

# A remark on global positioning from local distances

Amit Singer\*

Department of Mathematics, Program in Applied Mathematics, Yale University, New Haven, CT 06520

Communicated by Ronald R. Coifman, Yale University, New Haven, CT, October 18, 2007 (received for review August 17, 2007)

**Finding the global positioning of points in Euclidean space from a local or partial set of pairwise distances is a problem in geometry that emerges naturally in sensor networks and NMR spectroscopy of proteins. We observe that the eigenvectors of a certain sparse matrix exactly match the sought coordinates. This translates to a simple and efficient algorithm that is robust to noisy distance data.**

distance geometry | eigenvectors | multidimensional scaling | sensor networks

Determining the configuration of  $N$  points  $r_i$  in  $\mathbb{R}^p$  (e.g.,  $p = 2, 3$ ) given their noisy distance matrix  $\delta_{ij}$  is a long-standing problem. For example, the points  $r_i$  may represent coordinates of cities in the United States. Distances between nearby cities, such as New York and New Haven, are known, but no information is given for the distance between New York and Chicago. In general, for every city, we are given distance measurements to its  $k \ll N$  nearest neighbors or to neighbors that are, at most,  $\delta$  away. The distance measurements  $\delta_{ij}$  of  $d_{ij} = \|r_i - r_j\|$  may also incorporate errors. The problem is to find the unknown coordinates  $r_i = (x_i, y_i)$  of all cities from the noisy local distances.

Even in the absence of noise, the problem is solvable only if there are enough distance constraints. By solvable, we mean that there is a unique set of coordinates satisfying the given distance constraints up to rigid transformations (rotations, translations, and reflection). Alternatively, we say that the problem is solvable only if the underlying graph consisting of the  $N$  cities as vertices and the distance constraints as edges is rigid in  $\mathbb{R}^p$  (see, e.g., ref. 1). Graph rigidity had drawn a lot of attention over the years, see (1–5) for some results in this area.

When all possible  $N(N - 1)/2$  pairwise distances are known, the corresponding complete graph is rigid, and the coordinates can be computed by using a classical method known as multidimensional scaling (MDS) (6–8). The underlying principle of MDS is to use the law of cosines to convert distances into an inner product matrix, whose eigenvectors are the sought coordinates. The eigenvectors computed by MDS are also the solution to a specific minimization problem. However, when most of the distance matrix entries are missing, this minimization becomes significantly more challenging because of the low rank- $p$  constraint that is not convex. Various solutions to this constrained optimization problem have been proposed, including relaxation of the rank constraint by semidefinite programming (SDP) (9), regularization (10, 11), smart initialization (12), expectation maximization (13), and other relaxation schemes (14). Recently, the eigenvectors of the graph Laplacian (15) were used to approximate the large scale SDP problem by a much smaller one, noting that linear combinations of only a few Laplacian eigenvectors can well approximate the coordinates.

The problem of finding the global coordinates from noisy local distances arises naturally in localization of sensor networks (16–18) and protein structuring from NMR spectroscopy (19, 20), where noisy measurements of neighboring hydrogen atoms are inferred from the nuclear Overhauser effect (NOE). The theory and practice involves many different areas, such as machine learning (15), optimization techniques (14) and rigidity theory (1, 4, 5, 17, 21, 22). We find it practically impossible to give a fair account of all of the relevant literature and recommend the interested reader to see the references within those listed.

In this article, we describe locally rigid embedding (LRE), a simple and efficient algorithm to find the global configuration from locally noisy distances, under the simplifying assumption of local rigidity. We assume that every city, together with its neighboring cities, forms a rigid subgraph that can be embedded uniquely (up to a rigid transformation). That is, there are enough distance constraints among the neighboring cities that make the local structure rigid. The “pebble game” (23) is a fast algorithm to determine graph rigidity. Thus, given a dataset of distances, the local rigidity assumption can be tested quickly by applying the pebble game algorithm  $N$  times, once for each city. In *Summary*, we discuss the restrictiveness of the local rigidity assumption (see refs. 17, 21, 22 for conditions that ensure rigidity) as well as possible variants of LRE when the assumption does not hold.

