Distance Matrix Reconstruction from Incomplete Distance Information for Sensor Network Localization

P. Drineas*, A. Javed*, M. Magdon-Ismail*, G. Pandurangan†, R. Virrankoski‡ and A. Savvides‡
*CS Department
Rensselaer Polytechnic Institute, Troy, NY
Email: drinep, javeda, magdon@cs.rpi.edu
†CS Department
Purdue University, West Lafayette, IN
Email: gopal@cs.purdue.edu
‡EE Department
Yale University, New Haven, CT
Email: reino.virrankoski, andreas.savvides@yale.edu

***Abstract*— This paper focuses on the principled study of distance reconstruction for distance-based node localization. We address an important issue in node localization by showing that a highly incomplete set of inter-node distance measurements obtained in ad-hoc node deployments carries sufficient information for the accurate reconstruction of the missing distances, *even in the presence of noise and sensor node failures*. We provide an efficient and provably accurate algorithm for this reconstruction, and we show that the resulting error is bounded, decreasing at a rate that is inversely proportional to $\sqrt{n}$, the square root of the number of nodes in the region of deployment. Although this result is applicable to many localization schemes, in this paper we illustrate its use in conjunction with the popular MultiDimensional Scaling algorithm. Our analysis reveals valuable insights and key factors to consider during the sensor network setup phase, to improve the quality of the position estimates.**

I. Introduction

In the past few years the sensor network community has reached a consensus that knowledge of node locations is unquestionably one of the most desirable attributes of ad-hoc sensor networks. Knowledge of location can support many networking and maintenance services, and more importantly map the sensed data to physical space. Since the manual recording of node positions is a difficult task even for modest sized networks, the community has invested significant effort in creating algorithms that can derive locations based on inter-node measurements.

The simplest and most common embodiment of such algorithms considers the estimation of a coordinate system from a set of pairwise distance measurements among sensor nodes. However, it is well known, that in realistic deployments obstacles and large node separations render the collection of all n^2 distances infeasible. Many of the existing algorithms try to resolve this issue by providing heuristic approximations to the missing distances. The success of such techniques has invariably been measured experimentally. There is an alarming lack of simple algorithms with bounded running time complexity – either centralized or decentralized – that are able to *provably* localize the sensor nodes up to bounded error.

The work in this paper takes a forward step in this direction, by providing a simple and provable algorithm for the accurate reconstruction of the missing pairwise distance measurements. The main contribution of this paper is to show that highly incomplete distance matrices such as the ones obtained in ad-hoc deployments, contain sufficient information to allow the accurate reconstruction of the missing distances, even in the presence of noise. To this end, we describe a provable reconstruction algorithm with bounded error and illustrate its use in conjunction with the popular Multidimensional Scaling (MDS) algorithm [12], [13], [8]. However, we emphasize that this presentation focuses on matrix distance reconstruction. We acknowledge the fact that to obtain more accurate locations an additional iterative refinement phase similar to the ones described in [13] and [14] is necessary. This presentation does not delve into the

details of iterative refinement.

Section III gives an intuitive overview of the main results, followed by a detailed description in Sections IV and V and our evaluation results in Section VII.

II. Related Work

Node localization has been a subject of intense study in the recent literature. The various approaches may be classified based on whether they are assisted or ad-hoc, centralized or distributed, or based on the type of technologies they employ. Some approaches are based on radio received signal strength [15], [13], [10], others employ more accurate distance measurement technologies [11], and others assume a combination of angle and distance measurements [4], [10].

Our work is closely related to studies that use approximations to distance measurements. These include the MDS based approaches described in [12], [13], [8]. Novel distance reconstruction techniques via SemiDefinite Programming formulations (SDP) have been recently proposed in [3], [9], [14]. Our work addresses the same problem. However, to the best of our knowledge, no explicit connection between the accuracy of the reconstruction and the number of sensor nodes in the network has been provided in existing work.

There has been significant recent theoretical work in general *matrix reconstruction* problems, a special case of which is the Euclidean distance matrix reconstruction problem. In particular, Achlioptas and McSherry in [1], [2] proved that given randomly sampled elements of a matrix, it is possible to accurately approximate the spectral characteristics – singular values and singular vectors – of a matrix. Drineas *et. al.* in [5], [6] proved that it is also possible to approximate the spectral characteristics of a matrix by sampling a small constant number of rows and/or columns of a matrix. We refer the reader to the references for further details.

III. Distance Matrix Reconstruction

A. Problem Statement

In a sensor network localization problem, n sensor nodes are placed in the two (or three)[1] dimensional Euclidean space. Every sensor measures its distance (up to noise) to a subset of the other sensors. Given this (incomplete) distance information, the task is to recover the positions of the individual sensor nodes. More formally, let $\mathbf{x}_i \in \mathbb{R}^2$ denote the position of node i, $i \in 1 \dots n$. Let d_{ij} denote the Euclidean distance between nodes i and j for $i, j \in 1 \dots n$, i.e.,

$$d_{ij}^2 = \|\mathbf{x}_i - \mathbf{x}_j\|^2 = \mathbf{x}_i^T \mathbf{x}_i + \mathbf{x}_j^T \mathbf{x}_j - 2\mathbf{x}_i^T \mathbf{x}_j.$$

Let $\mathbf{X}$ denote the $n \times 2$ position matrix whose i^{th} row is $\mathbf{x}_i^T$, and let $\mathbf{D}$ denote the $n \times n$ distance matrix given by $D_{ij} = d_{ij}^2$. We emphasize that the entries of $\mathbf{D}$ are the *square* of the Euclidian distance d_{ij}. We assume that the sensors are distributed on a bounded domain, so $d_{ij} \in [0, d_{max}]$. Estimates $\tilde{d}_{ij}^2 = d_{ij}^2 + \epsilon_{ij}$ are measured for some pairs of nodes, where ϵ_{ij} models the measurement noise. We assume that the noise is zero mean and has bounded variance. However, we do not assume that it is Gaussian. The goal of localization is to recover estimates $\tilde{\mathbf{x}}_i \in \mathbb{R}^2$ that are "close", up to rotation/reflection and translation, to the $\mathbf{x}_i$ for all $i \in 1 \dots n$.

