either to another sequence in the feasible space or to a sequence in the infeasible space.

Next, we discuss the effect of RF channel random errors on the unknown node location sequence.

3.3 Unknown Node Location Sequence

The unknown node determines its location sequence by using RSS measurements of RF localization packets exchanged between itself and the reference nodes. The RSS measurements are subjected to random errors due to RF channel nonidealities such as multipath and shadowing. In the absence of such nonidealities, the RSS measurements accurately represent the distances between the unknown node and the reference nodes. If the reference nodes are ranked in a decreasing order of these RSS values, then this order represents the increasing order of their separation from the unknown node.

This is not true in reality. Reference nodes that are farther from the unknown node might measure higher RSS values than reference nodes that are closer. If the reference nodes are ranked on their respective RSS measurements, then the location sequence formed by these ranks will be a corrupted version of the original sequence. Corruption in an unknown node location sequence results in an erroneous estimation of its location. In the ideal case, when there is no corruption, the unknown node location would be the centroid of the region represented by its location sequence. However, corruption in its location sequence could erroneously estimate its location to be the centroid of some other region.

For example, if the ranks of reference nodes C and D are interchanged because of corruption due to RF channel nonidealities for Region 1 in Fig. 1c, then the new location sequence would be 1243 instead of 1234. Moreover, 1243 represents a region that is adjacent to the original region, as shown in Fig. 1d.

3.4 Feasible and Infeasible Sequences

As discussed previously, combinatorially, n reference nodes produce $O(n^n)$ location sequences. However, as shown in the previous section, a localization space with n reference nodes has only $O(n^4)$ distinct regions and, consequently,

only $O(n^4)$ *feasible* location sequences in the worst case. For given reference node locations, the location sequence table includes all feasible location sequences. All other sequences are *infeasible*. The nonidealities of the RF channel could corrupt a feasible location sequence either to another feasible sequence or to an infeasible sequence, as illustrated in Fig. 3. If the corrupted sequence is infeasible, then it would be possible to detect the corruption in the sequence, whereas, if the corrupted sequence is feasible, then corruption detection is not possible.

Here, we would like to emphasize the importance of low density of location sequences compared to the full sequence space. The low density of location sequences implies that many infeasible sequences are mapped to a single feasible sequence, and this, in turn, could provide robustness to location estimation against RF channel nonidealities.

Next, we present metrics to measure the distance between two location sequences.

3.5 Distance Metrics

The distance between two location sequences is essentially the difference in rank orders of different reference nodes. Fortunately, statistics [13] offers two metrics that capture this difference in rank orders: *Spearman's Rank Order Correlation Coefficient* and *Kendall's Tau*.

Given two location sequences $U = \{u_i\}$ and $V = \{v_i\}$, $1 \leq i \leq n$, where u_i and v_i are the ranks of reference nodes, the above two metrics are defined as follows:

1. *Spearman's Rank Order Correlation Coefficient* [13]. It is defined as the linear correlation coefficient of the ranks and is given by

$$\rho = 1 - \frac{6\sum_{i=1}^{n}(u_i - v_i)^2}{n(n^2-1)}. \tag{12}$$

2. *Kendall's Tau* [13]. In contrast to Spearman's coefficient, in which the correlation of exact ranks is calculated, this metric calculates the correlation between the relative ordering of ranks of the two sequences. It compares all the $\frac{n(n-1)}{2}$ possible pairs of ranks (u_i, v_i) and (u_j, v_j) to determine the number of matching and nonmatching pairs. A pair is matching or concordant if $u_i > u_j \Rightarrow v_i > v_j$ or $u_i < u_j \Rightarrow v_i < v_j$ and nonmatching or discordant if $u_i > u_j \Rightarrow v_i < v_j$ or $u_i < u_j \Rightarrow v_i > v_j$. The correlation between the two sequences is calculated as follows:

$$\tau = \frac{(n_c - n_d)}{\sqrt{n_c + n_d + n_{tu}}\sqrt{n_c + n_d + n_{tv}}}, \tag{13}$$

where n_c is the number of concordant pairs, n_d is the number of discordant pairs, n_{tu} is the number of ties in u, and n_{tv} is the number of ties in v.

The range of both ρ and τ is $[-1, 1]$. Next, we describe the procedure to determine the locations of unknown nodes by using their location sequences.

3.6 Location Determination

The location of the unknown node is determined as follows:

1. Calculate distances between the unknown node location sequence and all location sequences in the location sequence table by using the above distance metrics.
2. Choose the centroid represented by the location sequence that is closest to the unknown node location sequence as its location estimate.

Mathematically,

$$LocationEstimate = Centroid(arg \min_{1 \leq i \leq O(n^4)} \tau_i), \tag{14}$$

where τ_i is the Kendall's Tau or Spearman's correlation between the unknown node location sequence and the ith location sequence in the location sequence table.

Due to RF channel nonidealities, the unknown node location sequence could be a feasible sequence different from its uncorrupted version or an infeasible sequence. In any case, the above procedure maps it to the centroid of the nearest feasible location sequence in the location sequence table that represents a different region in the arrangement than the original uncorrupted version.

We measure the amount of corruption in the unknown node location sequence by calculating its distance from the uncorrupted version, using the above metrics, and denote it by T. We denote the distance between the corrupted unknown node location sequence and the nearest feasible sequence in the location sequence table by τ.

Calculating the Spearman's coefficient and Kendall's Tau between two sequences are $O(n)$ and $O(n^2)$ operations, respectively. Since the location sequence table is of size $O(n^4)$, searching through it takes $O(n^5)$ and $O(n^6)$ operations, respectively, for the above two metrics. Later in the paper, in Section 5, we compare the performance of the two distance metrics in terms of error in the unknown node location estimate.

4 Localization Scenarios

In this section, we illustrate two localization procedures for two different scenarios that are determined by the localization space size:

1. *The entire localization space is within the radio range of the unknown node*. In this case, the location sequence table remains constant for all locations of the unknown node in the localization space. Therefore, the localization procedure is given as follows:
   - Preconstruct and store the location sequence table by using the locations of the reference nodes.
   - When the unknown node initiates the localization process by broadcasting a localization packet, provide the stored location sequence table along with the RSS measurements from the reference nodes.
   - The unknown node determines its location sequence by using the RSS measurements and determines its location by searching through the provided location sequence table for the nearest feasible location sequence.

