Distributed Sensor Network Localization from Local Connectivity: Performance Analysis for the HOP-TERRAIN Algorithm

Amin Karbasi
LTHC & LCAV
Schol of Compter and Communication Sciences
EPFL, Switzereland
amin.karbasi@epfl.ch

Sewoong Oh
Electrical Engineering Department
Stanford Unviersity
Stanford, CA 94305, U.S.A
swoh@stanford.edu

ABSTRACT

This paper addresses the problem of determining the node locations in ad-hoc sensor networks when only connectivity information is available. In previous work, we showed that the localization algorithm MDS-MAP proposed by Y. Shang *et al.* is able to localize sensors up to a bounded error decreasing at a rate inversely proportional to the radio range r. The main limitation of MDS-MAP is the assumption that the available connectivity information is processed in a centralized way.

In this work we investigate a practically important question whether similar performance guarantees can be obtained in a distributed setting. In particular, we analyze the performance of the HOP-TERRAIN algorithm proposed by C. Savarese *et al.* This algorithm can be seen as a distributed version of the MDS-MAP algorithm. More precisely, assume that the radio range $r = o(1)$ and that the network consists of n sensors positioned randomly on a d-dimensional unit cube and $d+1$ anchors in general positions. We show that when only connectivity information is available, for every unknown node i, the Euclidean distance between the estimate $\hat{x}_i$ and the correct position x_i is bounded by

$$\|x_i - \hat{x}_i\| \leq \frac{r_0}{r} + o(1),$$

where $r_0 = C_d(\log n/n)^{\frac{1}{d}}$ for some constant C_d which only depends on d. Furthermore, we illustrate that a similar bound holds for the range-based model, when the approximate measurement for the distances is provided.

Categories and Subject Descriptors

C.2.1 [**Computer Systems Organization**]: Computer-Communication Networks Network Architecture and Design [Wireless Communication]

SIGMETRICS'10, June 14–18, 2010, New York, New York, USA.
Copyright 2010 ACM 978-1-4503-0038-4/10/06 ...$10.00.

General Terms

Theory, Performance, Algorithms

Keywords

distributed, localization, sensor network

1. INTRODUCTION

The problem we are interested in is determining the location of individual sensor nodes in a wireless ad-hoc sensor network. This problem of localization plays an important role in wireless sensor networks when the positions of the nodes are not provided in advance. One way to acquire the positions is by equipping all the sensors with a global positioning system (GPS). This not only adds considerable cost to the system, but more importantly, it does not work in indoor environments. As an alternative, we want an algorithm that can derive positions of sensors based on basic local information such as proximity (which nodes are within communication range of each others) or local distances (pairwise distances between neighboring sensors).

One consequence of the ad-hoc nature of the underlying networks is the lack of a central infrastructure. This fact renders the use of common centralized positioning algorithms [BY04, SRZF04]. We are interested in distributed algorithms which can be employed on large-scale ad-hoc sensor networks. Many distributed algorithms have been proposed where the success of them has mostly been measured experimentally [SLR02, NN03, SPS03]. To the best of our knowledge, the theoretical guarantees associated with the performance of these existing algorithms is rare with few exceptions such as [AEG$^+$06]. Such analytical bounds on the performance of distributed localization algorithms can provide answers to practical questions: for example, how large should the radio range be in order to get the error within a threshold? With this motivation in mind, our work takes a step in this direction. In particular, our analysis focuses on providing a bound on the performance of a popular robust positioning algorithm [SLR02] when applied to sensor localization from only connectivity information, which is a highly challenging problem. We prove that using HOP-TERRAIN [SLR02], we are able to localize sensors up to a bounded error in a connected network where most of the pairwise distances are unknown and only local connectivity information is given.

1.1 Related work

Recently, a number of localization algorithms have been proposed for sensor networks [SLR02, NN03, NSB03, RD07, SPS03, BY04, SRZF04, Sin08]. Based on the approach of processing the distance measurements, these algorithms can be classified into two categories: centralized algorithms and distributed algorithms. In centralized algorithms, all the distance measurements are sent to a single processor where the estimated positions are computed. Two well-known centralized localization algorithms are multidimensional scaling (MDS) based approaches [SRZF04] and semidefinite programming (SDP) based algorithms [BY04]. However, the centralized algorithms typically have low energy efficiency and low scalability due to dependency on a central processor and excessive communication overload.

Perhaps a more practical and interesting case is when there is no central infrastructure. [LR03] identifies a common three-phase structure of three popular distributed sensor localization algorithms, namely robust positioning [SLR02], ad-hoc positioning [NN03] and N-hop multilateration [SPS03] algorithm. Table 1 illustrates the structure of these algorithms. In the first phase nodes share information to collectively determine the distances from each of the nodes to a number of anchors. Anchors are special nodes with a priori knowledge of their own position in some global coordinate system. In the second phase nodes determine their position based on the estimated distances to the anchors provided by the first phase and the known positions of the anchors. In the last phase the initial estimated positions are iteratively refined. It is empirically demonstrated that these simple three-phase distributed sensor localization algorithms are robust and energy-efficient [LR03]. However, depending on which method is used in each phase there are different tradeoffs between localization accuracy, computation complexity and power requirements. In [NSB03], a distributed algorithm called the Gradient algorithm was proposed, which is similar to ad-hoc positioning [NN03] but uses a different method for estimating the average distance per hop.

Another distributed approach introduced in [IFMW04] is to pose the localization problem as an inference problem on a graphical model and solve it using Nonparametric Belief Propagation (NBP). It is naturally a distributed procedure and produces both an estimate of sensor locations and a representation of the location uncertainties. The estimated uncertainty may subsequently be used to determine the reliability of each sensor's location estimate. Despite recent developments in the algorithms for sensor localization little is known about the theoretical analysis supporting their empirical simulation results.

In an alternative line of work, the authors of [OKM10] provided a bound on the performance of a centralized localization algorithm known as MDS-MAP. When n sensors are randomly distributed in a unit d-dimensional hypercube, MDS-MAP is able to localize sensors up to a bounded error with only the pair-wise connectivity information, namely whether two nodes are within a given radio range r or not. This result crucially relies on the fact that there is a central processor with access to the inter-sensor distance measurements. However, centralized algorithms suffer from the scalability problem and require higher computational complexity. Hence, a distributed algorithm with similar performance bound is desirable. In this paper, we adapt the approach of [OKM10] to analyze the performance of a truly distributed sensor localization algorithm. We show that Hop-TERRAIN introduced in [SLR02] achieves a bounded error when only local connectivity information is given.

The organization of this paper is as follows. Section 2 describes a distributed sensor localization algorithm known as Hop-TERRAIN [SLR02] and states the main results on the performance. In Section 3, we provide detailed proof of the main theorems.

2. DISTRIBUTED LOCALIZATION ALGORITHM

In this section, we first define the basic mathematical model for the sensors and the measurements. We then describe the distributed localization algorithm and give a bound on the error between the correct position of the sensors and the estimated positions returned by the algorithm.