Existing algorithms for localization (e.g., the MDS-MAP algorithm of [12], [13]) start by using the incomplete and noisy distance information contained in the $\tilde{d}_{ij}$ to first reconstruct all the distances d_{ij}. The goal of this paper is to give provably accurate algorithms for *reconstructing* the entire distance matrix, given a small number of noisy pairwise distances $\tilde{d}_{ij}^2$. In particular, we obtain estimates $\bar{d}_{ij}^2$ for all $i, j \in 1 \dots n$ for which, modulo our assumptions,

$$\frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} (d_{ij}^2 - \bar{d}_{ij}^2)^2 = O\left(\frac{1}{\sqrt{n}}\right).$$

In words, the squared error per entry drops inversely proportional to the square root of the number of nodes in the sensor network. Thus, we lay a theoretical foundation upon which existing algorithms, such as MDS-MAP, may operate.

Notation. Let $\mathbf{1}_n$ be the n-dimensional vector of ones, and $\mathbf{I}_n$ the $n \times n$ identity matrix. For any matrix $\mathbf{A}$, $\|\mathbf{A}\|_F^2 = \sum_{i,j} A_{ij}^2$ and $\|\mathbf{A}\|_2 = \max_{\|\mathbf{y}\|=1} \|\mathbf{A}\mathbf{y}\|$.

B. MDSLocalize *Using Exact Distances*

To motivate the need for accurate reconstruction of the distance matrix, we can ask whether it is possible to recover the original positions $\mathbf{x}_i$ (up to rotation/reflection and translation), given *all* n^2 pairwise Euclidean distances, without any measurement noise. A SemiDefinite Programing approach used in [3] shows that the answer to this question is affirmative. It has been folklore knowledge that under the same assumptions, MultiDimensional Scaling (MDS) approaches do the same. We summarize the MDS algorithm below, and give a proof of Theorem 1 in the Appendix.

[1] In the interest of space, we only focus on the 2D case. The 3D case is a straight forward extension.

Algorithm MDSLOCALIZE

1) **Centering.** Compute $\boldsymbol{\tau}(\mathbf{D}) = -\frac{1}{2}\mathbf{LDL}$, where $\mathbf{L} = \mathbf{I}_n - (1/n)\mathbf{1}_n\mathbf{1}_n^T$.
2) **SVD.** Compute $\boldsymbol{\tau}_2(\mathbf{D})$, the best rank 2 approximation to $\boldsymbol{\tau}(\mathbf{D})$ using its Singular Value Decomposition, $\boldsymbol{\tau}_2(\mathbf{D}) = \mathbf{U}_2\boldsymbol{\Sigma}_2^2\mathbf{U}_2^T$.
3) **Return** $\tilde{\mathbf{X}} = \mathbf{U}_2\boldsymbol{\Sigma}_2$.

At the second step of MDSLOCALIZE, $\mathbf{U}_2$ is an $n \times 2$ matrix of the top two left singular vectors of $\boldsymbol{\tau}(\mathbf{D})$, and $\boldsymbol{\Sigma}_2$ is a 2×2 diagonal matrix. At the third step, the i^{th} row of $\tilde{\mathbf{X}}$ is the estimate $\tilde{\mathbf{x}}_i^T$.

Theorem 1: MDSLOCALIZE, when applied to the complete, exact distance matrix $\mathbf{D}$ returns estimates of the positions $\tilde{\mathbf{x}}_i$ that are equal (up to rotation/reflection and translation) to the true positions $\mathbf{x}_i$ for all i.

The above theorem immediately suggests an approach when some of the pairwise distances are missing: replace the missing entries by estimates and run MDS on this estimate of $\mathbf{D}$. Indeed, this approach has been suggested and experimentally evaluated in [12], where a missing distance between nodes i and j is *approximated* by its shortest path distance in the sensor network connectivity graph. The hope has always been that if the estimate of $\mathbf{D}$ is accurate enough, then the result of the MDSLOCALIZE procedure will mimic the statement of theorem 1. We will show here that the first step can be accomplished, namely that $\mathbf{D}$ can be reconstructed from partial information with provable accuracy. The analysis of running MDSLOCALIZE on this provably accurate reconstruction will be discussed in upcoming work.

C. Inferring Missing Distances

A crucial question naturally arises. Can one accurately approximate the missing distances, given a small subset of pairwise distances?

Lemma 1: The rank of $\mathbf{D}$ is at most 4.

Proof: Notice that

$$\mathbf{D} = \mathbf{1}_n\mathbf{z}^T + \mathbf{z}\mathbf{1}_n^T - 2\mathbf{X}\mathbf{X}^T, \tag{1}$$

where $\mathbf{z}$ is an $n \times 1$ vector whose i^{th} element is equal to $\|\mathbf{x}_i\|^2 = \mathbf{x}_i^T\mathbf{x}_i$. To conclude, observe that $\mathbf{D}$ is the sum of three matrices of ranks 1, 1, and at most 2. More generally, in d dimensions, the rank of the third matrix is at most d, giving $rank(\mathbf{D}) \leq d + 2$.

$\diamond$

This simple lemma lies at the *heart* of our work. The fact that $\mathbf{D}$ is of rank at most 4 explains, both rigorously and intuitively, the correctness of our algorithm and the quality of our bounds. Intuitively, it states that $\mathbf{D}$ has a lot of structure. Roughly speaking, even though $\mathbf{D}$ has n^2 entries, there exist only 4 linearly independent columns (or rows) in $\mathbf{D}$ or, equivalently, there exist only $8n$ degrees of freedom in $\mathbf{D}$. Thus, a carefully chosen $8n$ entries in $\mathbf{D}$ should suffice to reconstruct $\mathbf{D}$ exactly.