The essence of LRE follows the observation that one can construct an  $N \times N$  sparse weight matrix  $W$  that has the property that the coordinate vectors are an eigenspace. The matrix is constructed in linear time complexity and has only  $O(N)$  nonzero elements, which enables efficient computation of that small eigenspace. The matrix  $W$  is formed by preprocessing the local distance information to repeatedly embed each city and its neighboring points (e.g., by using MDS or SDP), which is possible by the assumption that every local neighborhood is rigid. Once the local coordinates are obtained, we calculate weights much like the locally linear embedding (LLE) recipe (24) and its multiple weights (MLLE) modification (25). Similar to recent dimensionality reduction methods (24, 26–28), the motif “think globally, fit locally” (29) is repeated here. This is just another example for the usefulness of eigenfunctions of operators on datasets and their ability to integrate local information to a consistent global solution. In contrast to propagation algorithms that start with some local embedding and incrementally embed additional points while accumulating errors, the global eigenvector computation of LRE takes into account all local information at once, which makes it robust to noise.

## Locally Rigid Embedding

We start by embedding points locally. For every point  $r_i$  ( $i = 1, \dots, N$ ) we examine its  $k_i$  neighbors  $r_{i_1}, r_{i_2}, \dots, r_{i_{k_i}}$  for which we have distance information. We find a  $p$ -dimensional embedding of those  $k_i + 1$  points. If the local  $(k_i + 1) \times (k_i + 1)$  distance matrix is fully known, then this embedding can be obtained by using MDS. If some local distances are missing, then the embedding can be found by using SDP or other optimization methods. The assumption of local rigidity ensures that in the noise-free case, such an embedding is unique up to a rigid transformation. Note that the neighbors are not required to be physically close but can, rather, be any collection of  $k_i$  points that the distance constraints among them make them rigid. This locally rigid embedding gives local coordinates for  $r_i, r_{i_1}, \dots, r_{i_{k_i}}$  up to translation, rotation, and, perhaps, reflection. We define  $k_i$  weights  $W_{i,j}$  that satisfy the following  $p + 1$  linear constraints

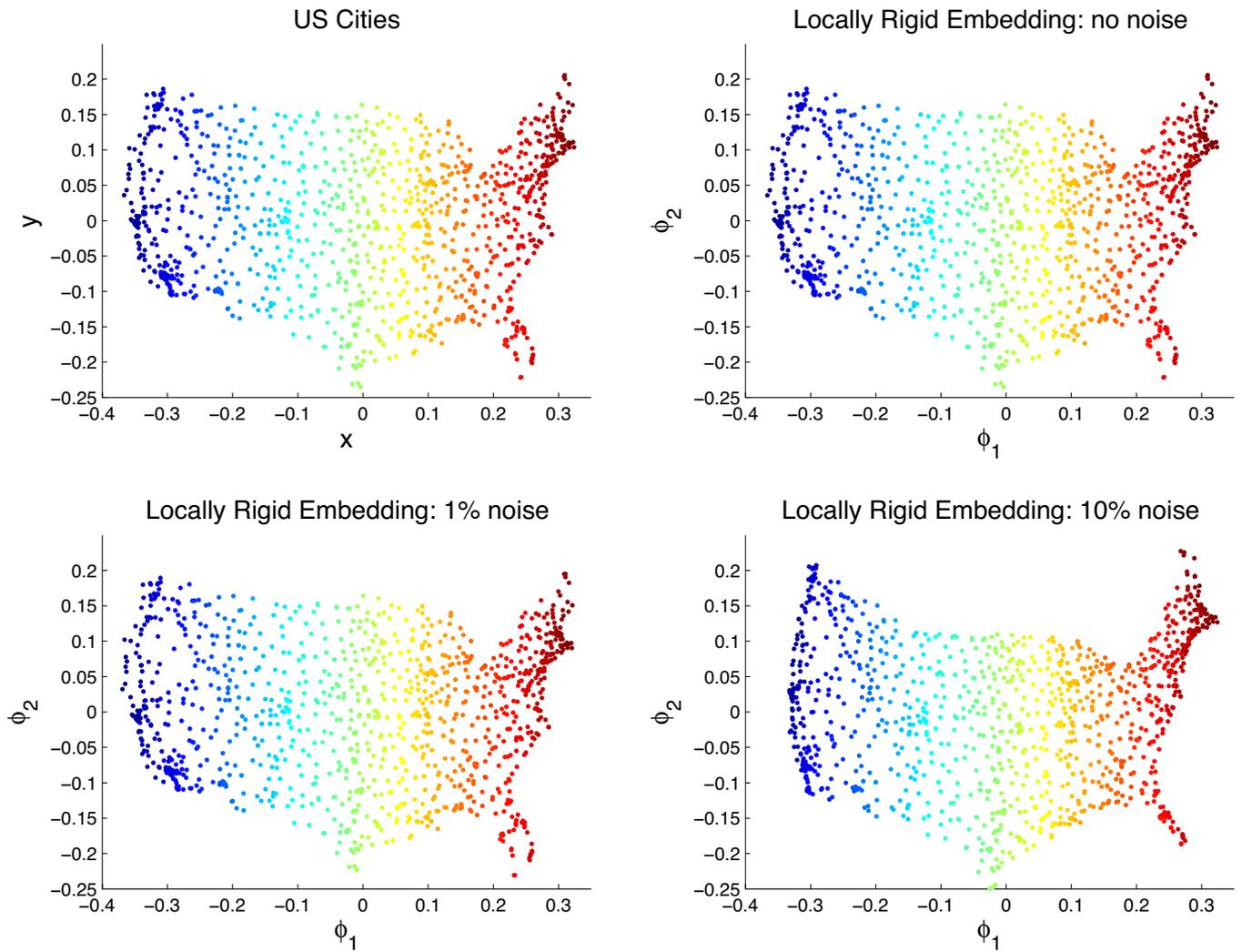
Author contributions: A.S. performed research and wrote the paper.