   Here, the time cost incurred by the unknown node to estimate its location is equal to the sum of the time to determine its location sequence, which is an $O(n \log n)$ operation, and the time to search through

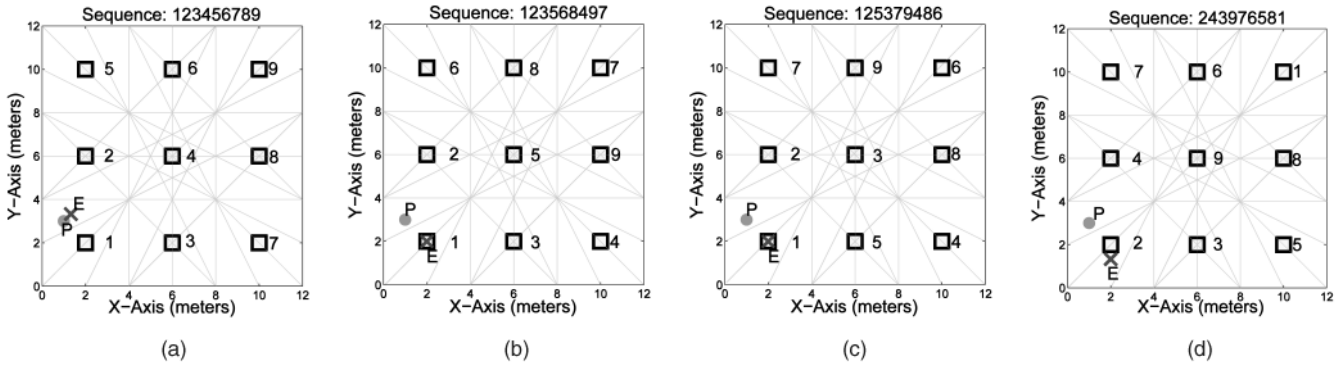

Fig. 4. *Robustness examples*: location estimate (E) for the unknown node (P) at (1,3) for a grid layout of nine reference nodes. The number adjacent to a reference node is its corresponding rank. The location error is expressed in meters, where the side length of the square localization area is 12 m. (a) $(T = 1, \tau = 1)$, Estimate (E): (1.33, 1.33), and Location Error: 0.46 m. (b) $(T = 0.722, \tau = 0.783)$, Estimate (E): (2.0, 2.0), and Location Error: 1.4 m. (c) $(T = 0.556, \tau = 0.667)$, Estimate (E): (2.0, 2.0), and Location Error: 1.4 m. (d) $(T = 0.111, \tau = 0.278)$, Estimate (E): (2.0, 1.33), and Location Error: 1.94 m.

the location sequence table, which is a $O(n^6)$ operation. The amount of memory space required is on the order of $O(n^5)$ bytes.

2. *The localization space is much larger than the radio range of the unknown node*. In this case, the location sequence table changes with the location of the unknown node as a different set of reference nodes are encountered at each location. Therefore, the localization procedure is given as follows:

   - The unknown node collects the locations and RSS measurements of the reference nodes in its radio range.
   - It constructs the location sequence table by using Algorithm 1 and the locations of the reference nodes and calculates its location sequence by using the RSS measurements.
   - It determines its location by searching for the nearest sequence in the location sequence table.

   In this case, the time cost incurred by the unknown node to estimate its location is equal to the sum of the time to calculate its location sequence, which is an $O(n \log n)$ operation, the time to construct the location sequence table, which is an $O(n^5 \log n)$ operation, and the time to search through it, which is a $O(n^6)$ operation. The memory requirement is $O(n^5)$ in this case also.

A wireless device that is typically used as an unknown node is of the form factor of an iPAQ [14] (that can communicate with the reference node devices, usually of the form factor of Berkeley MICA 2 motes [15]) which typically has a 300-MHz processor and 128-Mbyte RAM. In real-application scenarios, a typical value for the number of reference nodes (n) is less than 15, after which there is only a very marginal gain in location accuracy of the unknown node. Therefore, for a typical value of $n = 10$ reference nodes, the time and space requirements for the unknown node to construct the location sequence table are approximately 0.3 ms and 32 Kbytes, respectively. Also, the time required to search through it is approximately 0.45 ms. Thus, including the associated overhead, the total localization time taken by SBL is in milliseconds in typical application scenarios, which is very efficient. Next, we illustrate the robustness of our localization technique to RF channel nonidealities through some examples.

4.1 Examples

Fig. 4 shows a sample layout of nine reference nodes placed in a grid and a single unknown node (P). Fig. 4a plots the location estimate (E) for the ideal case when there are no erroneous ranks; that is, the location sequence is uncorrupted, or $T = 1$. In these examples, we use Kendall's Tau to measure the distance between sequences. Figs. 4b, 4c, and 4d show the location estimates for increasing corruption in unknown node location sequences. The location estimate error increases with increasing corruption or decreasing correlation T between the RSS location sequence and the true location sequence of P. These examples suggest that SBL is robust to multipath and shadowing effects of the RF channel up to some level. Intuitively, the three main reasons to which this robustness can be attributed to are

1. the low density $O(n^4)$ of the location sequence space relative to the entire sequence space of $O(n^n)$,
2. the inherent redundancy of comparing $\frac{n(n-1)}{2}$ rank pairs in calculating the distance between two sequences by using Kendall's Tau, and
3. the rank order in the location sequence of the unknown node due to two reference nodes with path losses PL_i and PL_j, which is robust to random errors in them up to a tolerance level of $|PL_i - PL_j|$.

5 Evaluation

In this section, we present a complete performance evaluation of SBL. First, we discuss its inherent location error characteristics, and then, by using simulations, we study its performance as a function of the RF channel and node deployment parameters. We also present a comparative study with three other state-of-the-art localization techniques.

5.1 Location Error Characteristics

In the location sequence table, each location sequence maps to the centroid of the region that it represents. Representing all locations in a region by its centroid comes at the cost of

TABLE 1
Typical Values and Ranges for Different Simulation Parameters

Parameter	Typical Value	Typical Range
P_T	4dBm (max.)	NA
$PL(d_0)$	55dB ($d_0 = 1$m) [18]	NA
η	4 (indoors) 4 (outdoors)	1 – 7 [17]
σ	7 (indoors) 4 (outdoors)	2 – 14 [17]
n	10	3 – 10
β	0.1 (one node in 10 sq.m)	$\{0.01, 0.04, 0.1, 1\}$

error in the location estimate of the location sequence. For face regions (in which the unknown node is more likely to be located), the location error is of the order of the square root of the area of the face. Since the number of faces for n reference nodes is $O(n^4)$, the average face area varies, which is proportional to $\frac{1}{n^4}$. Therefore, the average location estimate error for locations in a face region reduces, which is proportional to n^2.