D. Sampling $\mathbf{D}$

As discussed, only $8n$ entries in $\mathbf{D}$ should suffice for reconstruction, and hence localization. As a motivating example, consider an idealized setting, in which we could choose which entries of $\mathbf{D}$ to measure. Suppose we pick 4 linearly independent rows of $\mathbf{D}$, say (without loss of generality) the first 4 rows. This amounts to the unrealistic assumption that we are given all distances from the first 4 sensor nodes to *all* other nodes. Assume also that we are given at least 4 entries from every other row of $\mathbf{D}$, i.e., every sensor is able to compute its distance to at least 4 other sensors (a realistic assumption). The 4 entries in row j ($j > 4$) may be used to determine the linear combination of the first 4 rows that would give the j^{th} row, and hence determine the entire j^{th} row. We know that this process is feasible, since $\mathbf{D}$ has rank at most 4. Thus, the 4 given entries in each row suffice to reconstruct the entire row. Assuming that the measurements are noiseless, the reconstruction of $\mathbf{D}$ is perfect.

The assumption that the first 4 rows are given is clearly out of reach, since this would imply the existence of 4 extremely powerful sensor nodes, which can compute their distance to any other sensor node. In a realistic setting, we do not get to choose the entries of $\mathbf{D}$ that are measured. Instead, we can postulate a reasonable model under which the entries of $\mathbf{D}$ are "sampled", and ask whether these "sampled" entries are sufficient to recover the structure of $\mathbf{D}$, even in the presence of noise. The above discussion highlights two points. (i) $\mathbf{D}$ has a lot of structure, and a carefully chosen small sample of its entries will result in accurate reconstruction. Therefore, (ii) the relevant question is what realistic assumptions on the sampling of $\mathbf{D}$ give accurate reconstruction?

We describe a general, realistic model to answer the above question. Introduce an $n \times n$ sampling matrix $\mathbf{P}$ whose (i,j)-th entry $p_{ij} \in [0,1]$ denotes the probability that node i successfully measured its exact distance to node j, i.e., d_{ij}^2 is measured with probability p_{ij}, and is unknown with probability $1 - p_{ij}$. The measurements are corrupted, thus we measure $\tilde{d}_{ij}^2 = d_{ij}^2 + \epsilon_{ij}$ with

probability p_{ij}. Recall that ϵ_{ij} are independent zero mean, bounded variance random variables.

Our model includes the commonly assumed disk model which sets $p_{ij} \approx 1$ if $d_{ij} \leq R$, and $p_{ij} \approx 0$ otherwise. Here R denotes the sensor radius. Our model implicitly allows for operation in obstructed environments and varying signal propagation models, by allowing more general values for p_{ij}.

E. Assumptions

We need to make some assumptions on the p_{ij} in order to prove that localization is, in principle, feasible. Notice that some assumptions on the p_{ij} are clearly necessary in order to give any provable guarantees for localization. For example, if all but $O(1)$ of the p_{ij} are equal to zero, localization is impossible. We state our assumptions and defer a detailed discussion of their plausibility to Section VI, after the presentation of our reconstruction algorithm.

Assumption 1: All the p_{ij}'s are known.

Even though this assumption sounds quite strong, we will argue that it is essentially implicit in existing literature. More importantly, it is actually feasible to get realistic, accurate estimates of the p_{ij} in practical settings.

Assumption 2: $p_{ij} \geq p_\epsilon > 0$, for all $i, j = 1 \ldots n$, for some small positive constant p_ϵ.

In words, we assume that even far away sensors have a very small, non-zero probability of detecting their distance. This assumption might be true as sensor technology improves, or if the sensors are spread over small, bounded regions.

IV. SVD-RECONSTRUCT

We describe the reconstruction algorithm, which we will analyze in Section V. The algorithm is tantalizingly simple, and is motivated by recent important results regarding the reconstruction of low-rank matrices [1], [2].

SVD-RECONSTRUCT takes as input a fraction of the entries of $\mathbf{D}$ that are available, i.e., entries of $\mathbf{D}$ that correspond to pairs of nodes that were able to measure their pairwise distances – recall that $\tilde{D}_{ij} = \tilde{d}_{ij}^2 = d_{ij}^2 + \epsilon_{ij}$ is measured with (known) probability p_{ij}. Thus, the input to SVD-RECONSTRUCT is the matrix $\tilde{\mathbf{D}}$ given by

$$\tilde{D}_{ij} = \begin{cases} d_{ij}^2 + \epsilon_{ij} & \text{with probability } p_{ij}, \\ ? & \text{with probability } 1 - p_{ij}. \end{cases}$$

The ? denotes that the entry is unknown. The first step is to construct a new matrix $\mathbf{S}$ with entries

$$S_{ij} = \begin{cases} \frac{d_{ij}^2 + \epsilon_{ij} - \gamma_{ij}(1-p_{ij})}{p_{ij}} & \text{if } d_{ij} \text{ was detected } (p_{ij}), \\ \gamma_{ij} & \text{otherwise } (1 - p_{ij}). \end{cases}$$

$\mathbf{S}$ is well defined since the p_{ij} are known and non-zero. The γ_{ij} are values representing our "best guess" for the distance between nodes i and j, given that the two nodes were not able to detect their distance. These values naturally model side information that is available in practice. Our algorithm works for any choice for the γ_{ij}, e.g., all γ_{ij} might be set to zero. However, better choices for the γ_{ij} can improve the accuracy of the reconstruction. We will quantify this in equation (4), and in Section VII we will demonstrate the experimental performance of the SVD-RECONSTRUCT algorithm for various choices for the γ_{ij}.

The next step is to construct $\mathbf{S}_4$, the best rank 4 approximation to $\mathbf{S}$ (recall that $\mathbf{D}$ has rank at most 4).

Algorithm SVD-RECONSTRUCT

1) Given $\tilde{\mathbf{D}}$, construct $\mathbf{S}$.
2) Construct $\mathbf{S}_4$, the best rank 4 approximation to $\mathbf{S}$, using the Singular Value Decomposition of $\mathbf{S}$.
3) Run MDSLOCALIZE on $\mathbf{S}_4$ to obtain $\tilde{\mathbf{x}}_i$, $i = 1 \ldots n$, which approximate the $\mathbf{x}_i$ up to rotation/reflection and translation.