2.1 Model definition

Before discussing the distributed localization algorithm in detail, we first define the mathematical model. First, we assume that we have no fine control over the placement of the sensors which we call the *unknown nodes* (e.g., the nodes are dropped from an airplane). Formally, n sensors, or unknown nodes, are distributed uniformly at random within the d-dimensional hypercube $[0,1]^d$. Additionally, we assume that there are m special sensors, which we call *anchors*, with a priori knowledge of their own positions in some global coordinate. The anchors are assumed to be distributed uniformly at random in $[0,1]^d$ as well. However, it is reasonable to assume that we have some control over the positions of these special sensors, or the anchors, and in the following we show that we get similar performance bound with the minimum number of anchors ($m = d+1$) if the positions of the anchors are chosen properly.

Let $V_a = \{1,\dots,m\}$ denote the set of m vertices corresponding to the anchors and $V_u = \{m+1,\dots,m+n\}$ the set of n vertices corresponding to the unknown nodes. We consider the random geometric graph model $G(n,r) = (V,E,P)$ where $V = V_u \cup V_a$, $E \subseteq V \times V$ is a set of undirected edges that connect pairs of sensors which are close to each other, and $P : E \to \mathbb{R}^+$ is a non-negative real-valued function. We consider the function P as a mapping from a pair of connected nodes (i,j) to the approximate measurement for the distance between i and j, which we call the proximity measurement. Let $||\cdot||$ be the Euclidean norm in $\mathbb{R}^d$. Define a set of random positions of $n+m$ sensors $\mathcal{X} = \{x_1,\dots,x_m,x_{m+1},\dots,x_{m+n}\}$, where $x_a \in \mathbb{R}^d$ for $a \in \{1,\dots,m\}$ is the position of anchor a and and $x_i \in \mathbb{R}^d$ for $i \in \{m+1,\dots,m+n\}$ is the position of unknown node i. We assume that only the anchors have a priori information about their own positions. Then, a common model for the random geometric graph is the disc model where node i and j are connected if the Euclidean distance $d_{i,j} = ||x_i - x_j||$ is less than or equal to a positive radio range r. More formally,

$$(i,j) \in E \Leftrightarrow d_{i,j} \leq r \ .$$

To each edge $(i,j) \in E$, we associate the proximity measurement $P_{i,j}$ between sensors i and j, which is a function of the distance $d_{i,j}$ and random noise. In an ideal case when our measurements are exact, we have $P_{i,j} = d_{i,j}$. On the other extreme, when we are given only network connectivity

Table 1: Distributed localization algorithm classification

Phase	Robust positioning	Ad-hoc positioning	N-hop multilateration
1. Distance	DV-HOP	Euclidean	Sum-dist
2. Position	Lateration	Lateration	Min-max
3. Refinement	Yes	No	Yes

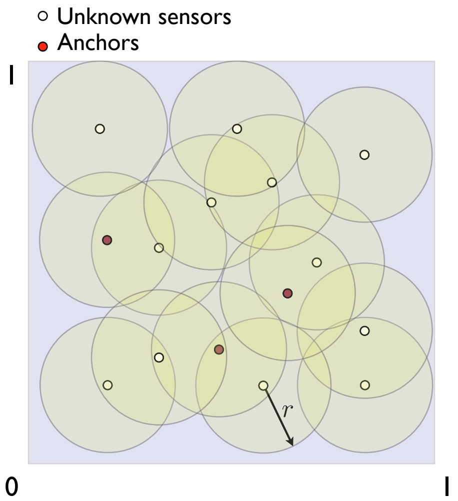

Figure 1: An example of a random geometric graph model in two dimensions where unknown sensors and anchors are distributed randomly. A node is connected to all other nodes that are within distance r of itself.

information and no distance information, we have constant $P_{i,j}$'s for all $(i,j) \in E$ (see Figure 1).

The sensor localization algorithms can be classified into two different categories. For the *connectivity-based model*, which is alternatively also known as the range-free model, only the connectivity information is available. Formally,

$$P_{i,j} = \begin{cases} r & \text{if } (i,j) \in E, \\ * & \text{otherwise,} \end{cases}$$

where a $*$ denotes that $d_{i,j} > r$.

For the *range-based model*, which is also known as the range-aware model, the distance measurement is available but may be corrupted by noise or have limited accuracy.

$$P_{i,j} = \begin{cases} [d_{i,j} + z_{i,j}]_+ & \text{if } (i,j) \in E, \\ * & \text{otherwise,} \end{cases}$$

where $z_{i,j}$ models the measurement noise (in the noiseless case $z_{i,j} = 0$), possibly a function of the distance $d_{i,j}$, and $[a]_+ \equiv \max\{0, a\}$. Common examples are the additive Gaussian noise model, where the $z_{i,j}$'s are i.i.d. Gaussian random variables, and the multiplicative noise model, where $P_{i,j} = [(1 + N_{i,j})d_{i,j}]_+$, for independent Gaussian random variables $N_{i,j}$'s.

In distributed sensor network localization not all the information is available at each node. Given the graph $G(n,r) = (V,E,P)$ with associated proximity measurements for each edges in E, we assume that each of the nodes is aware of the proximity measurements between itself and its adjacent neighbors and each of the anchors is also aware of its own position. Moreover, communication is only possible between adjacent neighboring nodes. The goal of distributed sensor network localization is for each node to find its estimated position that best fits the measured proximity with small number of communications.

The main contribution of this paper is that, to the best of our knowledge, we provide, for the first time, a performance bound for a distributed sensor localization algorithm when only the connectivity information is available. However, the algorithm can be readily applied to the *range-based model* without any modification. Further, given $G(V,E,P)$ from from the *range-based model* we can easily produce $G'(V,E,P')$ where $P'_{i,j} = r$ for all $(i,j) \in E$. This implies that the performance under the *range-based model* is also bounded by the main result in Theorems 2.1 and 2.2.

2.2 Algorithm

Based on the robust positioning algorithm introduced in [SLR02], the distributed sensor localization algorithm consists of two steps :

```
Algorithm : Hop-TERRAIN [SLR02]
1:  Each node $i$ computes the shortest paths
    $\{\hat{d}_{i,a} : a \in V_a\}$ between itself and the anchors;
2:  Each node $i$ derives an estimated position $\hat{x}_i$
    by triangulation with a least squares method.
```

According to the three phase classification presented in Table 1, this is closely related to the first two phases of the robust positioning algorithm. This algorithm uses a slightly different method for computing the shortest paths, which is compared in detail later in this section. Hence, through out this paper, we refer to this algorithm as HOP-TERRAIN, which denotes the first two steps of robust positioning algorithm in [SLR02].

Distributed shortest paths. The goal of the first step is for each of the unknown nodes to estimate the distances between itself and the anchors. This approximate distances will be used in the next triangulation step to derive an estimated position. The shortest path between an unknown node i and an anchor a in the graph G provides an estimate for the Euclidean distance $d_{i,a} = ||x_i - x_a||$, and for a carefully chosen radio range r this shortest path estimation is close to the actual distance as will be shown in Lemma 3.1.