Apart from the above location errors, the performance of SBL is affected by random errors in RSS measurements due to multipath and shadowing effects of the RF channel. In the rest of this section, we present results from simulation studies that capture the effect of these random errors on the performance of SBL.

5.2 Simulation Model

The most widely used simulation model to generate RSS samples as a function of distance in RF channels is the log-normal shadowing model [16]:

$$P_R(d) = P_T - PL(d_0) - 10\eta \log_{10} \frac{d}{d_0 + X_\sigma}, \qquad (15)$$

where P_R is the received signal power, P_T is the transmit power, and $PL(d_0)$ is the path loss for a reference distance of d_0. η is the path loss exponent, and the random variation in RSS is expressed as a Gaussian random variable of zero mean and σ^2 variance $X_\sigma = N(0, \sigma^2)$. All powers are in dBm, and all distances are in meters. In this model, we do not provision separately for any obstructions like walls. If obstructions are to be considered, then an extra constant needs to be subtracted from the right-hand side of the above equation to account for the attenuation in them (the constant depends on the type and number of obstructions).

5.3 Simulation Parameters

The accuracy of RF-based localization techniques depends on a number of parameters. Chief among these is the accuracy of RSS measurements. In an ideal world, in which RSS values are not affected by multipath fading effects, they represent true distances between nodes, which can lead to very accurate localization of unknown nodes. The ideal world is represented by $\sigma = 0$ in (15). However, in the real world, RSS values are corrupted by multipath fading effects. This has a profound influence on the accuracy of RF localization techniques. According to the above propagation model, RSS values are defined by η and σ values for the given environment. Since every RF environment can be characterized by η and σ values (see [17] and [18]), it is necessary to study the accuracy of RF localization techniques as a function of these two parameters.

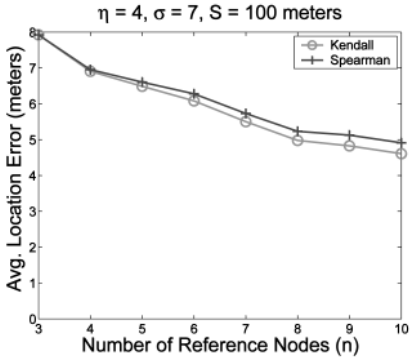

Fig. 5. Average location error as measured using Spearman's correlation and Kendall's Tau as a function of the number of reference nodes.

In addition, the density and number of reference nodes available to the unknown node has a significant influence on the number of reference nodes (see [2], [6], and so forth). Thus, the location estimate of any RF-based localization technique depends on a fundamental set of parameters, which can be broadly categorized as follows:

1. **RF Channel Characteristics** [17], [16]:
   - Path loss exponent (η): measures the power attenuation of RF signals relative to distance.
   - Standard deviation (σ): measures the standard deviation in RSS measurements due to log-normal shadowing.

   The values of η and σ change with the frequency of operation and the obstructions and disturbance in the environment.
2. **Node Deployment Parameters**:
   - Number of reference nodes (n) and
   - Reference node density (β).

Table 1 lists the typical values and ranges for different parameters used in our simulations.

5.4 Simulation Procedure

We assume that all reference nodes are in the radio range of each other and also that of the unknown node. A 48-bit arithmetic linear congruential pseudorandom number generator was used, and results were averaged over 100 random trials. In each trial, n reference nodes were placed uniformly and randomly in a square localization space of size $S \times S$ square meters, and the unknown node was placed at 100 different locations on a grid of $\frac{S}{10}$ separation. In total, the results presented are averaged over 10,000 different scenarios. In our simulations, we use $S = 100$ m.

The performance of SBL is measured in terms of *location error* for a wide range of RF channel conditions and node deployment parameters. Location error is defined as the euclidean distance between the location estimate and the actual location of the unknown node. The location error is averaged over 100 random trials, as described previously.

Fig. 5 plots the two distance metrics described in the previous section as a function of the number of reference nodes (n) or, in other words, the length of the location sequence. There is a growing difference, however small, between the two metrics with increasing length of the

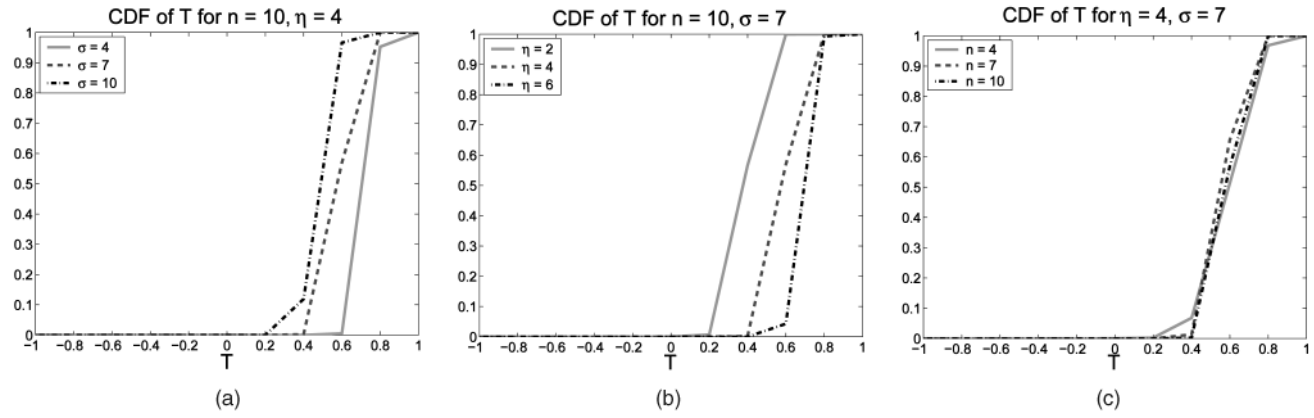

Fig. 6. *Sequence corruption*: cumulative distribution function of *Kendall's Tau* T between the RSS location sequence and the true location sequence for varying (a) standard deviation (σ), (b) path loss exponent (η), and (c) number of reference nodes (n).

