SYSTEMS OPTIMIZATION LABORATORY
DEPARTMENT OF MANAGEMENT SCIENCE AND ENGINEERING
STANFORD UNIVERSITY
STANFORD, CALIFORNIA 94305-4026

A Gradient Search Method to Round the Semidefinite Programming Relaxation Solution for Ad Hoc Wireless Sensor Network Localization

by
Tzu-Chen Liang, Ta-Chung Wang, and Yinyu Ye

TECHNICAL REPORT SOL 2004-2

December 2004

A Gradient Search Method to Round the Semidefinite Programming Relaxation Solution for Ad Hoc Wireless Sensor Network Localization

Tzu-Chen Liang[*], Ta-Chung Wang [†] and Yinyu Ye[‡]

August 28, 2004; Updated October 1, 2004

Abstract

In this report we develop an efficient and effective procedure to solve the distance geometry problem. This method is based on the semidefinite programming (SDP) relaxation proposed in [5, 6], and further improved by a gradient-based local search method. We first develop two alternative SDP relaxations for the distance geometry problem, which represent the (weighted) maximum-likelihood estimation (WMLE). Then, using the SDP relaxation solution as the initial point, we apply a gradient-based search method to further reducing the estimation error. We show that the gradient search method permits an exact line-search and it can always improve the SDP solution with or without distance measurement noises. A checkable bound of suboptimality can be used to ensure the solution quality. We demonstrate the effectiveness of the method from solving the 2-dimensional ad hoc wireless sensor network localization problem. Even for large scale problems with thousands of sensors, a satisfactory localization can be found on a single PC in few minutes.

1 Introduction

Recently, semidefinite programming (SDP) relaxations have been used for solving or approximating hard problems. The solution obtained by an SDP

[*]Department of Aeronautics & Astronautics, Stanford University, Stanford, CA 94305.

[†]Department of Aeronautics & Astronautics, Stanford University, Stanford, CA 94305.

[‡]Management Science and Engineering and, by courtesy, Electrical Engineering, Stanford University, Stanford, CA 94305. E-mail: yinyu-ye@stanford.edu.

relaxation is in general not optimal or even feasible for the original problem. Therefore a rounding technique must be applied to round the SDP solution to a suboptimal and feasible one for the original problem. In this report we develop a rounding method to solve the distance geometry problem, especially the 2-dimensional localization problem for ad hoc wireless sensor networks. All the results can be extended to solving higher dimensional distance geometry problems without any difficulty. However, for simplicity, the 2-dimensional localization problem will be used throughout this report.

A typical sensor network consists of a large number of sensors which are densely deployed in a geographical area. Sensors collect the local environmental information such as temperature or humidity and can communicate to each other. The advance of Micro-electro-mechanical system (MEMS) and wireless communication technology have made the sensor network a low cost and high efficiency method for environment observation. A successful example of habitat monitoring system is the Great Duck Island(GDI) [18] system. Other applications like battlefield surveillance and moving object tracking also attract large research efforts.

One key issue of sensor network research is to accurately locate the position of each sensor in a network. In 1996, the U.S Federal Communications Commission (FCC) required that all wireless service providers give location information to the Emergency 911 services [12]. Cellular base stations are used to locate mobile telephone users within a cell [9]. The FCC wireless E911 program is divided into two parts - Phase I and Phase II. Phase I requires carriers, upon appropriate request by a local Public Safety Answering Point (PSAP), to report the telephone number of a wireless 911 caller and the location of the antenna that received the call. Phase II requires wireless carriers to provide far more precise location information, within 50 to 100 meters in most cases. The FCC also established a four-year rollout schedule for Phase II, beginning October 1, 2001 and to be completed by December 31, 2005.