Formally, the shortest path between an unknown node i and an anchor a in the graph $G = (V,E,P)$ is defined as a path between two nodes such that the sum of the proximity measures of its constituent edges is minimized. We denote by $\hat{d}_{i,a}$ the computed shortest path and this provides the initial estimate for the distance between the node i and the anchor a. When only the connectivity information is available and the corresponding graph $G = (V,E,P)$ is defined as in the *connectivity-based model*, the shortest path $\hat{d}_{i,a}$ is equivalent to the minimum number of hops between a node i and an anchor a multiplied by the radio range r.

In order to find the minimum number of hops from an unknown node $i \in V_u$ to an anchors $a \in V_a$ in a distributed way, we use a method similar to DV-HOP [NN03]. Each unknown node maintains a table $\{x_a, h_a\}$ which is initially empty, where $x_a \in \mathbb{R}^d$ refers to the position of the anchor a and h_a to the number of hops from the unknown node to the anchor a. First, each of the anchors initiate a broadcast containing its known location and a hop count of 1. All of the one-hop neighbors surrounding the anchor, on receiving this broadcast, record the anchor's position and a hop count of 1, and then broadcast the anchor's known position and a hop count of 2. From then on, whenever a node receives a broadcast, it does one of the two things. If the broadcast refers to an anchor that is already in the record and the hop count is larger than or equal to what is recorded, then the node does nothing. Otherwise, if the broadcast refers to an anchor that is new or has a hop count that is smaller, the node updates its table with this new information on its memory and broadcast the new information after incrementing the hop count by one. When every node has computed the hop count to all the anchors, the number of hops is multiplied by the radio range r to estimate the distances between the node and the anchors. Note that to start triangulation, not all the hop counts to all the anchors are necessary. A node can start triangulation as soon as it has estimated distances to $d+1$ anchors. There is an obvious trade-off between number of communications and performance.

The above step of computing the minimum number of hops is the same distributed algorithm as described in DV-HOP. However, the main difference is that instead of multiplying the number of hops by a fixed radio range r, in DV-HOP, the number of hops is multiplied by an average hop distance. The average hop distance is computed from the known pairwise distances between anchors and the number of hops between the anchors. While numerical simulations show that the average hop distance provides a better estimate, the difference between the computed average hop distance and the radio range r becomes negligible as n grows large.

We are interested in a scalable system of n unknown nodes for large value of n. As n grows large, it is reasonable to assume that the average number of connected neighbors for each node should stay constant. This happens, in our model, if we chose the radio range $r = C/n^{1/d}$. However, the number of hops is well defined only if the graph G is connected. If G is not connected there might be a set of unknown nodes that are connected to too few anchors, resulting in underdetermined series of equations in the triangulation step. In the unit square, assuming sensor positions are drawn uniformly at random as define in the previous section, the graph is connected, with high probability, if $\pi r^2 > (\log n + c_n)/n$ for $c_n \to \infty$ [GK98]. A similar condition can be derived for generic d-dimensions as $C_d r^d > (\log n + c_n)/n$, where $C_d \leq \pi$ is a constant that depends on d. Hence, we focus in the regime where the average number of connected neighbors is slowly increasing with n, namely, $r = \alpha(\log n/n)^{1/d}$ for some positive constant α such that the graph is connected with high probability.

As will be shown in Lemma 3.1, the key observation of the shortest paths step is that the estimation is guaranteed to be arbitrarily close to the correct distance for properly chosen radio range $r = \alpha(\log n/n)^{1/d}$ and large enough n. Moreover, this distributed shortest paths algorithm can be done efficiently with total complexity of $O(n\,m)$.

Triangulation using least squares. In the second step, each unknown node $i \in V_u$ uses a set of estimated distances $\{\hat{d}_{i,a} : a \in V_a\}$ together with the known positions of the anchors to perform a triangulation. the resulting estimated position is denoted by $\hat{x}_i$. For each node, the triangulation consists of solving a single instance of a least squares problem ($Ax = b$) and this process is known as Lateration [SRB01, LR03].

For an unknown node i, the position vector x_i and the anchor positions $\{x_a : a \in \{1, \dots, m\}\}$ satisfy the following series of equations:

$$\begin{aligned} ||x_1 - x_i||^2 &= d_{i,1}^2 \ , \\ &\vdots \\ ||x_m - x_i||^2 &= d_{i,m}^2 \ . \end{aligned}$$

This set of equations can be linearized by subtracting each line from the next line.

$$\begin{aligned} &||x_2||^2 - ||x_1||^2 + 2(x_1 - x_2)^T x_i \\ &\quad = d_{i,2}^2 - d_{i,1}^2 \ , \\ &\vdots \\ &||x_m||^2 - ||x_{m-1}||^2 + 2(x_{m-1} - x_m)^T x_i \\ &\quad = d_{i,m}^2 - d_{i,m-1}^2 \ . \end{aligned}$$

By reordering the terms, we get a series of linear equations for node i in the form $A\,x_i = b_0^{(i)}$, for $A \in \mathbb{R}^{(m-1)\times d}$ and $b \in \mathbb{R}^{m-1}$ defined as

$$\begin{aligned} A &\equiv \begin{bmatrix} 2(x_1 - x_2)^T \\ \vdots \\ 2(x_{m-1} - x_m)^T \end{bmatrix} , \\ b_0^{(i)} &\equiv \begin{bmatrix} ||x_1||^2 - ||x_2||^2 + d_{i,2}^2 - d_{i,1}^2 \\ \vdots \\ ||x_{m-1}||^2 - ||x_m||^2 + d_{i,m}^2 - d_{i,m-1}^2 \end{bmatrix} . \end{aligned}$$

Note that the matrix A does not depend on the particular unknown node i and all the entries are known exactly to all the nodes after the distributed shortest paths step. However, the vector $b_0^{(i)}$ is not available at node i, since $d_{i,a}$'s are not known. Hence we use an estimation $b^{(i)}$, which is defined from $b_0^{(i)}$ by replacing $d_{i,a}$ by $\hat{d}_{i,a}$ everywhere. Then, finding the optimal estimation $\hat{x}_i$ of x_i that minimizes the mean squared error is solved in a closed form using a standard least squares approach:

$$\hat{x}_i = (A^T A)^{-1} A^T b^{(i)} \ . \tag{1}$$

For bounded $d = o(1)$, a single least squares has complexity $O(m)$, and applying it n times results in the overall complexity of $O(n\,m)$. No communication between the nodes is necessary for this step.

2.3 Main results

Our main result establishes that HOP-TERRAIN [SLR02] achieves an arbitrarily small error for a radio range $r = \alpha(\log n/n)^{1/d}$ with a large enough constant α, when we have only the connectivity information as in the case of the *connectivity-based model*. The same bound holds immediately for the *range-based model*, when we have an approximate measurements for the distances, and the same algorithm can be applied without any modification. The extra information can be readily incorporated into the algorithm to compute better estimates for the actual distances between the unknown nodes and the anchors.

Define

$$r_0 \equiv 8\sqrt{3}\, d^{3/2} \left(\frac{\log n}{n}\right)^{\frac{1}{d}} \ , \tag{2}$$

THEOREM 2.1. *Assume n sensors and m anchors are distributed uniformly at random in the $[0,1]^d$ hypercube for a bounded dimension $d = O(1)$. For a given radio range $r > r_0$ and the number of anchors $m = \Omega(\log n)$, the following is true with high probability. For all unknown nodes $i \in V_u$, the Euclidean distance between the estimate $\hat{x}_i$ given by* HOP-TERRAIN *and the correct position x_i is bounded by*

$$||x_i - \hat{x}_i|| \leq \frac{r_0}{r} + O(r) \ , \tag{3}$$

where r_0 is defined in Eq. (2).