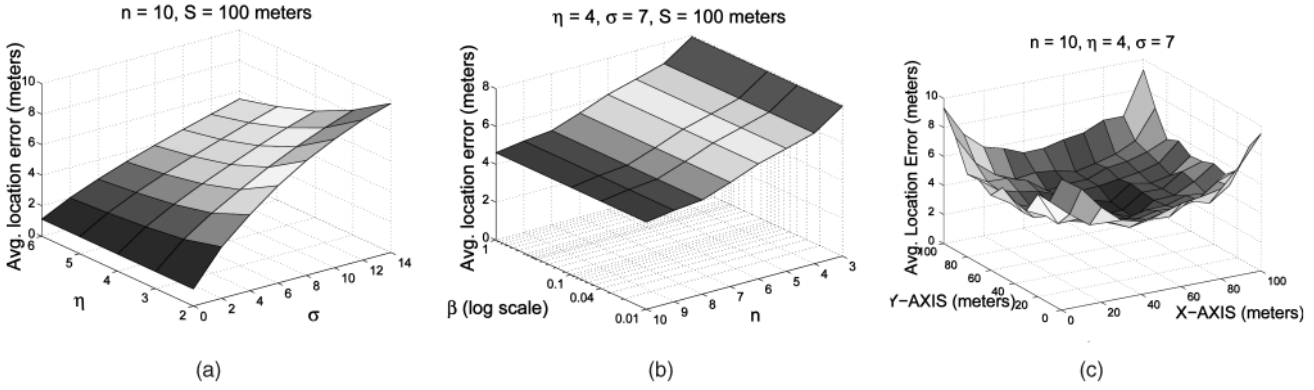

Fig. 7. *Performance*. (a) Average location error as a function of RF channel parameters: standard deviation (σ) and path loss exponent (η). (b) Average location error as a function of node deployment parameters: number of reference nodes (n) and reference node density (β). (c) Average location error as a function of the location of the unknown node.

sequence, with Kendall's Tau performing increasingly better than Spearman's correlation in terms of the location estimate error.

5.5 Simulation Results: Sequence Corruption

Fig. 6 plots the corruption in location sequences, represented by T, due to RF channel and node deployment parameters. According to these results, the corruption in location sequences

- increases with increasing randomness in the RF channel represented by standard deviation in RSS σ (Fig. 6a),
- decreases with increasing path loss exponent η (Fig. 6b), and
- is independent of the number of reference nodes in the localization space n (Fig. 6c).

5.6 Simulation Results: Performance Study

Fig. 7 plots the average location error due to SBL as a function of RF channel and node deployment parameters. The main results are as follows:

- Location error due to SBL is higher for RF channels with higher standard deviation (σ) values (Fig. 7a). This is due to higher levels of corruption in location sequences at higher values of σ.
- Location error due to SBL is lower for RF channels with higher path loss exponent (η) values (Fig. 7b). This is due to lower levels of corruption in location sequences at higher η values.
- Location error due to SBL reduces with increasing number of reference nodes (n), suggesting that longer sequences are more robust to RF channel nonidealities than shorter sequences (Fig. 7b).
- Location error due to SBL reduces with increasing reference node density β, as shown in Fig. 7b.
- Location error due to SBL depends on the location of the unknown node. Fig. 7c plots the average location error for all possible unknown node locations in the localization space. It shows that unknown node locations that are closer to the center of the localization space have lower location error than unknown node locations closer to the boundaries of the localization space. This can be verified from the observation (for example, Fig. 1b) that, for any arrangement of bisector lines, the faces and edges toward the center of the localization space have smaller areas and lengths, respectively, compared to that of at its boundaries. Consequently, for unknown node locations toward the center of the localization space, the location to which the nearest feasible sequence of the corrupted sequence maps will be closer to the true location of the unknown node than

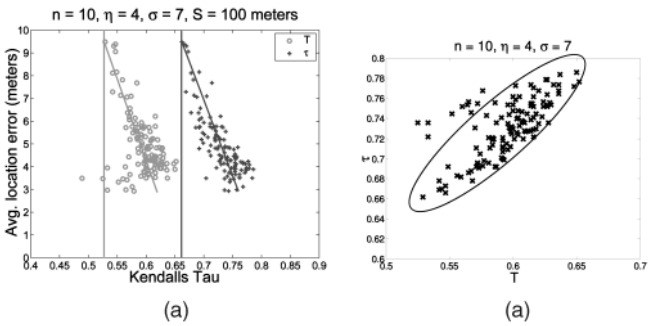

Fig. 8. (a) Average location error as a function of the sequence corruption (T) and as a function of the distance (τ) between the corrupted sequence and its nearest feasible sequence in the location sequence table. (b) Correlation between τ and T.

for locations toward the boundaries. This results in lower location errors for unknown node locations toward the center of the localization space than for locations toward its boundaries.

- Fig. 8a plots the average location error as a function of Kendall's Tau values T and τ, and Fig. 8b plots τ as a function of T. The figures suggest the following:
  - The location error is correlated to T, which is the corruption due to RF channel.
  - The location error is correlated to τ, which is the distance between the corrupted sequence and the nearest feasible sequence.
  - A correlation exists between τ and T.

  This suggests that τ, which is a measurable quantity, as opposed to T, could be used as a quantitative indicator of the location error due to SBL. Also, owing to its correlation to T, it could also be used as an approximate indicator of the state of the RF channel.

5.7 Simulation Results: Comparative Study

We compare SBL with three other comparable (see [19]) state-of-the-art localization techniques: *least squares estimator (LSE)*, *proximity localization*, and *three centroid*.