The author declares no conflict of interest.

\*E-mail: amit.singer@yale.edu.

© 2008 by The National Academy of Sciences of the USA





**Fig. 1.** Locally rigid embedding of US cities. (Upper Left) Map of 1,097 cities in the United States. (Upper Right) The locally rigid embedding obtained from clean local distances. (Lower Left) LRE with 1% noise. (Lower Right) LRE with 10% noise. Cities are colored by their  $x$  coordinate.

$$r_i = A\phi_i, \quad \text{for } i = 1, \dots, N.$$

The squared distance between  $r_i$  and  $r_j$  is

$$d_{ij}^2 = \|r_i - r_j\|^2 = (\phi_i - \phi_j)^T A^T A (\phi_i - \phi_j). \quad [7]$$

Every pair of points for which the distance  $d_{ij}$  is known gives a linear equation for the coefficients of the matrix  $A^T A$ . Solving that equation set results in  $A^T A$ , whose Cholesky decomposition yields  $A$  (up to an orthogonal transformation).

The locally rigid embedding incorporates errors when the distances are noisy. The noise breaks the degeneracy and splits the eigenvalues. If the noise level is small, then we still expect the first  $p$  nontrivial eigenvectors to match the original coordinates, because the small noise does not overcome the spectral gap

$$\min_{\lambda_i \neq 1} |1 - \lambda_i|,$$

where  $\lambda_i$  are the eigenvalues of the noiseless  $W$ . For higher levels of noise, the noise may be large enough so that some of the first  $p$  eigenvalues cross the spectral gap, causing the coordinate vectors to spill over the remaining eigenvectors.

To overcome this problem, we allow  $A$  to be  $p \times m$  instead of  $p \times p$ , with  $m > p$ , but still  $m \ll N$ . In other words, every coordinate is approximated as a linear combination of the  $m$  nontrivial eigenvectors  $\phi^1, \dots, \phi^m$ , whose eigenvalues are the closest to 1. We replace Eq. 7 by a minimization problem for the  $m \times m$  PSD matrix  $P = A^T A$

$$\min \sum_{i \sim j} [(\phi_i - \phi_j)^T P (\phi_i - \phi_j) - \Delta_{ij}^2]^2, \quad [8]$$

where the sum is over all pairs  $i \approx j$  for which there exists a distance measurement  $\Delta_{ij}$ . The least-squares solution for  $P$  is often not valid, when it ends up not being a PSD matrix or when its rank is  $> p$ . It was shown in ref. 15 that the optimization problem (Eq. 8) with the PSD constraint  $P > 0$  can be formulated as SDP by using the Schur complement lemma. The matrix  $A$  is reconstructed from the singular vectors of  $P$  corresponding to the largest  $p$  singular values.