The proof is provided in Section 3. In the regime where $r = o(1)$, the above theorem implies that the error is inversely proportional to the radio range r. As described in the previous section, we are interested in the regime where $r = \alpha(\log n/n)^{1/d}$ for some constant α. Given a small positive constant δ, this implies that HOP-TERRAIN is guaranteed to produce estimated positions that satisfy $||x_i - \hat{x}_i|| \leq \delta$ for all $i \in V_u$ with a large enough constant α and large enough n.

When the number of anchors is bounded and the positions of the anchors are chosen randomly, it is possible that, in the triangulation step, we get an ill-conditioned matrix $A^T A$, resulting in an large estimation error. This happens, for instance, if three anchors fall close to a line. However, as mentioned in the previous section, it is reasonable to assume that, for the anchors, the system designer has some control over where they are placed. In that case, the next remark shows that when the positions of anchors are properly chosen, only $d+1$ anchors suffice to get a similar bound on the performance. Note that this is the minimum number of anchors necessary for triangulation. For simplicity we assume that one anchor is placed at the origin and d anchors are placed at positions corresponding to d-dimensional unit vectors. Namely, the position of the $d+1$ anchors are $\{[0,\ldots,0], [1,0,\ldots,0], [0,1,0,\ldots,0], [0,\ldots,0,1]\ \}$. (see figure 2)

THEOREM 2.2. *Assume that n sensors are distributed uniformly at random in the $[0,1]^d$ hypercube for a bounded dimension $d = O(1)$. Also, assume that there are $d+1$ anchors, one of which is placed at the origin, and the position vectors of the d remaining anchors are the d-dimensional unit vectors. For a given radio range $r > r_0$, the following is true with high probability. For all unknown nodes $i \in V_u$, the Euclidean distance between the estimate $\hat{x}_i$ given by* HOP-TERRAIN *and the correct position x_i is bounded by*

$$||x_i - \hat{x}_i|| \leq \frac{r_1}{r} + O(r) \ , \tag{4}$$

where r_0 is defined in Eq. (2) and $r_1 \equiv d\, r_0/2\sqrt{3}$.

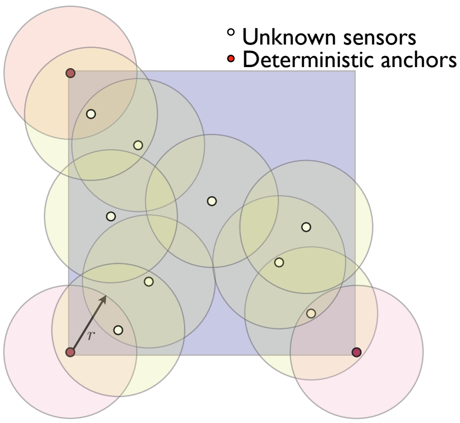

Figure 2: Three anchors in fixed positions for a two-dimensional sensor localization.

The proof of this remark closely follows that of Theorem 2.1, and is provided in Section 3. However, there is nothing particular about the position of the anchors in unit vectors. Any $d+1$ anchors in general position will give similar bound. Only difference is that the constant term in the definition of r_1 changes with the anchor positions.

When $r = \alpha(\log n/n)^{1/d}$ for some positive parameter α, the error bound in (4) is $||x_i - \hat{x}_i|| \leq C_1/\alpha + C_2\alpha\sqrt{\log n/n}$ for some numerical constants C_1 and C_2. The first term is inversely proportional to α and is independent of n, where as the second term is linearly dependent in α and vanishes as n grows large. This is illustrated in Figure 3, which shows numerical simulations with n sensors randomly distributed in the 2-dimensional unit square. We used 100 random samples for each point in the figure to compute the root mean squared error, $\{(1/n)\sum_{i=1}^n ||x_i - \hat{x}_i||^2\}^{1/2}$, averaged over the 100 samples. In the first figure, where we have only the connectivity information, we can see both contributions: the linear term which depends on n and the $1/\alpha$ term which is less sensitive to n. In the second figure, where we have the exact distance measurements without noise, we can see that the linear term almost vanishes even for $n = 5000$, and overall error is much smaller. A network of $n = 200$ nodes randomly placed in the unit square is shown in Figure 4. Three anchors in fixed positions are displayed by solid blue circles. In this experiment the distance measurements are from the range based model with multiplicative noise, where $P_{i,j} = [(1 + N_{i,j})d_{i,j}]_+$ for i.i.d. Gaussian $N_{i,j}$ with zero mean and variance 0.1. The noisy distance measurement is revealed only between the nodes within radio range $r = \sqrt{0.8 \log n/n}$. Figure 5 shows the final solution of HOP-TERRAIN. The circles represent the correct positions and the solid line represents the errors of the estimation from the correct position. The average error in this example is about 0.075.

2.4 Some technical remark and notations

In the following, whenever we write that a property $\mathcal{A}$ holds with high probability (w.h.p.), we mean that $\mathbb{P}(\mathcal{A})$

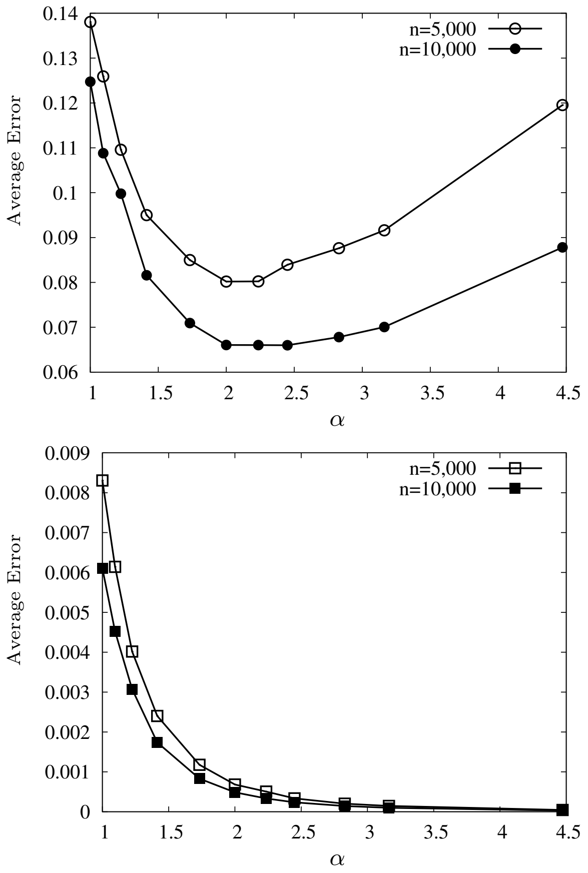

Figure 3: Average distance between the correct position $\{x_i\}$ and estimation $\{\hat{x}_i\}$ using Hop-TERRAIN as a function of α, for $r = \alpha\sqrt{\log n/n}$ with n sensors in the unit square under *connectivity based model* (above) and *range based model* (below).

approaches 1 as the number of sensors n goes to infinity. Given a matrix $A \in \mathbb{R}^{m\times n}$, the spectral norm of A is denoted by $||A||_2$, and the Frobenius norm is denoted by $||A||_F$. For a vector $a \in \mathbb{R}^n$, $||a||$ denotes the Euclidean norm. Finally, we use C to denote generic constants that do not depend on n.