Usually sensor networks consist of a large number of sensors and it is too expensive to use Global Positioning System (GPS)[16] to locate their positions. Therefore positioning methods using their communication and neighboring distance measurements are developed. The ad hoc sensor network localization problem is: assuming the accurate positions of some nodes, called anchors, are known, how to use them and partial pair-wise distance measurements to locate the positions of all sensor nodes in the sensor network, see [4, 8, 10, 13, 14, 22, 21, 20, 23]. The difficulty of this problem arising in various faces: Firstly, the distance measurements always have some noise or uncertainty. Since the effect of the measurement uncertainty usually depends on the geometrical relationship between sensors and is not known a priori, the unbiased model is not easy to build. Furthermore, even if the distance measurements are perfectly accurate, the sufficient condition for this sensor network to be localizable cannot be identified easily, see [11, 15].

In practice, increasing the number of anchors and the radio range of communication usually make localization result more satisfactory, but this also implies growing of the network cost. Hence reliable algorithms to solve the localization problem, as well as the knowledge of how to deploy a sensor network such that the sensors can be located accurately, are both important research issues. Various techniques have been developed to overcome the measurement uncertainties. Most of these methods are based on minimizing some global error functions (called the objective function), which can be different when the model of uncertainty changes. Depending on the kind of optimization problem being formulated, the characteristic and computation complexity varies. For example, the maximum likelihood estimation for sensor network localization problem is a non-convex optimization problem. Existing algorithms are restricted in problem sizes, see [19]. On the other hand, if we relax the distance constraints, the problem can be formulated as a second order cone program (SOCP) and superiorly scalable algorithms can be applied ([10, 26]). However, the solution is acceptable only when the anchors are densely deployed on the boundary of the sensor network. Another approach is to allow a large radio range for every sensor so that they can communicate with many other sensors or anchors. But this needs a higher consumption of battery power for each sensor.

Most of existing approaches are corresponding to various degrees of relaxation of the original problem. One recent relaxation is the SDP relaxation developed in [5, 6], which is closely related to SDP relaxations for other distance geometry problems, see [1, 2, 7, 17]. It was subsequently proved that the localization problem can be solved in polynomial time under uniquely localizable assumption [24]. However, if the sensor network is not uniquely localizable, there must exist a higher (rank) dimensional localization that minimizes the objective function, and SDP relaxation always produces this maximal (rank) dimensional solution. A classical way to obtain a 2-dimensional solution is to project the high dimensional solutions on the 2-dimensional space, which seems generally not satisfactory. Moreover, in a real ad hoc wireless sensor network, the distance measurements inevitably have noises or uncertainties. Using this uncertain distance information, we can only formulate the localization problem as an optimization problem that minimizes the difference between the measured distances and the distances of the estimated locations. The max-rank property of the SDP relaxation [24] makes the relaxation solution almost always lies in a higher (rank) dimensional space, and no robust and reliable method is available to round the high dimensional SDP solution.

In this report we develop a new technique to round the SDP relaxation solution using a gradient-based descent method. One should notice that gradient method generally does not work when the problem itself is non-convex. However, our results show that the SDP relaxation solution establishes a globally good initial point for the gradient-based local search method; and a better solution can always be found after applying the gradient local search. It is demonstrated that the improved solution is very close to the optimal one by comparing to the SDP lower bound of the minimal objective value.

2 SDP relaxations

We would like to first introduce some notations and the mathematical formulation of ad hoc sensor network localization problem. For two symmetric matrices A and B, $A \succeq B$ means $A - B \succeq 0$, i.e. $A - B$ is a positive semidefinite matrix. We use I_d, e and $\mathbf{0}$ to denote the $d \times d$ identity matrix, the vector of all ones and the vector of all zeros, whose dimensions will be clear in the context. e_i is a vector with all zeros except its ith entry, which is one. The 2-norm of a vector x is denoted as $\|x\|$.