The entries of $\mathbf{S}$ satisfy two important properties. Their expectation $\mathbf{E}[S_{ij}]$ is equal to d_{ij}^2 for all i and j (recall that the expectation of ϵ_{ij} is equal to zero), and their variance is bounded since the p_{ij} are bounded away from zero; see Section V for details. These two properties will allow us to use the bounds of [1], [2] to prove that $\mathbf{S}_4$, the best rank 4 approximation to $\mathbf{S}$, is "close" to $\mathbf{D}$. More specifically, we shall obtain bounds for $\|\mathbf{D} - \mathbf{S}_4\|_F^2$.

V. ANALYSIS OF SVD-RECONSTRUCT

The main goal of this paper is to lay a formal foundation for localization by giving provably accurate algorithms for reconstructing $\mathbf{D}$ from highly incomplete distance information. We now show that SVD-RECONSTRUCT is one such algorithm. Instrumental to this goal will be the fact that $\mathbf{D}$ has low rank (lemma 1).

The following lemma is crucial to the analysis. Its essential content is that $\mathbf{S}$ is an unbiased estimator for $\mathbf{D}$.

Lemma 2: For all i, j,

$$\mathbf{E}[S_{ij} - D_{ij}] = 0.$$

We give the proof in the Appendix. The lemma holds because of our careful choice of the *scaling factors*

for the entries of $\mathbf{S}$. We now show that $\mathbf{S}_4$ is close to $\mathbf{D}$, which implies that SVD-RECONSTRUCT accurately recovers $\mathbf{D}$.

Lemma 3 (Theorem 1, [2]): Let $\mathbf{S}_4$ be constructed as described in SVD-RECONSTRUCT. Then,

$$\begin{aligned} \|\mathbf{D}-\mathbf{S}_4\|_F &\leq \|(\mathbf{D}-\mathbf{S})_4\|_F \\ &+ 2\sqrt{\|(\mathbf{D}-\mathbf{S})_4\|_F \, \|\mathbf{D}\|_F} \end{aligned}$$

and also

$$\|\mathbf{D}-\mathbf{S}_4\|_2 \leq 2\|(\mathbf{D}-\mathbf{S})_4\|_2 .$$

The above lemma is essentially Theorem 1 of [2], using the fact that $\|\mathbf{D}-\mathbf{D}_4\|_F = \|\mathbf{D}-\mathbf{D}_4\|_2 = 0$. We now present a bound for $\|(\mathbf{D}-\mathbf{S})_4\|_F$. To prove this bound we first need to bound $\|(\mathbf{D}-\mathbf{S})_4\|_2 = \|\mathbf{D}-\mathbf{S}\|_2$. Towards that end we use Theorem 5 of [2].

Lemma 4 (Theorem 5, [2]): Let σ_S^2 denote an upper bound for the variance of the entries of $\mathbf{S}$, or equivalently, $\mathbf{Var}[S_{ij}] \leq \sigma_S^2$ for all $i,j = 1\ldots n$. Then, with probability at least $1-1/(2n)$, for sufficiently large n,

$$\|\mathbf{D}-\mathbf{S}\|_2 \leq 4\sigma_S\sqrt{2n}, \quad (2)$$

$$\|(\mathbf{D}-\mathbf{S})_4\|_F \leq 12\sigma_S\sqrt{2n}. \quad (3)$$

Combining lemmas 3 and 4 we can easily derive a bound on the quality of $\mathbf{S}_4$ as an approximation to $\mathbf{D}$.

Lemma 5: $\mathbf{S}_4$ is a "good" approximation to $\mathbf{D}$, since with probability at least $1-1/(2n)$,

$$\begin{aligned} \|\mathbf{D}-\mathbf{S}_4\|_F &\leq 12\sigma_S\sqrt{2n} + 8\sqrt{\sigma_S\sqrt{2n}\,\|\mathbf{D}\|_F} \\ \|\mathbf{D}-\mathbf{S}_4\|_2 &\leq 8\sigma_S\sqrt{2n}. \end{aligned}$$

See the Appendix for a proof of the above lemma. We now bound the σ_S term in lemmas 4 and 5. We will use the fact that ϵ_{ij} is zero mean and its variance is bounded by σ_ϵ^2. Indeed (for details see Appendix)

$$\mathbf{Var}[S_{ij}] \leq \frac{2}{p_{ij}}\left((d_{ij}^2-\gamma_{ij})^2+\sigma_\epsilon^2\right).$$

Notice that the quality of the bound improves if γ_{ij} is close to d_{ij}. Overall, using $p_{ij} \geq p_\epsilon$ (Assumption 2),

$$\sigma_S^2 \leq \frac{2}{p_\epsilon}\max_{i,j}\left((d_{ij}^2-\gamma_{ij})^2+\sigma_\epsilon^2\right). \quad (4)$$

The following theorem summarizes our results regarding the accuracy of the reconstruction process, and argues that the average reconstruction error per entry decreases *inversely proportional* to the square root of the number of nodes in the sensor network.

Theorem 2: Let $\mathbf{S}_4$ be constructed as described in the SVD-RECONSTRUCT algorithm. Then, with probability at least $1-1/(2n)$,

$$\|\mathbf{D}-\mathbf{S}_4\|_F \leq 12\sigma_S\sqrt{2n} + 8\sqrt{\sigma_S\sqrt{2n}\,\|\mathbf{D}\|_F},$$

where σ_S^2 is bounded as in equation (4). Let $d_{\max}$ denote the $\max_{i,j} d_{ij}$ over all $i,j \in 1\ldots n$. Since $\|\mathbf{D}\|_F \leq nd_{\max}$, assuming that p_ϵ is any small constant,

$$\|\mathbf{D}-\mathbf{S}_4\|_F^2 \leq O(nd_{\max}^4 + n^{3/2}d_{\max}^3).$$

Thus, the average square error per entry in $\mathbf{S}_4$ is

$$O(d_{\max}^4/n + d_{\max}^3/\sqrt{n}).$$

Assuming that $d_{\max}$ is independent of n, the error decreases inversely proportional to the $\sqrt{n}$.