3. PROOF OF THE MAIN THEOREMS

In this section we provide the proofs of the theorems 2.1 and 2.2. Detailed proofs of the technical lemmas are provided in the following sections.

For an unknown node i, the estimation $\hat{x}_i$ is given in Eq. (1).

$$\begin{aligned} ||x_i - \hat{x}_i|| &= ||(A^TA)^{-1}A^Tb_0^{(i)} - (A^TA)^{-1}A^Tb^{(i)}|| \\ &\le ||(A^TA)^{-1}A^T||_2||b_0^{(i)} - b^{(i)}|| \ , \end{aligned}$$

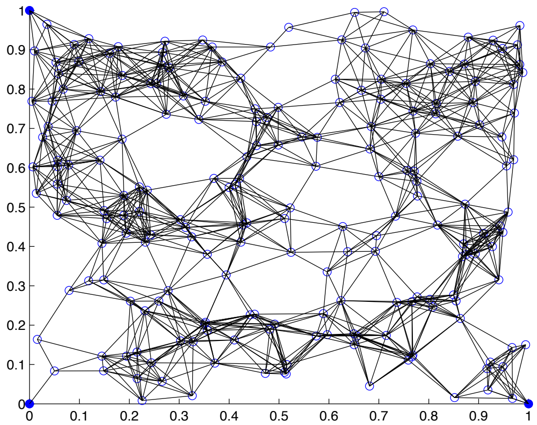

Figure 4: 200 nodes randomly placed in the unit square and 3 anchors in fixed positions. The radio range is $r = \sqrt{0.8 * \log n/n}$.

First, to bound $||b_0^{(i)} - b^{(i)}||$, we use the following key result on the shortest paths. The main idea is that, for sensors with uniformly random positions, shortest paths provide arbitrarily close estimation to the correct distances for large enough radio range r. Define

$$\tilde{r}_0(\beta) \equiv 8\sqrt{d}\left(\frac{(1+\beta)\log n}{n}\right)^{\frac{1}{d}} \ . \qquad (5)$$

For simplicity we denote $\tilde{r}_0(0)$ by $\tilde{r}_0$.

Lemma 3.1. (Bound on the shortest paths) *Under the hypothesis of Theorem 2.1, w.h.p., the shortest paths between all pairs of nodes in the graph $G(V,E,P)$ are uniformly bounded by*

$$\hat{d}^2_{i,j} \le (1+\frac{\tilde{r}_0}{r})d^2_{i,j} + O(r) \ ,$$

for $r > r_0$, where $\tilde{r}_0$ is defined as in Eq. (5) and r_0 as in Eq. (2).

The proof of this lemma is given in Section 3.1. Since $d^2_{i,j} \le d$ for all i and j, we have

$$\begin{aligned} ||b_0^{(i)} - b^{(i)}|| &= \Big(\sum_{k=1}^{m-1}\big(d^2_{k,2} - d^2_{k,1} - \hat{d}^2_{k,2} + \hat{d}^2_{k,1}\big)^2\Big)^{1/2} \\ &\le \sqrt{m-1}\frac{\tilde{r}_0}{r}\,d + O(\sqrt{m}r) \ , \qquad (6) \end{aligned}$$

Next, to bound $||(A^TA)^{-1}A^T||_2$, we use the following lemma.

Lemma 3.2. *Under the hypothesis of Theorem 2.1, the following are true. Assuming deterministic anchor model, where $m = d+1$ anchors are placed on the positions $x_1 = [1,0,\dots,0]$, $x_2 = [0,1,0,\dots,0]$, $x_3 = [0,\dots,0,1]$ and $x_m = [0,0,\dots,0]$.*

$$||(A^TA)^{-1}A^T||_2 \le \frac{d}{2} \ .$$

Assuming random anchor model, for $m = \Omega(\log n)$ anchors chosen uniformly at random among n sensors,

$$||(A^TA)^{-1}A^T||_2 \le \sqrt{\frac{3}{m-1}} \ ,$$

with high probability.

and this finishes the proof of Theorems 2.1 and 2.2.

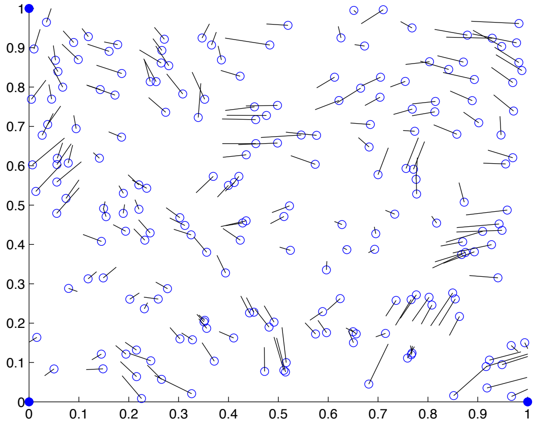

Figure 5: Location estimation using Hop-TERRAIN**.**

3.1 Proof of the bound on the shortest paths

In this section, we prove a slightly stronger version of Lemma 3.1. We will show that under the hypothesis of Theorem 2.1, for any $\beta \geq 0$ and all sensors $i \neq j$, there exists a constant C such that, with probability larger than $1 - \frac{Cn^{-\beta}}{(1+\beta)\log n}$, the shortest paths between all the pairs of nodes are uniformly bounded by

$$\hat{d}_{i,j}^2 \leq (1 + \frac{\tilde{r}_0(\beta)}{r})d_{i,j}^2 + O(r) \ , \qquad (7)$$

for $r > r_0$, where $\tilde{r}_0(\beta)$ is defined as in Eq. (5) and r_0 as in Eq. (2). Especially, Lemma 3.1 follows if we set $\beta = 0$.

We start by applying bin-covering technique to the random geometric points in $[0,1]^d$ in a similar way as in [MP05]. We cover the space $[0,1]^d$ with a set of non-overlapping hypercubes whose edge lengths are δ. Thus there are total $\lceil 1/\delta \rceil^d$ bins, each of volume δ^d. In formula, bin $(i_1, \dots, i_d)$ is the hypercube $[(i_1 - 1)\delta, i_1\delta) \times \cdots \times [(i_d - 1)\delta, i_d\delta)$, for $i_k \in \{1, \dots, \lceil 1/\delta \rceil\}$ and $k \in \{1, \dots, d\}$. When n nodes are deployed in $[0,1]^d$ uniformly at random, we say a bin is empty if there is no node inside the bin. We want to ensure that, with high probability, there are no empty bins.