The following notations are related to the ad hoc sensor network localization problem. In an ad hoc sensor network in $\Re^2$ with m anchors and n sensors, an anchor is a node whose location a_k in $\Re^2$, $k = 1, 2, ..., m$, is known, and a sensor is a node whose location has yet to be decided and denoted by x_j in $\Re^2$, $j = 1, 2, ..., n$. For a pair of sensors x_j and x_i, their Euclidean distance is denoted as d_{ji}. Similarly, for a pair of sensor x_j and anchor a_k their Euclidean distance is denoted as d_{jk}. In general, not all pairs of sensor/sensor and sensor/anchor distances are known, so the known pairwise distance of sensor/sensor and sensor/anchor are denoted as $(j, i) \in N_x$ and $(j, k) \in N_a$, respectively. One should notice that this problem is in fact a 2-dimensional case of general distance geometry problems. The algorithms we propose are not restricted in $\Re^2$, but for illustration all our examples are chosen from the localization problem in $\Re^2$.

Mathematically, the localization problem in $\Re^2$ can be stated as: given m anchor locations $a_k, k = 1, 2, ..., m$ and some distance measurements $d_{ji}, (j, i) \in N_x, d_{jk}, (j, k) \in N_a$, find $x_j, j = 1, 2, ..., n$, the locations of n

sensors, such that

$$\begin{aligned}\|x_j - x_i\|^2 = d_{ji}^2, \forall (j,i) \in N_x \\ \|x_j - a_k\|^2 = d_{jk}^2, \forall (j,k) \in N_a\end{aligned}$$

Let matrix $X = [x_1,\ x_2,\ ...,\ x_n]$. Then, the problem can be written in matrix form,

$$\begin{array}{ll} \text{find} & X \in \Re^{2\times n}, Y \in \Re^{n\times n} \\ & (e_i - e_j)^T Y (e_i - e_j) = d_{ji}^2, \forall (j,i) \in N_x \\ \text{s.t.} & (a_k; -e_j)^T \begin{pmatrix} I_2 & X \\ X^T & Y \end{pmatrix} (a_k; -e_j) = d_{jk}^2, \forall (j,k) \in N_a \\ & Y = X^T X \end{array} \tag{1}$$

If we relax the last equality constraint to inequality $Y \succeq X^T X$, it becomes a convex optimization problem in the standard SDP form,

$$\begin{array}{ll} \text{find} & Z \in \Re^{(n+2)\times(n+2)} \\ & (0; e_i - e_j)^T Z (0; e_i - e_j) = d_{ji}^2, \forall (j,i) \in N_x \\ \text{s.t.} & (a_k; -e_j)^T Z (a_k; -e_j) = d_{jk}^2, \forall (j,k) \in N_a \\ & Z \succeq 0 \end{array} \tag{2}$$

where $Z = \begin{pmatrix} I_2 & X \\ X^T & Y \end{pmatrix}$.

This problem has been discussed in [24] in detail, and a condition of uniquely localizability is given. We repeat it here,

Definition 1 *Problem (1) is uniquely localizable if it has a unique feasible solution $\bar{X}$ in $\Re^{2\times n}$ and there is no x_j in $\Re^h, j = 1, ..., n$, where $h > 2$ (excluding the case appending all zeros to $\bar{X}$), such that*

$$\begin{aligned} \|x_j - x_i\|^2 &= d_{ji}^2, \forall (i,j) \in N_x \\ \|x_j - (a_k; \mathbf{0})\|^2 &= d_{jk}^2, \forall (j,k) \in N_a. \end{aligned} \tag{3}$$

The latter condition in the definition says that the problem cannot be localized in a higher dimensional space where anchor points are augmented to $(a_k; \mathbf{0}) \in \Re^h, j = 1, ..., m$. The importance of [24] is it firstly states that if the problem is uniquely localizable, then relaxation problem (2) solves (1) exactly. To tell whether a problem is uniquely localizable or not before solving it is not easy. But once solve the SDP relaxation and observe whether

$Y = X^T X$ in the solution, we immediately know if the problem is uniquely localizable or not.

For the localization problem with measurement noises, the story can be quite different. In general there is no solution to satisfy the constraints in (1). Therefore, an optimization problem is set up to minimize the difference between the measured distance values and the distances formulated from the estimated solutions. In this report, we assume that distance noises are randomly generated according to the following formula,

$$d_{ji} = \hat{d}_{ji}(1+\rho); \tag{4}$$

where $\hat{d}_{ji}$ represents the true distance between nodes (either sensors or anchors) i and j, $\rho \in \mathcal{N}(0, nf^2)$ is a random variable, and nf, called *noisy factor*, is used to adjust the variance of the distance uncertainty.