### Numerical Results

We consider a dataset of  $n = 1,097$  cities in the United States, see Fig. 1 (Upper Left). For each city, we have distance measurements to its 18 nearest cities ( $k_i = k = 18$ ). For simplicity, we assume that the  $\binom{18}{2}$  distances between the neighboring cities

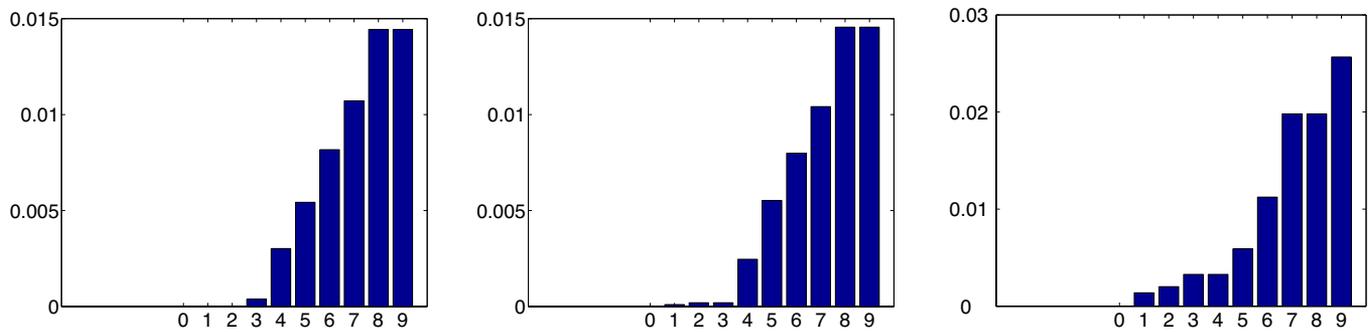


Fig. 2. Numerical spectrum of  $W$  for different levels of noise: Clean distances (Left), 1% noise (Center), and 10% noise (Right).

are also collected, so that MDS is used to locally embed the neighboring cities. We tested the locally rigid embedding algorithm for three different levels of multiplicative Gaussian noise: no noise, 1% noise, and 10% noise. That is,  $\Delta_{ij} = d_{ij}[1 + N(0, \delta^2)]$ , with  $\delta = 0, 0.01, 0.1$ . We used SeDuMi (30) for solving the SDP (Eq. 8).

The behavior of the numerical spectrum for the different noise levels is illustrated in Fig. 2, where the magnitude of  $1 - \lambda_i$  for the 10 eigenvalues that are closest to 1 is plotted. It is apparent in the noiseless case that  $\lambda = 1$  is degenerated with multiplicity 3 corresponding to  $\mathbf{1}$ ,  $x$ ,  $y$ . Note that the spectral gap is quite small. The LRE finds the exact configuration without introducing any errors as can be seen in Fig. 1 (Upper Right).

Adding 1% noise breaks the degeneracy and splits the eigenvalues, and because of the small spectral gap, the coordinates spill over the fourth eigenvector. Therefore, for that noise level, we solved the minimization problem (Eq. 8) with  $P$  being a  $3 \times 3$  matrix ( $m = 3$ ). The largest singular values of  $P$  were 101.7, 42.5, and 0.13, indicating a successful two-dimensional embedding.

When the noise is increased more, the spectrum changes even further. We used  $m = 7$  for the noise level of 10%. The largest singular values of the optimal  $P$  were 76.6, 8.3, 2.0, and 0.002, which implies that the added noise effectively increased the dimension of points from two to three. The LRE distorts boundary points but does a good job in the interior.

### Algorithm

In this section, we summarize the various steps of LRE that should be taken to find the global coordinates from a dataset of vertices and distances by using the symmetric matrix  $S$ .