For a given δ, define a parameter $\beta \equiv (\delta^d \, n/\log n) - 1$. Since a bin is empty with probability $(1 - \delta^d)^n$, we apply union bound over all the $\lceil 1/\delta \rceil^d$ bins to get,

$$\mathbb{P}(\text{no bins are empty}) \geq 1 - \left\lceil \frac{1}{\delta} \right\rceil^d (1 - \delta^d)^n \qquad (8)$$

$$\geq 1 - \frac{C\,n}{(1+\beta)\log n}\Big(1 - \frac{(1+\beta)\log n}{n}\Big)^n \qquad (9)$$

$$\geq 1 - \frac{C\,n^{-\beta}}{(1+\beta)\log n} \ , \qquad (10)$$

where in (9) we used the fact that there exists a constant C such that $\lceil 1/\delta \rceil^d \leq C/\delta^d$, and in (10) we used $(1 - 1/n)^n \leq e^{-1}$, which is true for any positive n.

Assuming that, with high probability, no bins are empty, we first show that the shortest paths is bounded by a function $F(d_{i,j})$ that only depends on the distance between the two nodes. Let $\hat{d}_{i,j}$ denote the shortest path between nodes i and j and $d_{i,j}$ denote the Euclidean distance. Define a function $F : \mathbb{R}^+ \to \mathbb{R}^+$ as

$$F(y) = \begin{cases} r & \text{if } y \leq r \ , \\ (k+1)r & \text{if } y \in \mathcal{L}_k \quad \text{for } k \in \{1, 2, \dots\} \ , \end{cases}$$

where $\mathcal{L}_k$ denotes the interval $(r + (k-1)(r - 2\sqrt{d})\delta, r + k(r - 2\sqrt{d})\delta]$.

We will show, by induction, that for all pairs (i,j),

$$\hat{d}_{i,j} \leq F(d_{i,j}) \ . \qquad (11)$$

First, in the case when $d_{i,j} \leq r$, by definition of *connectivity based model*, nodes i and j are connected by an edge in the graph G, whence $\hat{d}_{i,j} = r$.

Next, assume that the bound in Eq. (11) is true for all pairs (l,m) with $d_{l,m} \leq r + (k-1)(r - 2\sqrt{d})\delta$. We consider two nodes i and j at distance $d_{i,j} \in \mathcal{L}_k$. Consider a line segment $l_{i,j}$ in a d-dimensional space with one end at x_i and the other at x_j, where x_i and x_j denote the positions of nodes i and j, respectively. Let p be the point in the line $l_{i,j}$ which is at distance $r - \sqrt{d}\delta$ from x_i. Then, we focus on the bin that contains p. By assumption that no bins are empty, we know that we can always find at least one node in the bin. Let k denote any one of those nodes in the bin. Then we use following inequality which is true for all nodes (i,k,j).

$$\hat{d}_{i,j} \leq \hat{d}_{i,k} + \hat{d}_{k,j} \ .$$

Each of these two terms can be bounded using triangular inequality. To bound the first term, note that two nodes in the same bin are at most distance $\sqrt{d}\delta$ apart. Since p and x_k are in the same bin and p is at distance $r - \sqrt{d}\delta$ from node x_i by construction, we know that $d_{i,k} \leq ||x_i - p|| + ||p - x_k|| \leq r$, whence $\hat{d}_{i,k} = r$. Analogously for the second term, $d_{k,j} \leq ||x_j - p|| + ||p - x_k|| \leq r + (k-1)(r - 2\sqrt{d})\delta$, which implies that $\hat{d}_{k,j} \leq F(d_{k,j}) = kr$. Hence, we proved that if Eq. (11) is true for pairs (i,j) such that $d_{i,j} \leq r + (k-1)(r - 2\sqrt{d})\delta$, then $\hat{d}_{i,j} \leq (k+1)r$ for pairs (i,j) such that $d_{i,j} \in \mathcal{L}_k$. By induction, this proves that the bound in Eq. (11) is true for all pairs (i,j).

Let $\mu = (r/2\sqrt{d})\Big(n/((1+\beta)\log n)\Big)^{1/d}$. Then, the function $F(y)$ can be easily upper bounded by an affine function $F_a(y) = (1 + 1/(\mu - 1))y + r$. Hence we have following bound on the shortest paths between any two nodes i and j.

$$\hat{d}_{i,j} \leq \left(1 + \frac{1}{\frac{r}{2\sqrt{d}}\left(\frac{n}{(1+\beta)\log n}\right)^{1/d} - 1} \right) d_{i,j} + r \ . \qquad (12)$$

Figure 6 illustrates the comparison of the upper bounds $F(d_{i,j})$ and $F_a(d_{i,j})$, and the trivial lower bound $\hat{d}_{i,j} \geq d_{i,j}$ in a simulation with parameters $d = 2$, $n = 6000$ and $r = \sqrt{64 \log n/n}$. The simulation data shows the distribution of shortest paths between all pairs of nodes with respect to the actual pairwise distances, which confirms that shortest paths lie between the analytical upper and lower bounds. While the gap between the upper and lower bound is seemingly large, in the regime where $r = \alpha\sqrt{\log n/n}$ with a constant α, the vertical gap r vanishes as n goes to infinity and the slope of the affine upper bound can be made arbitrarily small by increasing the radio range r or equivalently taking large enough α. The bound on the squared shortest paths $\hat{d}_{i,j}^2$ can be derived from the bound on the shortest

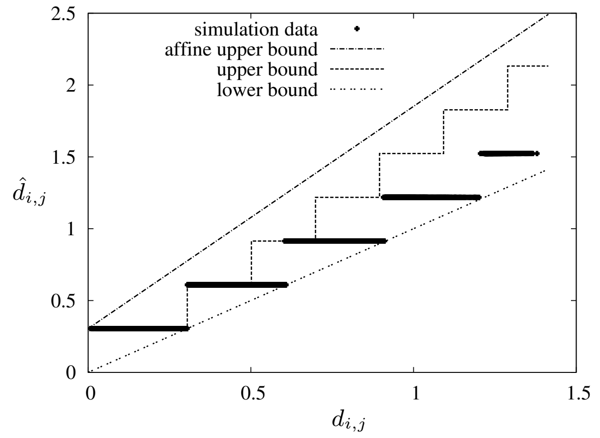

Figure 6: comparison of upper and lower bound of shortest paths $\{\hat{d}_{i,j}\}$ with respect to the correct distance $\{d_{i,j}\}$ computed for $n = 6000$ sensors in 2-dimensional square $[0, 1]^2$ under *connectivity based model.*

paths in Eq. (12) after some calculus.

$$\begin{aligned} \hat{d}^2_{i,j} &\le \left\{\frac{\mu}{\mu-1} d_{i,j} + r\right\}^2 & (13)\\ &= \frac{\mu^2}{(\mu-1)^2} d^2_{i,j} + r^2 + 2\frac{\mu}{\mu-1} d_{i,j}\, r & (14)\\ &= \left(1 + \frac{2\mu-1}{(\mu-1)^2}\right) d^2_{i,j} + O(r) & (15)\\ &\le \left(1 + \frac{4}{\mu}\right) d^2_{i,j} + O(r)\ . & (16) \end{aligned}$$

where in (15) we used the fact that $(\mu/(\mu-1))d_{i,j} = O(1)$, which follows from the assumptions $(r/4\sqrt{d})^d > (1+\beta)\log n/n$ and $d = O(1)$. In (16), we used the inequality $(2\mu-1)/(\mu-1)^2 \le 4/\mu$, which is true for $\mu \ge 2+\sqrt{3}$. Substituting μ in the formula, this finishes the proof of the desired bound in Eq. (7).