VI. DISCUSSION

We briefly discuss the impact of the assumptions of Section III-E in light of the SVD-RECONSTRUCT algorithm. Consider Assumption 1. Traditionally [12], MDSLOCALIZE has been run on a reconstructed distance matrix

$$S_{ij} = \begin{cases} d_{ij}^2+\epsilon_{ij} & \text{if } d_{ij} \text{ was detected,} \\ \gamma_{ij} & \text{otherwise,} \end{cases}$$

where γ_{ij} is the shortest path distance between i and j on the sensor network connectivity graph. In the context of constructing $\mathbf{S}$, this corresponds to setting $p_{ij} \approx 1$ if the distance is measured, and $p_{ij} \approx 0$ otherwise. Thus, the traditional setting implicitly assumes that the p_{ij} are known, i.e., p_{ij} is closely approximated by a step function of d_{ij}.

Our setting is more general, since it admits the possibility that the probability for a sensor to detect its distance to another sensor may smoothly decay. In such a situation, one needs to be more careful in selecting S_{ij}. Specifically, the p_{ij} need to be incorporated into S_{ij}. Note that this automatically happens in the traditional setting because of the assumed form for the p_{ij}. The drawback of this more general, and more realistic setting is that one needs to know the p_{ij}. In practice, this is a reasonable requirement, since prior to deploying the sensors, one can gather a great deal of technical information on the sensors. For example, through rigorous repeated experimentation, one can obtain near exact estimates on how a signal transmitted by a sensor degrades as a function of distance. This suffices to derive simple formulas for the probability p_{ij} based on various random models of the background noise. It turns out that such (unbiased, bounded variance) estimates of the p_{ij} suffice. A detailed discussion of relaxing the requirement that the exact p_{ij} are known is deferred to a full version of this paper.

We now turn our attention to Assumption 2, which states that even far away sensors have some arbitrarily

small, though non-zero probability of detecting their distance. As sensor technology improves such an assumption becomes only a mild restriction. In general, p_{ij} is a continuous, non-linear, decreasing function of the distance d_{ij} between the two nodes i and j. Simple models for the detection probabilities can be derived for RF sensors [16], based on the fact that the received power decreases inversely proportional to the square of the distance from the source. Since sensors are deployed in a bounded region, the detection probability among a pair of sensors might become very small, however, it remains bounded away from zero.

One may, however, encounter settings where two sensors have essentially zero probability of detecting their distance. For example, if the sensors are so far apart that the signal to noise ratio is too small, then there is no chance that the sensors will detect their distance. Our results do not strictly apply to this setting in the global sense, however they do apply in the local sense. Specifically, in any "local" region, it is certainly the case that p_ϵ is bounded away from zero. Our results imply that in this local region, which corresponds to a submatrix of the full distance matrix, the distances can be reconstructed accurately. Thus for this particular local region, the positions of the sensors can be recovered in their own local coordinate system. The global localization problem then becomes equivalent to a problem of meshing together several *provably accurate* local "maps" into a single global map, where each local map can be in its own coordinate system.

VII. Evaluation

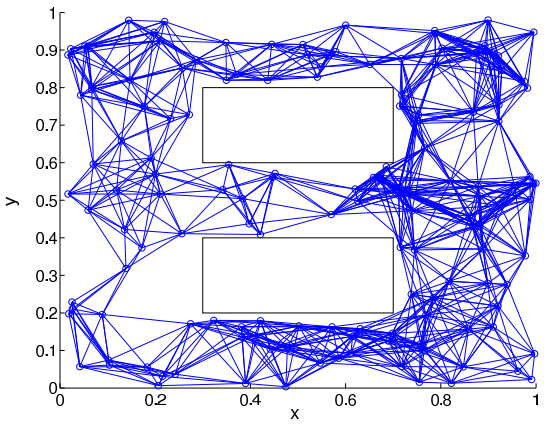

Fig. 1: Example corridor shaped scenario

We evaluate the trends of the reconstruction algorithm on two main types of deployment, uniform and corridor based. We assume that each node detects nodes that are within a radius of $R = 0.165$ with probability one; if two nodes are at distance more than 0.165 the probability that they detect each other is $p_\epsilon = 1/100$. Thus we satisfy Assumption 2, while at the same time the connectivity of the sensor network remains essentially the same.

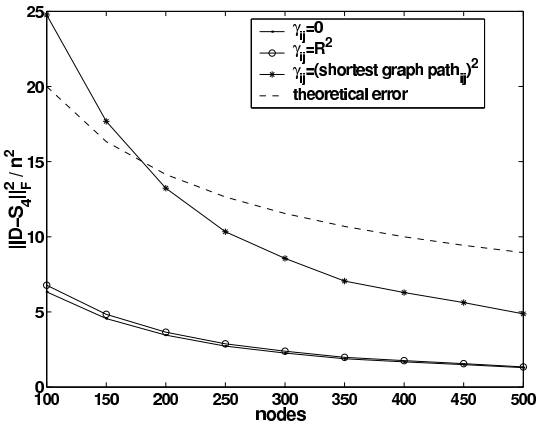

Fig. 2: Uniform Deployment w/o noise

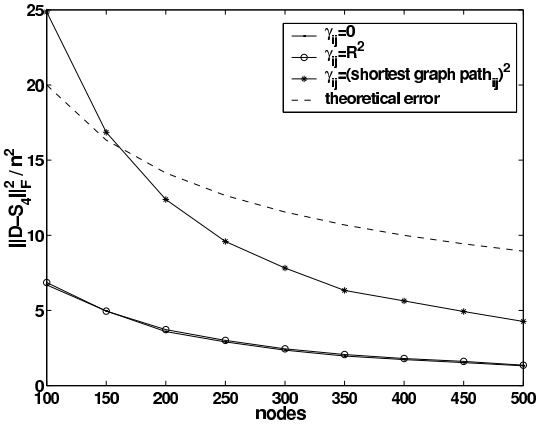

Fig. 3: Uniform Deployment with noise

In the uniform scenarios nodes are randomly scattered in a 1×1 square field following a uniform distribution. In the corridor shaped scenarios, nodes are scattered on a 1×1 square using the same uniform distribution. Corridors are formed by creating two rectangular gaps inside the square field as shown in Fig 1. For each scenario, we evaluate the reconstruction trend for network sizes ranging from 100 to 500 nodes with 10 scenarios for each size. The average connectivity ranges from (roughly) 5 to (roughly) 42. We subsequently plot the average for each size. We evaluate the quality of our reconstruction