- *LSE*. It is identical to the maximum likelihood location estimator [1], [2] and works as follows:
  - Measure the distance between each of the reference nodes and the unknown node by using

$$d_{mi} = 10^{\frac{P_T - PL(d_0) - \overline{P_{Ri}}}{10\eta}}, \tag{16}$$

    where d_{mi} is the measured distance and $\overline{P_{Ri}}$ is the mean received signal power between a given reference node i and the unknown node. Accurate distance measurement requires accurate estimation of the path loss exponent (η) of the environment. This requires expensive ranging techniques and/or extensive preconfiguration surveys of the localization space.
  - For each grid point location in the localization space, determine the sum of the squares of differences in the measured distances and the true euclidean distances of all the reference nodes from the grid point:

$$\Sigma_{(x,y)} = \sum_{i=0}^{n-1} (d_i^{(x,y)} - d_{mi})^2, \tag{17}$$

    where $d_i^{(x,y)}$ is the euclidean distance between the grid location (x, y) and the reference node i.
  - Choose the grid point location with the least value of the above sum $\Sigma_{(x,y)}$ as the location of the unknown node. In our study, we consider a grid resolution (r) that is 100 times higher than the dimensions of the localization space; that is, for a localization space of $S \times S$ square meters, we search $r^2 = 10,000$ grid points, with a separation of $\frac{S}{100}$ meters between them, to determine the location of the unknown node.
- *Proximity localization*. The location of the closest reference node by RSS value is chosen as the location of the unknown node. This is an extreme special case of SBL, in which the sequence is of length 1.
- *Three centroid*. The centroid of all the reference nodes in the radio range of the unknown node is chosen as its location [7]. Since, in our case, all reference nodes are in the radio range of the unknown node, the location error would be independent of the RF channel characteristics. In order to measure the effect of these characteristics on the centroid technique, we choose the centroid of the closest three reference nodes by RSS values as the location of the unknown node.

Fig. 9 plots the average location error due to SBL, LSE, proximity localization, and three centroid as a function of the standard deviation in RSS log-normal distribution σ for different values of path loss exponents η and for different values of number of reference nodes n. The main results of the comparison are the following:

- SBL performs better than proximity localization and three centroid over a range of RF channel and node deployment parameters.
- SBL performs better than LSE for higher values of σ, whereas LSE performs better than SBL for lower values of σ. There is a crossover value of σ between the error due to SBL and LSE, and this value of σ is higher for environments that have more attenuation, that is, higher values of path loss exponent η. There is no significant change in the value of crossover σ with changing number of reference nodes n.
- For lower values of σ, the location error due to SBL decreases faster than location error due to LSE for increasing values of n. This can be seen in Figs. 9a, 9b, and 9c, in which the difference between the location error due to SBL and LSE reduces with increasing values of n.
- LSE is outperformed by all other localization techniques after some value of σ, and this value is the lowest for SBL.

It should be noted that, in the above simulations, LSE operates at a considerable advantage over other techniques as the exact value of the path loss exponent η is known. This

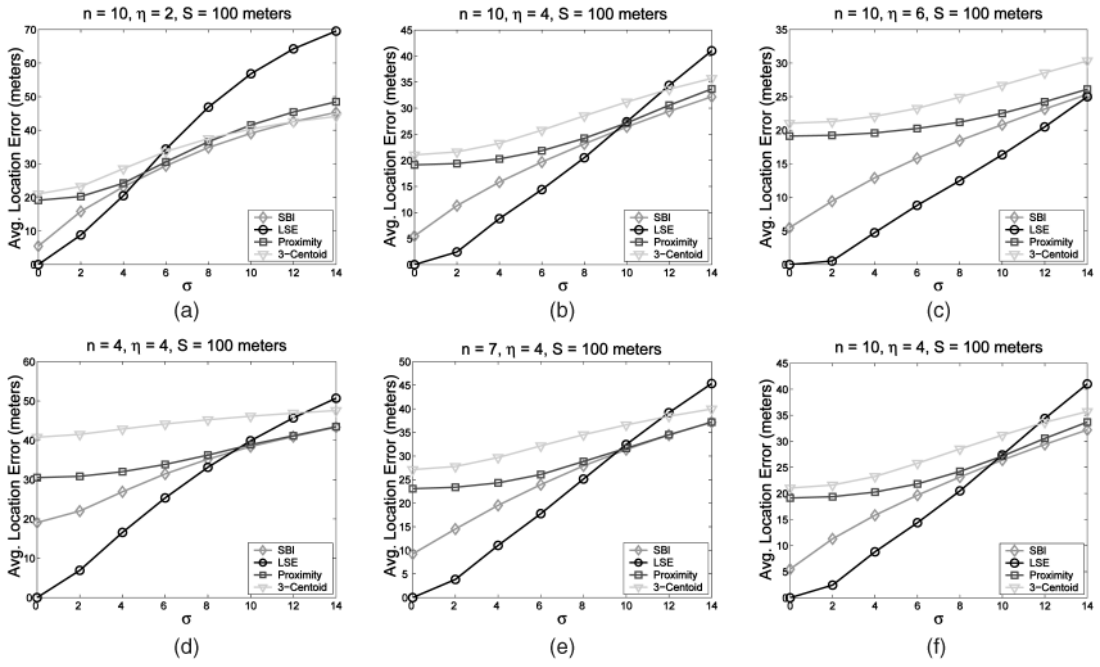

Fig. 9. *Comparison*: average location error due to SBL, LSE, proximity localization, and three centroid as a function of the standard deviation of RSS log-normal distribution σ for different values of path loss exponent η ((a) $\eta = 2$, and $n = 10$, (b) $\eta = 4$, and $n = 10$, and (c) $\eta = 6$, and $n = 10$) and for different values of number of reference nodes n ((a) $n = 4$, and $\eta = 4$, (b) $n = 7$, and $\eta = 4$, and (c) $n = 10$, and $\eta = 4$).

advantage vanishes in real-world scenarios, where the value of η is very difficult to estimate accurately, owing to its dependence on the area features such as walls, furniture, and so forth. Thus, LSE may not perform as well in real-world scenarios. Table 2 compares the time and space complexities of SBL with that of the other three localization techniques. We believe that the efficiency of SBL can be increased significantly by using more efficient location sequence table search algorithms as opposed to a naive search.

6 Real-World Experiments

The performance of SBL in real systems is studied through two experiments, representing different RF channel and node deployment parameters, and conducted using Berkeley MICA 2 motes [15]. The first experiment was conducted in a parking lot, which represents a relatively obstruction-free RF channel, and the second experiment was conducted in an office building with many rooms and furniture, which represents a typical indoor environment. For comparison, the locations of the unknown nodes were also estimated using the three localization techniques—*LSE*, *proximity localization*, and *three centroid*—described in the previous section.