Note that although for the sake of simplicity, we focus on $[0, 1]^d$ hypercube, our analysis easily generalizes to any bounded convex set and homogeneous Poisson process model with density $\rho = n$. The homogeneous Poisson process model is characterized by the probability that there are exactly k nodes appearing in any region with volume A : $\mathbb{P}(k_A = k) = \frac{(\rho A)^k}{k!} e^{-\rho A}$. Here, k_A is a random variable defined as the number of nodes in a region of volume A.

3.2 Proof of Lemma 3.2

By using the singular value decomposition of a tall $m-1 \times d$ matrix A, we know that it can be written as $A = U\Sigma V^T$ where U is an orthogonal matrix, V is a unitary matrix and Σ is a diagonal matrix. Then,

$$(A^T A)^{-1} A = U\Sigma^{-1}V^T.$$

Hence,

$$||(A^T A)^{-1} A||_2 = \frac{1}{\sigma_{min}(A)}, \tag{17}$$

where $\sigma_{min}(A)$ is the smallest singular value of A. This means that in order to upper bound $||(A^T A)^{-1}A||_2$ we need to lower bound the smallest singular value of A.

3.2.1 Deterministic Model

By putting the sensors in the mentioned positions the $d \times d$ matrix A will be Toeplitz and have the following form.

$$A = 2\begin{bmatrix} 1 & -1 & 0 & \cdots & 0 \\ 0 & 1 & -1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 & -1 \\ 0 & \cdots & 0 & 0 & 1 \end{bmatrix}.$$

We can easily find the inverse of matrix A.

$$A^{-1} = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 0 & 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 & 1 \\ 0 & \cdots & 0 & 0 & 1 \end{bmatrix}.$$

Note that the maximum singular value of A^{-1} and the minimum singular value of A are related as follows.

$$\sigma_{min}(A) = \frac{1}{\sigma_{max}(A^{-1})}. \tag{18}$$

To find the maximum singular value of A^{-1} need to calculate the maximum eigenvalue of $A^{-1}\left(A^{-1}\right)^T$ which has the form

$$A^{-1}\left(A^{-1}\right)^T = \frac{1}{4}\begin{bmatrix} d & d-1 & d-2 & \cdots & 1 \\ d-1 & d-1 & d-2 & \cdots & 1 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 2 & \cdots & 2 & 2 & 1 \\ 1 & \cdots & 1 & 1 & 1 \end{bmatrix}.$$

Using the Gershgorin circle theorem (see appendix B) we can find an upper bound on the maximum eigenvalue of $A^{-1}\left(A^{-1}\right)^T$.

$$\lambda_{\max}\left(A^{-1}\left(A^{-1}\right)^T\right) \le \frac{d^2}{4}, \tag{19}$$

Hence, by combining (17) and (19) we get

$$||(A^T A)^{-1} A||_2 \le \frac{d}{2}. \tag{20}$$

3.2.2 Random Model

Let the symmetric matrix B be defined as $A^T A$. The diagonal entries of B can be written as

$$b_{i,i} = 4\sum_{k=1}^{m-1}(x_{k,i} - x_{k+1,i})^2, \tag{21}$$

for $1 \le i \le d$ and the off-diagonal entries as

$$b_{i,j} = 4\sum_{k=1}^{m-1}(x_{k,i} - x_{k+1,i})(x_{k,j} - x_{k+1,j}), \tag{22}$$

for $1 \le i \ne j \le d$ where $x_{k,i}$ is the i-th element of vector x_k. In the following lemmas, we show that with high probability as m increases the diagonal entries of B will all be of the order of m, i.e., $b_{i,i} = \Theta(m)$, and the off-diagonal entries will be bounded from above by $m^{\frac{1}{2}+\epsilon}$, i.e., $b_{i,j} = o(m)$.

LEMMA 3.3. *For any $\epsilon > 0$ the diagonal entries of B are bounded as follows.*

$$\mathbb{P}\left(|b_{i,i} - 2(m-1)/3| > 4m^{\frac{1}{2}+\epsilon}\right) \le 4e^{-m^{2\epsilon}}.$$

The idea is to use Hoeffding's Inequality (see appendix A) for the sum of independent and bounded random variables. To this end, we need to divide the sum in (21) into sums of even and odd terms as follows:

$$b_{i,i} = b_e^i + b_o^i,$$

where

$$\begin{aligned} b_e^i &= 4 \sum_{k \in \text{even}} (x_{k,i} - x_{k+1,i})^2, && (23) \\ b_o^i &= 4 \sum_{k \in \text{odd}} (x_{k,i} - x_{k+1,i})^2. && (24) \end{aligned}$$

This separation ensures that the random variables in summations (23) and (24) are independent. Let the random variable z_k^i denote the term $4(x_{k,i} - x_{k+1,i})^2$ in (23). Since $z_k^i \in [0,4]$ and all the terms in b_e^i are independent of each other, we can use Hoeffding's Inequality to upper bound the probability of the deviation of b_e^i from its expected value:

$$\mathbb{P}\left(|b_e^i - (m-1)/3| > 2m^{\frac{1}{2}+\epsilon}\right) \le 2e^{-m^{2\epsilon}}, \qquad (25)$$

for any fixed $\epsilon > 0$. The same bound holds for b_o. Namely,

$$\mathbb{P}\left(|b_o^i - (m-1)/3| > 2m^{\frac{1}{2}+\epsilon}\right) \le 2e^{-m^{2\epsilon}}. \qquad (26)$$

Hence,

$$\begin{aligned} &\mathbb{P}\left(|b_{i,i} - 2(m-1)/3| > 4m^{\frac{1}{2}+\epsilon}\right) \\ &\overset{(a)}{\le} \mathbb{P}\left(|b_e - (m-1)/3| + |b_o - (m-1)/3| > 4m^{\frac{1}{2}+\epsilon}\right) \\ &\overset{(b)}{\le} 4e^{-m^{2\epsilon}}, \end{aligned}$$

where in (a) we used triangular inequality and in (b) we used the union bound.

Lemma 3.4. *For any $\epsilon > 0$ the off-diagonal entries of B are bounded as follows.*

$$\mathbb{P}\left(|b_{i,j}| > 16m^{\frac{1}{2}+\epsilon}\right) \le 4e^{-m^{2\epsilon}}.$$

The proof follows the same lines as the proof of Lemma 3.3.