2.1 Maximum Likelihood Estimation and its SDP relaxation

One approach, called maximum likelihood estimation, is introduced below. Let $d : \Re^{2\times 2} \rightarrow \Re$ be the distance function between a sensor/anchor or sensor/sensor pair. Suppose there are some measurement errors between x_j/a_k and x_j/x_i denoted by ω_{jk} and ω_{ji}, respectively,

$$\begin{aligned} d_{jk} &= d(x_j, a_k) + \omega_{jk}, \forall (j,k) \in N_a \\ d_{ji} &= d(x_j, x_i) + \omega_{ji}, \forall (j,i) \in N_x \end{aligned}$$

where we assume each $\omega_{jk} \sim \mathcal{N}(0, \sigma_{jk}^2)$ and $\omega_{ji} \sim \mathcal{N}(0, \sigma_{ji}^2)$ and they are independent.

Let the maximum likelihood function p to estimate X, using all distance measure information, be

$$\begin{aligned} & p((d_{jk}, (j,k) \in N_a; d_{ji}, (j,i) \in N_x), X) = \\ & \quad \prod_{j,k;(j,k)\in N_a} \frac{1}{2\pi^{\frac{1}{2}}\sigma_{jk}} \exp\left(-\frac{1}{2\sigma_{jk}^2}(d_{jk} - d(x_j, a_k))^2\right) \\ & \quad \prod_{j,i;(j,i)\in N_x} \frac{1}{2\pi^{\frac{1}{2}}\sigma_{ji}} \exp\left(-\frac{1}{2\sigma_{ji}^2}(d_{ji} - d(x_j, x_i))^2\right) \end{aligned}$$

and the maximum likelihood estimation be

$$X_{ml} = \arg\max_X p((d_{jk}, (j,k) \in N_a; d_{ji}, (j,i) \in N_x), X).$$

Then, X_{ml} can be written explicitly as

$$X_{ml} = \arg\min_{X} \left(\begin{array}{c} \sum_{j,k;(j,k)\in N_a} \frac{1}{\sigma_{jk}^2} \left(d_{jk} - d(x_j, a_k)\right)^2 \\ + \sum_{j,i;(j,i)\in N_x} \frac{1}{\sigma_{ji}^2} \left(d_{ji} - d(x_j, x_i)\right)^2 \end{array} \right) \tag{5}$$

Hence the following optimization problem solves the maximum likelihood estimation problem

$$\begin{array}{ll} \min & \sum_{j,k;(j,k)\in N_a} \frac{1}{\sigma_{jk}^2} \epsilon_{jk} + \sum_{j,i;(j,i)\in N_x} \frac{1}{\sigma_{ji}^2} \epsilon_{ji} \\ \text{s.t.} & \begin{array}{c} \left(\|x_j - x_i\| - d_{ji}\right)^2 = \epsilon_{ji}, \forall (j,i) \in N_x \\ \left(\|x_j - a_k\| - d_{jk}\right)^2 = \epsilon_{jk}, \forall (j,k) \in N_a. \end{array} \end{array} \tag{6}$$

If the variance of distance measurements are not known, any reasonable assumption can be applied. Problem (6) is not a convex optimization problem, but we can construct its SDP relaxation problem:

$$\begin{array}{ll} \min & \sum_{j,k;(j,k)\in N_a} \frac{1}{\sigma_{jk}^2} \epsilon_{jk} + \sum_{j,i;(j,i)\in N_x} \frac{1}{\sigma_{ji}^2} \epsilon_{ji} \\ \text{s. t.} & \begin{array}{c} \left(-d_{ji}; 1\right)^T D_{ji} \left(-d_{ji}; 1\right) = \epsilon_{ji}, \forall (j,i) \in N_x \\ \left(-d_{jk}; 1\right)^T D_{jk} \left(-d_{jk}; 1\right) = \epsilon_{jk}, \forall (j,k) \in N_a \\ \left(0; e_j - e_i\right)^T Z \left(0; e_j - e_i\right) = v_{ji}, \forall (j,i) \in N_x \\ \left(a_k; -e_j\right)^T Z \left(a_k; -e_j\right) = v_{jk}, \forall (j,k) \in N_a \\ D_{ji} \succeq 0, \forall (j,i) \in N_x \\ D_{jk} \succeq 0, \forall (j,k) \in N_a \\ Z \succeq 0 \end{array} \end{array} \tag{7}$$

where $Z = \begin{pmatrix} I_2 & X \\ X^T & Y \end{pmatrix}$ and $D_{ji} = \begin{pmatrix} 1 & u_{ji} \\ u_{ji} & v_{ji} \end{pmatrix}, \forall (j,i) \in N_x, D_{jk} = \begin{pmatrix} 1 & u_{jk} \\ u_{jk} & v_{jk} \end{pmatrix}, \forall (j,k) \in N_a$.

2.2 Weighted Maximum Likelihood Estimation and its SDP relaxation

Another formulation to minimize the estimation error is

$$\begin{array}{ll} \min & \sum_{j,i;(j,i)\in N_x} \varepsilon_{ji}^2 + \sum_{j,k;(j,k)\in N_a} \varepsilon_{jk}^2 \\ \text{s.t.} & \begin{array}{c} \|x_j - x_i\|^2 - d_{ji}^2 = \varepsilon_{ji}, \forall (j,i) \in N_x \\ \|x_j - a_k\|^2 - d_{kj}^2 = \varepsilon_{jk}, \forall (j,k) \in N_a \end{array} \end{array} \tag{8}$$

The objective function can be written as,

$$\begin{array}{c}\sum_{j,i;(j,i)\in N_x}(\|x_j - x_i\| + d_{ji})^2(\|x_j - x_i\| - d_{ji})^2 \\ +\sum_{j,k;(j,k)\in N_a}(\|x_j - a_k\| + d_{jk})^2(\|x_j - a_k\| - d_{jk})^2\end{array} \tag{9}$$

If we assume the measurement errors are not too large; $\|x_j - x_i\| \approx d_{ji}$ and $\|x_j - a_k\| \approx d_{jk}$, the objective is approximately equivalent to minimize

$$\sum_{j,i;(j,i)\in N_x} d_{ji}^2(\|x_j - x_i\| - d_{ji})^2 + \sum_{j;k;(j,k)\in N_a} d_{jk}^2(\|x_j - a_k\| - d_{jk})^2$$

Comparing this with the objective function of (6), one can immediately see that if $\sigma_{ji} = \frac{1}{d_{ji}}$ and $\sigma_{jk} = \frac{1}{d_{jk}}$, the solution of (8) is actually the maximum likelihood estimation solution. We call it weighted maximum likelihood estimation.

The SDP relaxation of (8) can be written as

$$\begin{array}{ll} \min & \alpha \\ \text{s.t.} & \begin{array}{c}(0; e_j - e_i)^T Z\,(0; e_j - e_i) - d_{ji}^2 = \varepsilon_{ji}, \forall (j,i) \in N_x \\ (a_k; -e_j)^T Z\,(a_k; -e_j) - d_{jk}^2 = \varepsilon_{jk}, \forall (j,k) \in N_a \\ \left(\sum_{(i,j)\in N_x} \varepsilon_{ji}^2 + \sum_{(i,j)\in N_a} \varepsilon_{jk}^2\right)^{1/2} \le \alpha \\ Z \succeq 0\end{array}\end{array} \tag{10}$$

where $Z = \begin{pmatrix} I_2 & X \\ X^T & Y \end{pmatrix}$. Note that this problem is almost as simple as the SDP relaxation (2), and it has merely one more second order cone constraint on α and εs.