TABLE 2
Comparison of Worst Case Computational Complexities of SBL, LSE, Proximity Localization, and Three Centroid

	SBL	LSE	Proximity	3-Centroid
Time	$O(n^6)$	$O(nr^2)$	$O(n \log n)$	$O(n \log n)$
Space	$O(n^5)$	$O(r^2)$	$O(n)$	$O(n)$

6.1 Outdoor Experiment: Parking Lot

The RF channel in an outdoor parking lot represents a class of relatively obstruction-free channels. As shown in Fig. 10, 11 MICA 2 motes were placed randomly on the ground. All motes were in the line of sight of each other, and all of them were programmed to broadcast a single packet without interfering with each other.[2] The motes recorded the RSS values of the received packets and stored them in their electrically erasable programmable read-only memory (Eeprom), which were later used offline for location estimation.

The locations of all the motes were estimated and compared with their true locations. Since all motes were in the radio range of each other, each mote had 10 reference nodes. For the LSE method, to estimate the distances between the motes, the RSS model described by (15) in Section 5.2 was used, as there were no obstructions between the motes in this experiment. The performance of the LSE technique depends on the value of the path loss exponent η for the area in which the experiment was conducted. For this experiment, we used the true distances and the corresponding RSS values between the reference nodes and the unknown node to estimate the value of η. Fig. 10a plots the RSS values as a function of distance. Linear regression analysis applied to the RSS versus distance data gives its slope as -2.9, implying that $\eta = 2.9$. We used this value of η to evaluate the LSE technique.

Fig. 10b compares the true mote locations with SBL location estimates for all the motes. The figure also shows the arrangement induced by the perpendicular bisectors

2. We had actually measured the RSS of 100 packets in 1 minute and observed that their standard deviation was less than 0.5 dBm. Therefore, we decided to use only a single packet for localization. In real-application scenarios, this would help in conserving energy at the mote and reducing the delay in localization without affecting its accuracy.

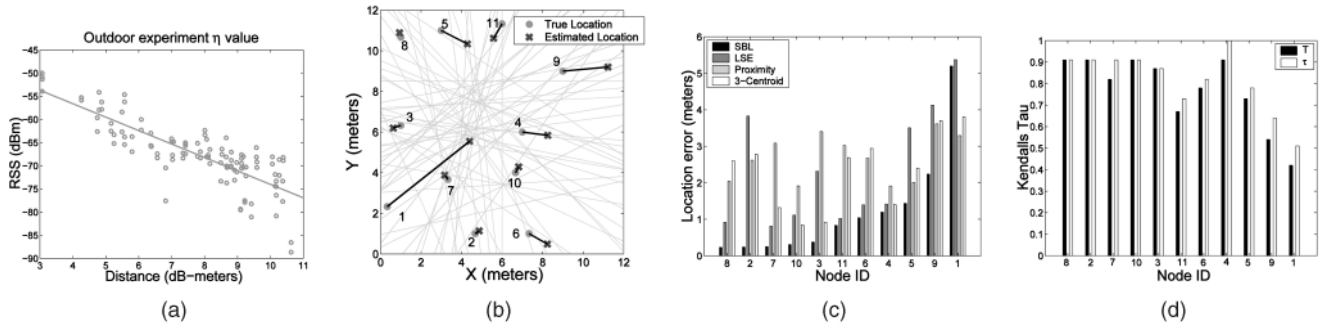

Fig. 10. *Outdoor experiment*: 11 MICA 2 motes placed randomly in an area of 144 square meters were used as reference nodes and unknown nodes. Consequently, each unknown node had 10 reference nodes. (a) Path loss exponent calculation $\eta = 2.9$. (b) Comparison between true locations and SBL location estimates. (c) Location error due to SBL, LSE, proximity localization, and three centroid (the nodes are ordered in increasing error of SBL). (d) Corruption measure T and error indicator τ.

between all pairs of reference nodes. Fig. 10c plots the absolute error in meters at each mote location due to all the four techniques. Evidently, SBL performs better than proximity localization and three centroid in 10 out of 11 cases, and it performs better than LSE in all the 11 cases.

Fig. 10d plots the sequence corruption (T) at each mote location and the distance (τ) between the corrupted sequence and the nearest feasible sequence in the location sequence table for all the 11 nodes. The correlation between T and τ can be clearly seen in the figure. Comparing Fig. 10c and Fig. 10d, broad correlations between T and the location error and between τ and the location error can be observed for SBL. For example, the location error is the highest for node IDs 1 and 9, in that order, and τ is the lowest for the same node IDs in the same order. Also, the location error is almost equal for nodes 8, 2, 7 and 10. This trend is also reflected in the values of τ for those nodes.

Table 3 shows the average absolute location error in meters, with varying numbers of reference nodes considered by the unknown node. For each node in the experiment, n reference nodes were chosen in turn from the 10 available reference nodes. Thus, for n reference nodes, the location error is averaged over $\binom{10}{n}$ values. The table shows that, in an outdoor environment, SBL performs better than the other three localization techniques, irrespective of the number of reference nodes.

6.2 Indoor Experiment: Office Building

Office buildings with features such as rooms, corridors, furniture, and other obstructions represent a distinct class of RF channels. We placed 12 MICA 2 motes (reference nodes) on the ground randomly in a corner of the Electrical Engineering building at the University of Southern California (USC), spanning different rooms and corridors. Fig. 11 shows a schematic of the experimental setup. In this experiment, an unknown node was placed at five different locations, and these locations were estimated using all the 12 motes as reference nodes. As in the outdoor experiment, the unknown node was programmed to broadcast a single packet from each location, and the reference nodes recorded the RSS values of this packet in their respective Eeprom, which were later used offline for location estimation.

TABLE 3
Comparison of Average Location Error in Meters for the Outdoor Experiment

n	SBL	LSE	Proximity	3-Centroid
5	2.355236	2.796348	3.594085	3.924164
7	1.751180	2.534984	3.130865	3.050054
9	1.322921	2.410194	2.844499	2.525028
10	1.224741	2.381832	2.743362	2.336667

Unlike in the outdoor experiment, not all motes were in the line of sight of each other, even though they were in each other's radio range. A subset of the motes had obstructions in between them in the form of walls. As for the outdoor experiment, for the LSE method, the value of η was calculated using linear regression analysis for RSS versus distance values between the reference nodes and the unknown node. Fig. 11a shows the data. In this case, the value of η is 2.2.