for $\gamma_{ij} = 0$, $\gamma_{ij} = R^2$ and $\gamma_{ij} =$ shortest path^2. The reconstruction trends are shown in Figs 2, 3, 4, and 5.

Figures 2 and 4 show the trend when measurements are noise free. Figures 3 and 5 display the same results when distance measurements are corrupted by a noise drawn from a zero mean uniform distribution that is 63% of the actual measurement. Clearly, the plots verify the main result of our work: the accuracy of the localization drops inversely proportional to the square root of the number of nodes in the sensor network. The similarity between the theoretical error bound curve and the curves for the cases $\gamma_{ij} = 0$ and $\gamma_{ij} = R^2$ is indeed striking. As predicted by equation (7), noise does not significantly affect the distance matrix reconstruction error, since the variance of the noise (σ_ϵ^2) is dominated by the first term of equation (7).

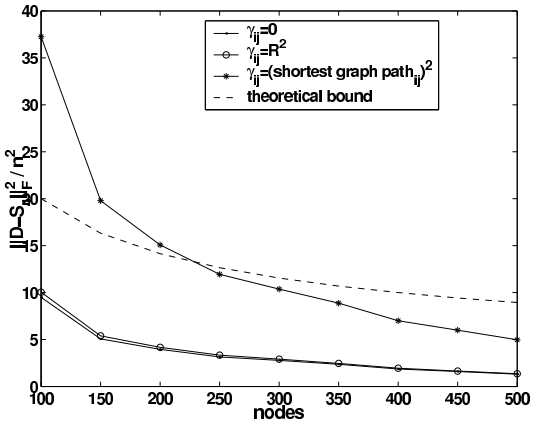

Fig. 4: Corridor Deployment w/o noise

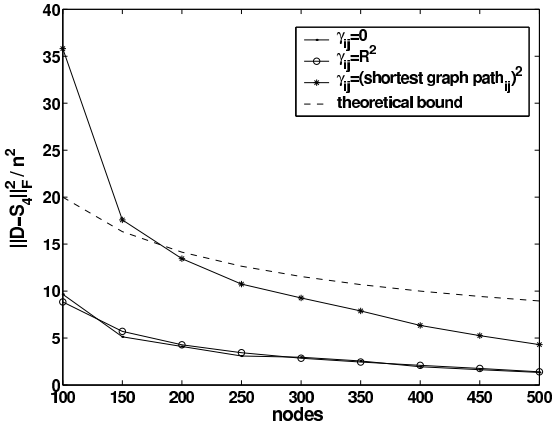

Fig. 5: Corridor Deployment with noise

Finally, we evaluate the SVD-reconstruct algorithm on the following deployment scenario. Consider a situation where two different types of sensors S_1 and S_2 are deployed in adversarial environments, where even though two sensors are within range of each other they might still fail to detect and measure their pairwise distance. Let sensors of type S_1 fail with probability p_1 and sensors of type S_2 fail with probability p_2. These probabilities may be inferred from past deployment experience. We assume that the radius of either type of sensors is R. We scatter sensor nodes of both types uniformly at random over a 1×1 square field.

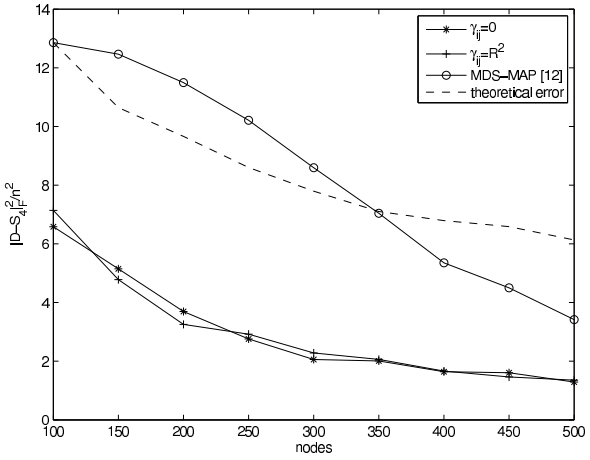

Fig. 6: Comparison with MDS-MAP ($p_1 = \frac{2}{3}, p_2 = \frac{3}{4}, R = 0.1$)

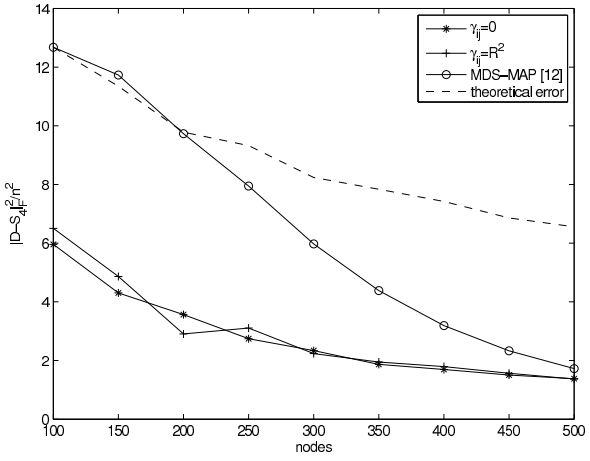

Fig. 7: Comparison with MDS-MAP ($p_1 = \frac{1}{2}, p_2 = \frac{3}{4}, R = 0.1$)