1. **Input:** A weighted graph  $G = (V, E, \Delta)$ ,  $\delta_{ij}$  is the measured distance between vertices  $i$  and  $j$ , and  $|V| = N$  is the number of vertices.  $G$  is assumed to be locally rigid.
2. Allocate memory for a sparse  $N \times N$  matrix  $S$ .
3. For  $i = 1$  to  $N$ 
  - (a) Set  $k_i$  to be the degree of vertex  $i$ .
  - (b)  $i_1, \dots, i_{k_i}$  are the neighbors of  $i$ :  $(i, i_j) \in E$  for  $j = 1, \dots, k_i$ .
  - (c) Use SDP or MDS to find an embedding  $r_i, r_{i_1}, \dots, r_{i_{k_i}}$  of  $i, i_1, \dots, i_{k_i}$ .
  - (d) Form the  $(p + 1) \times (k_i + 1)$  matrix  $\tilde{R}_i$  following Eq. 5.
  - (e) Compute an orthogonal basis  $\tilde{w}_{i,1}, \dots, \tilde{w}_{i,k_i-p}$  for the nullspace of  $\tilde{R}_i$ .
  - (f) Update  $S \leftarrow S + \sum_{j=1}^{k_i-p} \tilde{w}_{i,j} \tilde{w}_{i,j}^T$ . ( $\tilde{w}_{i,j}$  are obtained from  $\tilde{w}_{i,j}$  by zero padding.)
4. Compute  $m + 1$  eigenvectors of  $S$  with eigenvalues closest to 0:  $\phi_0, \phi_1, \dots, \phi_m$ , where  $\phi_0$  is the all-ones vector.
5. Use SDP as in Eq. 8 to find coordinate vectors  $x_1, \dots, x_p$  as linear combinations of  $\phi_1, \dots, \phi_m$ .

### Summary and Discussion

We presented an observation that leads to a simple algorithm for finding the global configuration of points from their local noisy distances under the assumption of local rigidity. The LRE algorithm that uses the matrix  $W$  is similar to the LLE algorithm (24) for dimensionality reduction. The input to LLE are high-dimensional data points for which LLE constructs the weight matrix  $W$  by repeatedly solving overdetermined linear equation systems. In the global positioning problem the input is not high-dimensional data points, but rather distance constraints that we preprocess (using MDS or SDP) to get a locally rigid low-dimensional embedding. Therefore, the weights are obtained by solving an underdetermined system (Eqs. 1-2) instead of an overdetermined system. The LRE variant that uses the symmetric matrix  $S$  is similar to the MLE algorithm (25) in the sense that it uses more than a single weight vector, taking all basis vectors of the nullspace into account in the construction of  $S$ .

The assumption of local rigidity of the input graph is crucial for the algorithm to succeed. It is reasonable to ask whether or not this assumption holds in practice and how the algorithm can be modified to prevail even if the assumption fails to hold. These are important research questions that remain to be addressed. Even if a few vertices have corresponding subgraphs that are not rigid (nonlocalizable), so their weights are not included in the construction of the stress matrix  $S$ , the matrix  $S$  may still have a nullspace of dimension three. This can happen if the nonlocalizable vertices have enough localizable neighbors. Still, the algorithm fails when there are too many nonlocalizable vertices. For such nonlocalizable vertices, one possible approach is to consider larger subgraphs, such as the one containing all two-hop neighbors (the neighbors of the neighbors), hoping that the larger subgraph would be rigid. A different approach may be to consider smaller subgraphs, hoping that the removal of some neighbors would make the subgraph rigid. This is obviously possible if the vertex is included in a triangle (or some other small simple rigid subgraph) but at risk that the small subgraph would be too small to be rigidly connected to the remaining graph. It is not clear which approach would work better and requires numerical experiments. However, this is beyond the scope of the current article. There are many other interesting questions that follow from both the theoretical and algorithmic aspects, such as proving error bounds, finding other suitable operators for curves and surfaces, and dealing with specific constraints that arise from real-life datasets.

**ACKNOWLEDGMENTS.** I thank Ronald R. Coifman, who inspired this work. The center of mass operator emerged from my joint work with Yoel Shkolnisky on CryoEM. Dan Spielman is acknowledged for designing the symmetrized weight matrix. I would also like to thank Jia Fang, Yosi Keller, Yuval Kluger, and Richard Yang for valuable discussions on sensor networks, NMR spectroscopy, and rigidity. Finally, thanks are due to Rob Kirby for making valuable suggestions that improved the clarity of the manuscript. This work was supported by the National Geospatial Intelligence Agency (NGA) NGA University Research Initiatives, 2006.