Fig. 11b compares the SBL location estimates of the five unknown node locations with their true locations. It can be seen that the path of the location estimates closely follows the true path of the unknown node. Fig. 11c plots the location estimate error due to SBL, LSE, proximity localization, and three-centroid techniques for each unknown node location. It can be observed that SBL performs better than LSE and three centroid in four out of five cases and better than proximity localization in two out of five cases. The reason that proximity localization is performing well is the presence of reference nodes in close proximity to each location of the unknown node.

Fig. 11d plots the sequence corruption (T) at each mote location and the distance (τ) between the corrupted sequence and the nearest feasible sequence in the location sequence table for all the five unknown node locations. Comparing this figure and Fig. 10d shows that sequences are more corrupted in the indoor experiment than the outdoor experiment, which was expected. Also, as in the outdoor experiment, there is a clear correlation between T and τ for the indoor experiment. However, the correlations between T and location error and between τ and location error are not as clear as that in the outdoor experiment.

Table 4 shows the average absolute location error in meters, with varying numbers of reference nodes considered by the unknown node. Similar to the outdoor experiment, for each node in the experiment, n reference nodes were chosen in turn from the 12 available reference nodes and, thus, for n reference nodes, the location error is averaged over $\binom{12}{n}$ values. The table shows that, in this

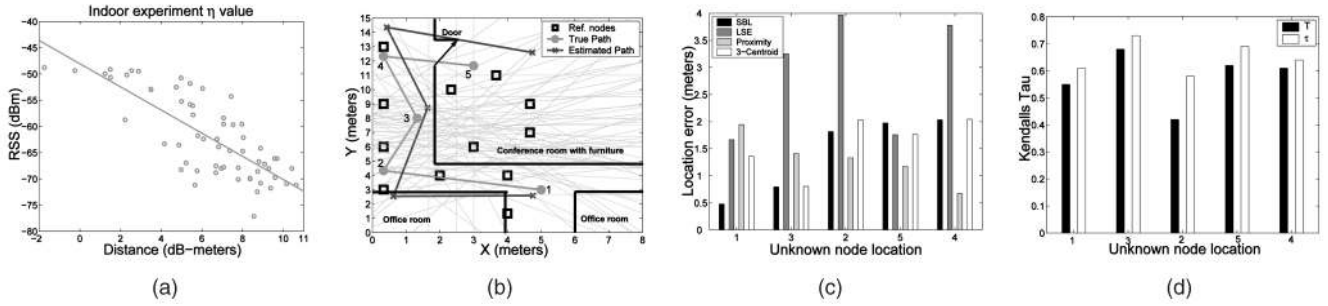

Fig. 11. *Indoor experiment*: 12 MICA 2 motes placed randomly in an area of 120 square meters were used as reference nodes. The location of the unknown node was estimated for five different locations using the 12 reference nodes. (a) Path loss exponent calculation $\eta = 2.2$. (b) Comparison between true path and SBL estimated path. (c) Location error due to SBL, LSE, proximity localization, and three centroid (the nodes are ordered in increasing error of SBL). (d) Corruption measure T and error indicator τ.

scenario, SBL outperforms LSE for all the cases and that proximity localization (which is an extreme special case of SBL) performs better than SBL. As mentioned previously, the reason for this is that the choice of unknown node locations is such that there is a reference node in close proximity to each such location.

6.3 Discussion

Experimental results show that localization techniques are more accurate for relatively clutter-free RF channel environments (outdoors with line of sight) than RF channels with many obstructions (indoor environment). Also, the performance of LSE in real-world scenarios is worse than in simulations, as conjectured in Section 5.7. This is mainly because the radio propagation model of (15) is an approximate model, and the location estimate accuracy for the LSE technique depends heavily on the accuracy of the η estimate. The RSS measurements in the experiments depend on antenna orientations, antenna height, and transmitter/receiver nondeterminism. For simulations, these issues can be captured within the log-normal random term in (15).

7 Related Work

In an earlier work [19], we presented a novel localization algorithm called *Ecolocation* that uses *location constraints* for robust localization. A location constraint is a relationship between the distances of two reference nodes from the unknown node that determines its proximity to either reference nodes, as shown in Fig. 1a. Location constraints can be graphically represented by perpendicular bisectors between reference nodes (Section 2), and each location sequence can be written as a set of location constraints. Thus, the location constraint set is also unique to each region in the arrangement.

TABLE 4
Comparison of Average Location Error in Meters for the Indoor Experiment

n	SBL	LSE	Proximity	3-Centroid
5	3.381912	3.522072	2.421350	3.232579
7	2.718227	3.249378	1.936757	2.545885
9	2.194005	3.118476	1.620071	2.008185
11	1.792685	3.012694	1.392431	1.671318
12	1.417431	2.882787	1.404000	2.111980

In this localization algorithm, the unknown node determines its set of location constraints by using RSS measurements and estimates its location by searching through grid points in the localization space to determine the grid point with the highest number of matched location constraints. In [19], we show that this is an $O(\frac{n^2S^2}{r^2})$ time operation, where S is the side of the square localization space and r is the resolution of grid points. Thus, the localization algorithm using location constraints is dependent on the localization space size, the resolution of the grid points, and the number of reference nodes. In contrast, the localization algorithm using location sequences depends only on the number of reference nodes, albeit at higher time cost of $O(n^6)$. In fact, constraint-based localization results tend to SBL ones for very high values of grid point resolution r. The cost differences suggest that, for smaller localization spaces and lower location accuracy requirements, a constraint-based localization is better compared to a sequence-based one, whereas the reverse is true for bigger localization spaces and higher location accuracy requirements.

In related works, Chakrabarty et al. [9] and Ray et al. [10] use identity codes to determine the location of sensor nodes in grid and nongrid sensor fields, respectively. Here, each grid point or region in the localization space is identified by a unique set of reference node IDs whose signals can reach the point or region, and this unique set is an identity code for that point or region. The two main drawbacks of this approach are that 1) in order to uniquely identify all unknown node locations in the localization space, the reference nodes need to be placed carefully according to rules determined by an optimization algorithm, and that 2) for acceptable location accuracies, the number of reference nodes required is prohibitively expensive, and for sparse networks of reference nodes, the accuracy is coarse grained in the order of radio range. For example, the number of reference nodes required to uniquely identify the location of an unknown node using identity codes is $O(p^m)$, where m is the number of dimensions of the localization space, and p is the number of grid points per dimension [9].