Now consider the 4 possibilities that arise in this set-

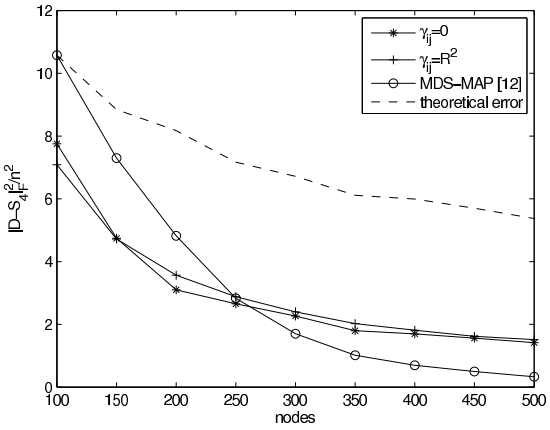

Fig. 8: Comparison with MDS-MAP ($p_1 = \frac{1}{2}, p_2 = \frac{3}{4}, R = 0.165$)

ting. If two sensor nodes of type S_1 are within distance R from each other, they will detect their distance with probability $(1-p_1)^2$; if two sensor nodes of type S_2 are within distance R from each other, they will detect their distance with probability $(1-p_2)^2$; if one sensor node of type S_1 and one sensor node of type S_2 are within distance R from each other, they will detect their distance with probability $(1-p_1)(1-p_2)$; if two sensor nodes of any type are farther than R they will detect their distance with a small, fixed probability $1/100$ (we do not account for individual failure probabilities in this case). Figures 6 and 7 show that for $R = 0.1$ and two different choices for the failure probabilities p_1 and p_2 the SVD-Reconstruct outperforms the MDS-MAP algorithm of [12]. This effect is particularly pronounced in sparse deployments, and is due to the careful rescaling of the known distances by the apriori known failure probabilities. However, as R increases (Figure 8), the comparative advantage of SVD-Reconstruct decreases; in particular, for dense deployments the MDS-MAP algorithm of [12] seems to marginally outperform SVD-Reconstruct.

VIII. Conclusions and Future Work

In this paper we described a first step towards provable algorithms for sensor network localization, by demonstrating that – under some assumptions – reconstruction of Euclidean distance matrices from partial information is, in principle, feasible. Clearly, many important questions remain open. Our current work focuses on four directions. *(i)* We seek to relax the assumptions of Section III-E. *(ii)* We investigate the error bounds of applying the MDSLocalize algorithm on the *reconstructed* distance matrix $\mathbf{S}_4$. *(iii)* We investigate fully distributed, gossip-based protocols for MDSLocalize and SVD-Reconstruct, with provable running time and message size guarantees. *(iv)* We intend to evaluate these algorithms on a real testbed at ENALAB at Yale University.

References

[1] D. Achlioptas and F. McSherry, *Fast Computation of Low Rank Matrix Approximations*, Proceedings of the 33rd Annual ACM Symposium on Theory of Computing, pp. 611–618, 2001.

[2] D. Achlioptas and F. McSherry, *Fast Computation of Low Rank Matrix Approximations*, submitted.

[3] P. Biswas and Y. Ye, *Semidefinite Programming for Ad-Hoc Wireless Localization*, Proceedings of Information Processing in Sensor Networks, pp. 46 - 53, USA.

[4] K.K. Chintalapoudi, A. Dhariwal, R. Govindan, and G. Sukhatme, *On the Feasibility of Ad-Hoc Localization Systems*, USC Technical Report No. 03-796, 2003.

[5] P. Drineas and R. Kannan, *Pass efficient algorithms for approximating large matrices*, Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 223–232, 2003.

[6] P. Drineas, R. Kannan and M.W. Mahoney, *Fast Monte Carlo Algorithms for Matrices III: Computing a Compressed Approximate Matrix Decomposition*,Yale University, Department of Computer Science, YALEU/DCS/TR-1271, 2004.

[7] D. Goldenberg, A. Krishnamurthy, W. Maness, Y. R. Yang, A. Yound, and A. Savvides, *Network Localization in Partially Localizable Networks*, to appear in Proceedings of IEEE INFOCOM, 2005.

[8] J. Xi, H. Zha, *Sensor Positioning in Wireless Ad-hoc Sensor Networks Using Multidimensional Scaling*, Proceedings of IEEE INFOCOM, 2004.

[9] Tzu-Chen Liang, T. Wang, and Y. Ye, *A Gradient Search Method to Round the Semidefinite Programming Relaxation Solution for Ad-Hoc Wireless Sensor Network Localization*, working paper available at http://www.stanford.edu/ yyye/formal-report5.pdf.

[10] D.Niculescu and B. Nath, *VOR Base Stations for Indoor 802.11 Positioning*, Proceeding of IEEE MOBICOM, 2004.

[11] A. Savvides, H. Park, and M. B. Srivastava, *The n-hop Multilateration Primitive for Node Localization Problems*, Proceedings of IEEE MOBICOM, 443–451, 2003.

[12] Y. Shang, W. Ruml, Y. Zhang, and M. Fromherz, *Localization from mere connectivity*, ACM MobiHoc, pp. 201–212, 2003.

[13] Y. Shang and W. Ruml, *Improved MDS-Based Localization*, Proceedings of IEEE INFOCOM, 2004.

[14] A. M. C. So and Y. Ye, *Theory of Semidefinite Programming for Sensor Network Localization*, working paper available at http://www.stanford.edu/ yyye/local-theory.pdf.

[15] R. Stoleru and J. Stankovic, *Probability Grid: A Location Estimation Scheme for Wireless Sensor Networks*, Proceedings of Sensor and Ad-Hoc Communications and Networks Conference (SECON), 2004.

[16] M. Zuniga and B. Krishnamachari, *Analyzing the Transitional Region in Low Power Wireless Links*, Proceedings of Sensor and Ad-Hoc Communications and Networks Conference (SECON), 2004.