Using the Gershgorin circle theorem (see appendix B) we can find a lower bound on the minimum eigenvalue of B.

$$\lambda_{\min}(B) \ge \min_i (b_{i,i} - R_i), \qquad (27)$$

where

$$R_i = \sum_{j \ne i} |b_{i,j}|.$$

Now, let $\mathbb{B}_{ii}$ denote the event that $\{b_{i,i} < 2(m-1)/3 - 4m^{\frac{1}{2}+\epsilon}\}$ and $\mathbb{B}_{ij}$ (for $i \ne j$) denote the event that $\{b_{i,j} > 16m^{\frac{1}{2}+\epsilon}\}$. Since the matrix B is symmetric, we only have $d(d+1)/2$ degrees of freedom. Lemma 3.3 and 3.4 provide us with a bound on the probability of each event. Therefore, by using the union bound we get

$$\begin{aligned} \mathbb{P}\left(\bigcup_{i \le j} \overline{\mathbb{B}_{ij}}\right) &\le 1 - \sum_{i \le j} \mathbb{P}(\mathbb{B}_{ij}) \\ &= 1 - 3d^2 e^{-m^{2\epsilon}}. \end{aligned}$$

Therefore with probability at least $1 - 3d^2 e^{-m^{2\epsilon}}$ we have

$$b_{i,i} - R_i \ge \frac{2(m-1)}{3} - 16d \cdot m^{\frac{1}{2}+\epsilon}, \qquad (28)$$

for all $1 \le i \le d$. As m grows, the RHS of (28) can be lower bounded by $(m-1)/3$. By combining (27) and (28) we can conclude that

$$\mathbb{P}\left(\lambda_{\min}(B) \ge \frac{(m-1)}{3}\right) \ge 1 - 3d^2 e^{-m^{2\epsilon}}. \qquad (29)$$

As a result, from (17) and (29) we have

$$\mathbb{P}\left(||(A^T A)^{-1} A||_2 \le \sqrt{\frac{3}{m-1}}\right) \ge 1 - 3d^2 e^{-m^{2\epsilon}}, \qquad (30)$$

which shows that as m grows with high probability we have $||(A^T A)^{-1} A||_2 \le \sqrt{\frac{3}{m-1}}$.

4. CONCLUSION

In many applications of wireless sensor networks, it is crucial to determine the location of nodes. Distributed localization of nodes is a key to enable most of these applications. For this matter, numerous algorithms have been recently proposed where the efficiency and success of them have been mostly demonstrated by simulations. In this paper, we investigated the distributed sensor localization problem from a theoretical point of view and provided analytical bounds on the performance of such an algorithm. More precisely, we analyzed the HOP-TERRAIN algorithm and showed that even when only the connectivity information was given, the Euclidean distance between the estimate and the correct position of every unknown node is bounded and decays at a rate inversely proportional to the radio range. In the case of noisy distance measurements, we observe the same behavior and a similar bound holds.

Acknowledgment

We thank Andrea Montanari and Rüdiger Urbanke for stimulating discussions on the subject of this paper.

5. REFERENCES

[AEG+06] J. Aspnes, T. Eren, D. K. Goldenberg, A. S. Morse, W. Whiteley, Y. R. Yang, B. D. O. Anderson, and P. N. Belhumeur. A theory of network localization. *IEEE Transactions on Mobile Computing*, 5(12):1663–1678, December 2006.

[BY04] P. Biswas and Y. Ye. Semidefinite programming for ad hoc wireless sensor network localization. In *IPSN '04: Proceedings of the 3rd international symposium on Information processing in sensor networks*, pages 46–54, New York, NY, USA, 2004. ACM.

[GK98] P. Gupta and P.R. Kumar. Critical power for asymptotic connectivity. In *Proceedings of the 37th IEEE Conference on Decision and Control*, volume 1, pages 1106–1110 vol.1, 1998.

[HJ85] R. Horn and C. Johnson. *Matrix analysis.* Cambridge University Press, 1985.

[Hoe63] W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58:13âĂŞ30, 1963.

[IFMW04] A. T. Ihler, J. W. Fisher, R. L. Moses, and A. S. Willsky. Nonparametric belief propagation for self-calibration in sensor networks. In *IPSN '04: Proceedings of the 3rd international symposium on Information processing in sensor networks*, pages 225–233, New York, NY, USA, 2004. ACM.

[LR03] K. Langendoen and N. Reijers. Distributed localization in wireless sensor networks: a quantitative comparison. *Comput. Netw.*, 43(4):499–518, 2003.

[MP05] S. Muthukrishnan and G. Pandurangan. The bin-covering technique for thresholding random geometric graph properties. In *SODA '05: Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 989–998, Philadelphia, PA, USA, 2005.

[NN03] D. Niculescu and B. Nath. DV based positioning in ad hoc networks. *Journal of Telecommunication Systems*, 22:267–280, 2003.

[NSB03] R. Nagpal, H. Shrobe, and J. Bachrach. Organizing a global coordinate system from local information on an ad hoc sensor network. In *IPSN '03: Proceedings of the 2nd international conference on Information processing in sensor networks*, pages 333–348, 2003.

[OKM10] S. Oh, A. Karbasi, and A. Montanari. Sensor network localization from local connectivity: Performance analysis for the mds-map algorithm. In *2010 IEEE Information Theory Workshop (ITW 2010)*, January 6-8 2010. `http://infoscience.epfl.ch/record/140635`.

[RD07] M. Rudafshani and S. Datta. Localization in wireless sensor networks. In *IPSN '07: Proceedings of the 6th international conference on Information processing in sensor networks*, pages 51–60, New York, NY, USA, 2007. ACM.

[Sin08] A. Singer. A remark on global positioning from local distances. *Proceedings of the National Academy of Sciences*, 105(28):9507–9511, 2008.

[SLR02] C. Savarese, K. Langendoen, and J. Rabaey. Robust positioning algorithms for distributed ad-hoc wireless sensor networks. In *USENIX Technical Annual Conference*, pages 317–328, Monterey, CA, June 2002.

[SPS03] A. Savvides, H. Park, and M. Srivastava. The n-hop multilateration primitive for node localization problems. *Mob. Netw. Appl.*, 8(4):443–451, 2003.

[SRB01] C. Savarese, J. Rabaey, and J. Beutel. Locationing in distributed ad-hoc wireless sensor networks. In *in ICASSP*, pages 2037–2040, 2001.

[SRZF04] Y. Shang, W. Ruml, Y. Zhang, and M. P. J. Fromherz. Localization from connectivity in sensor networks. *IEEE Trans. Parallel Distrib. Syst.*, 15(11):961–974, 2004.

APPENDIX

A. HOEFFDING'S INEQUALITY

Hoeffding's inequality [Hoe63] is a result in probability theory that gives an upper bound on the probability for the sum of random variables to deviate from its expected value. Let $z_1, z_2, \dots, z_n$ be independent and bounded random variables such that $z_k \in [a_k, b_k]$ with probability one. Let $s_n = \sum_{k=1}^n z_k$. Then for any $\delta > 0$, we have

$$\mathbb{P}\left(|s_n - \mathbb{E}[s_n]| \geq \delta\right) \leq 2e^{-\frac{2\delta^2}{\sum_{k=1}^n (b_k - a_k)^2}}.$$

B. GERSHGORIN CIRCLE THEOREM

The Gershgorin circle theorem [HJ85] identifies a region in the complex plane that contains all the eigenvalues of a complex square matrix. For an $n \times n$ matrix A, define

$$R_i = \sum_{j \neq i} |a_{i,j}|.$$

Then each eigenvalue of A is in at least one of the disks

$$\{z : |z - a_{i,i}| \leq R_i\}.$$