In another related work, He et al. [6] propose an RF-based localization technique in which the unknown node location is determined by the intersection of all triangles, formed by reference nodes, that are likely to bound it. The unknown node determines its existence inside a triangle by comparing its measured RSS values to that of its neighbors to detect a trend in RSS values in any particular direction.

This technique depends on the weak assumption that signal strength decreases monotonically with distance, which is not true in real-world scenarios.

8 Conclusion and Future Work

In this paper, we presented a simple and novel localization technique based on location sequences called SBL. In SBL, location sequences are used to uniquely identify distinct regions in the localization space. The location of the unknown node is estimated by first determining its location sequence using RSS measurements of RF signals between the unknown node and the reference nodes and then searching through a predetermined list of all feasible location sequences in the localization space, called the location sequence table, to find the region represented by the "nearest" one. In this chapter, we derived expressions for the maximum number of location sequences and presented an algorithm to construct the location sequence table. We described distance metrics that measure the distance between location sequences and used them to determine the corruption in location sequences due to RF channel nonidealities. We identified an approximate indicator of the extent of location estimation error by using the same distance metrics. Through examples, we demonstrated the robustness of SBL to RF channel nonidealities. Through exhaustive simulations and systematic real mote experiments, we evaluated the performance of our localization system and presented a comparison with other state-of-the-art localization techniques for different RF channel and node deployment parameters. Results showed that SBL performs well and better than other state-of-the-art localization techniques in both indoor and outdoor environments.

As part of future work, we would like to incorporate location probability into the location sequence table. Owing to the features and topology of objects and obstructions in the localization space, unknown nodes are more likely to be in some locations than others. This could be incorporated into SBL by weighing feasible location sequences in the location sequence table in proportion to the location likelihoods of the regions that they represent.

Acknowledgments

The authors wish to thank Abtin Keshavarzian of Bosch Research, Palo Alto, California, and Professor Isaac Cohen of the Computer Science Department, University of Southern California for valuable discussions on the subject.

References

[1] N. Patwari and A.O. Hero III, "Using Proximity and Quantized RSS for Sensor Localization in Wireless Networks," *Proc. Second Int'l Workshop Wireless Sensor Networks and Applications (WSNA '03)*, Sept. 2003.

[2] K. Yedavalli, "Location Determination Using IEEE 802.llb," master's thesis, Univ. of Colorado, Boulder, Dec. 2002.

[3] A. Savvides, H. Park, and M. Srivastava, "The Bits and Flops of the N-Hop Multilateration Primitive for Node Localization Problems," *Proc. First Int'l Workshop Wireless Sensor Networks and Applications (WSNA '02)*, Sept. 2002.

[4] A. Savvides, C. Han, and M. Srivastava, "Dynamic Fine Grained Localization in Ad Hoc Sensor Networks," *Proc. ACM MobiCom*, pp. 166-179, July 2001.

[5] N.B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The Cricket Location-Support System," *Proc. ACM MobiCom*, Aug. 2000.

[6] T. He, B.B.C. Huang, J. Stankovic, and T. Abdelzaher, "Range-Free Localization Schemes for Large Scale Sensor Networks," *Proc. ACM MobiCom*, Sept. 2003.

[7] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-Less Low-Cost Outdoor Localization for Very Small Devices," *IEEE Personal Comm. Magazine*, Oct. 2000.

[8] V. Bahl and V.N. Padmanabhan, "RADAR: An In-Building RF-Based User Location and Tracking System," *Proc. IEEE INFOCOM*, 2000.

[9] K. Chakrabarty, S.S. Iyengar, H. Qi, and E. Cho, "Grid Coverage for Surveillance and Target Location in Distributed Sensor Networks," *IEEE Trans. Computers*, vol. 51, no. 12, pp. 1448-1453, Dec. 2002.

[10] S. Ray, D. Starobinski, A. Trachtenberg, and R. Ungrangsi, "Robust Location Detection with Sensor Networks," *IEEE J. Selected Areas in Comm.*, special issue on fundamental performance limits of wireless sensor networks, vol. 22, no. 6, pp. 1016-1025, Aug. 2004.

[11] M. Maroti, P. Volgyesi, S. Dora, B. Kusy, A. Nadas, A. Ledeczi, G. Balogh, and K. Molnar, "Radio Interferometric Geolocation," *Proc. Third Int'l Conf. Embedded Ntworked Sensor Systems (SenSys '05)*, pp. 1-12, 2005.

[12] M. de Berg, M. van Krevald, M. Overmars, and O. Schwarzkopf, *Computational Geometry—Algorithms and Applications*, second ed. Springer, 2000.

[13] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling, *Numerical Recipes in C: The Art of Scientific Computing*, second ed. Cambridge Univ. Press, 1992.

[14] http://welcome.hp.com/country/us/en/prodserv/handheld.html, 2007.

[15] http://www.xbow.com/Products/productsdetails.aspx?sid=72, 2007.

[16] T.S. Rappaport, *Wireless Communications: Principles and Practice.* Prentice Hall, 1999.

[17] H. Hashemi, "The Indoor Radio Propagation Channel," *Proc. IEEE*, vol. 81, no. 7, pp. 943-968, July 1993.

[18] M. Zuniga and B. Krishnamachari, "Analyzing the Transitional Region in Low Power Wireless Links," *Proc. First IEEE Int'l Conf. Sensor and Ad Hoc Comm. and Networks (SECON '04)*, Oct. 2004.

[19] K. Yedavalli, B. Krishnamachari, S. Ravula, and B. Srinivasan, "Ecolocation: A Sequence Based Technique for RF Localization in Wireless Sensor Networks," *Proc. Fourth Int'l Conf. Information Processing in Sensor Networks (IPSN '05)*, Apr. 2005.

Kiran Yedavalli received the BTech degree in electrical engineering from the Indian Institute of Technology, Madras, in 2000, the MS degree from the University of Colorado, Boulder, in 2002, and the PhD degree from the University of Southern California, Los Angeles, in 2007. He is currently with Cisco Systems Inc., San Jose, California.

Bhaskar Krishnamachari received the BE degree in electrical engineering from the Cooper Union, New York, in 1998 and the MS and PhD degrees from Cornell University in 1999 and 2002, respectively. He is currently the Philip and Cayley MacDonald Early Career Chair Assistant Professor in the Department of Electrical Engineering, University of Southern California. His primary research interest is the design and analysis of efficient mechanisms for operating wireless sensor networks.