APPENDIX

Proof of Theorem 1: After the first step of the algorithm, $\boldsymbol{\tau}(\mathbf{D})$ is an $n \times n$ matrix whose (i,j)-th entry is equal to the inner product $\left(\mathbf{x}_i^T - (1/n)\mathbf{1}_n^T\mathbf{X}\right)\left(\mathbf{x}_j^T - (1/n)\mathbf{1}_n^T\mathbf{X}\right)^T$. In words, the (i,j)-th entry of $\boldsymbol{\tau}(\mathbf{D})$ is equal to the inner product of the coordinate vectors corresponding to the i-th and the j-th sensors, translated in a coordinate system whose origin is the point $(1/n)\sum_{i=1}^n \mathbf{x}_i$. Notice that

$$\mathbf{D} = \mathbf{1}_n\mathbf{z}^T + \mathbf{z}\mathbf{1}_n^T - 2\mathbf{X}\mathbf{X}^T, \tag{5}$$

where $\mathbf{z}$ is an $n \times 1$ vector whose i-th element is equal to $\|\mathbf{x}_i\|^2$. Then,

$$\begin{aligned}\boldsymbol{\tau}(\mathbf{D}) &= -\frac{1}{2}\mathbf{L}(\mathbf{1}_n\mathbf{z}^T + \mathbf{z}\mathbf{1}_n^T - 2\mathbf{X}\mathbf{X}^T)\mathbf{L} = \\ &= \left(\mathbf{X} - (1/n)\mathbf{1}_n\mathbf{1}_n^T\mathbf{X}\right)\left(\mathbf{X} - (1/n)\mathbf{1}_n\mathbf{1}_n^T\mathbf{X}\right)^T\end{aligned}$$

Notice that $\boldsymbol{\tau}(\mathbf{D})$ is a symmetric positive semidefinite matrix of rank at most 2 and its Singular Value Decomposition (computed at the second step of the algorithm) has the same left and right singular vectors. Thus,

$$\tilde{\mathbf{X}} = \mathbf{U}_2\boldsymbol{\Sigma}_\mathbf{2} = \left(\mathbf{X} - (1/n)\mathbf{1}_n\mathbf{1}_n^T\mathbf{X}\right)W, \tag{6}$$

for some 2×2 orthonormal matrix W. Clearly, $(1/n)\mathbf{1}_n^T\mathbf{X} = (1/n)\sum_{i=1}^n \mathbf{x}_i$ is the translation and W is the rotation/reflection. Thus, up to rotation/reflection and translation, we have recovered the original coordinates $\mathbf{X}$.

Proof of Lemma 2:

$$\begin{aligned}\mathbf{E}[S_{ij}] &= \mathbf{Pr}[\epsilon_{ij} = \epsilon] \\ &\cdot \left(\frac{d_{ij}^2 + \epsilon - \gamma_{ij}}{p_{ij}}p_{ij} + (1-p_{ij})\gamma_{ij}|\epsilon_{ij} = \epsilon\right) \\ &= d_{ij}^2 = D_{ij}.\end{aligned}$$

Proof of Lemma 4: The first part of the lemma is an instantiation of Theorem 5 of [2]. For the second part, notice that

$$\begin{aligned}\|(\mathbf{D}-\mathbf{S})_4\|_F^2 &= \sum_{i=1}^4 \sigma_i^2\left((\mathbf{D}-\mathbf{S})_4\right) \\ &\leq 4\sigma_1^2\left((\mathbf{D}-\mathbf{S})_4\right) \\ &= 4\left\|(\mathbf{D}-\mathbf{S})_4\right\|_2^2 \\ &= 4\left\|\mathbf{D}-\mathbf{S}\right\|_2^2 \\ &\leq 128\sigma_S^2 n,\end{aligned}$$

and the lemma follows by taking square roots of the two sides.

Bounding the variance of the entries of S_{ij} (σ_S^2):

$$\begin{aligned}&\mathbf{Var}[S_{ij}] = \\ =\ &\mathbf{Var}[D_{ij} - S_{ij}] = \mathbf{E}\left[(D_{ij} - S_{ij})^2\right] \\ =\ &\mathbf{Pr}[\epsilon_{ij} = \epsilon]\left(p_{ij}\left(\frac{d_{ij}^2 + \epsilon - \gamma_{ij}(1-p_{ij})}{p_{ij}} - d_{ij}^2\right)^2\right. \\ +\ &\left.(1-p_{ij})\left(\gamma_{ij} - d_{ij}^2\right)^2|\epsilon_{ij} = \epsilon\right) \\ =\ &\mathbf{Pr}[\epsilon_{ij} = \epsilon]\left(p_{ij}\left(\frac{(d_{ij}^2 - \gamma_{ij})(1-p_{ij})}{p_{ij}} + \frac{\epsilon}{p_{ij}}\right)^2\right. \\ +\ &\left.(1-p_{ij})\left(\gamma_{ij} - d_{ij}^2\right)^2|\epsilon_{ij} = \epsilon\right) \\ \leq\ &\frac{2(d_{ij}^2 - \gamma_{ij})^2(1-p_{ij})^2}{p_{ij}} + \frac{2\mathbf{E}\left[\epsilon_{ij}^2\right]}{p_{ij}} \\ +\ &(1-p_{ij})\left(\gamma_{ij} - d_{ij}^2\right)^2 \\ =\ &\frac{(d_{ij}^2 - \gamma_{ij})^2(1-p_{ij})(2-p_{ij})}{p_{ij}} + \frac{2\sigma_\epsilon^2}{p_{ij}} \\ \leq\ &\frac{2}{p_{ij}}\left((d_{ij}^2 - \gamma_{ij})^2 + \sigma_\epsilon^2\right).\end{aligned}$$

Notice that the quality of the bound improves if γ_{ij} is close to d_{ij}. Overall,

$$\sigma_S^2 \leq \max_{i,j}\frac{2}{p_{ij}}\left((d_{ij}^2 - \gamma_{ij})^2 + \sigma_\epsilon^2\right). \tag{7}$$