If the distance measurements are exactly correct and the sensor network is uniquely localizable, then all three formulations (2), (7), and (10) solves the true sensor locations. Moreover, both (7) and (10) can deal with the localization problem with distance noises. One may notice the assumption that the noise variance is inversely proportional to the measured distance may not be true in general. However, the reason we still use (10) will be clear when the gradient search method is introduced later.

2.3 The High-Rank Property of SDP Relaxations

When the distance measurements have errors, one may expect (7) or (10) to be good approximations of (6). However, due to the "max-rank" property of the SDP relaxation described in [24], the solution of (7) and (10) may not be satisfactory. This property is illustrated in the following example,

Example 1 *Consider an ad hoc sensor network in* $\Re^2$ *with 3 anchors and 1 sensor. The locations of sensors are at* $a_1 = (1,0)^T, a_2 = (0,1)^T, a_3 = (-1,-1)^T$*. The measured distances between* a_1, a_2, a_3 *and* x *are* $1.020, 1.041, 1.412$ *respectively. By solving the SDP relaxation (7), the objective function value is zero and the optimal* Z^* *is*

$$
\begin{aligned}
Z^* &= \begin{pmatrix} I_2 & x^* \\ x^{*T} & y^* \end{pmatrix} \\
&= \begin{pmatrix} 1 & 0 & 0.0061 \\ 0 & 1 & -0.0571 \\ 0.0061 & -0.0571 & 0.0958 \end{pmatrix}
\end{aligned}
$$

Sine it is not a rank 2 *solution, we know that a higher dimensional localization exists. One way to interpret this solution is: for this only one unknown sensor, the solution space is lifted to* $\Re^3$*, and its projection on* $\Re^2$ *is* $x^* = [0.0061, -0.0571]^T$*. The third coordinate of* x^* *is equal to* $\sqrt{y^* - x^{*T}x^*} = 0.0925$*. This says a point* $\bar{x} = [0.0061, -0.0571, 0.0925]^T \in \Re^3$ *satisfies the three distance constraints.*

In this example we see that when the three distance measurements contradict to each other and there is no localization in $\Re^2$, it is still possible to locate the sensor in a higher dimensional space and to make the objective function value zero. The optimal solution in a higher dimensional space always results in a smaller objective function value than the one constrained in $\Re^2$ even if both of them are nonzero. Hence, because of the relaxation of the rank requirement, the solution is "lifted" to a higher dimensional space.

A main research topic is how to round the higher-dimension (higher rank) SDP solution into a lower-dimension (rank-2) solution. One way is to ignore the augmented dimensions and use the projection x^* as a suboptimal solution, which is the case in [5]. In the rest of the current paper, we will show how to use a gradient-based local search method to round and improve the SDP relaxation solution. The method turns out to be extremely effective and efficient.

3 A Gradient Search Method

3.1 An Inspiring Example

A typical solution using the MLE SDP relaxation (7) for an ad hoc sensor network with 5 anchors and 45 sensors ($m = 5, n = 45$), radio range 0.35, are shown in Figure 1. (The radio range indicates that the distance values between any two nodes are known to the solver if they are below the range; otherwise they are unknown.) The diamonds and the circles indicate the true positions of the anchors and the sensors, respectively. The red stars show the SDP localizations (The same notations will be used in all examples of this report). This problem is uniquely localizable so that all the sensors are located correctly. Now for the same problem, some distance noises are added. Each d_{ji} or d_{jk} below the radio range is collapsed by a Gaussian noise with mean zero and variance equal to 10% of the actual distance (noise factor $nf = 0.10$). The new result is shown in Figure 2. By comparing with the true sensor locations plotted in circles, it is easy to see that most localizations are far from their actual locations. Since this network is uniquely localizable, the localization errors are completely due to the distance measurement errors. A higher dimensional localization is then found by the SDP relaxation solved by SEDUMI [25], and apparently it is not quite acceptable by its projection on the 2-dimensional space. Even worse, the localization errors propagate from the sensors near some anchors to the sensors far from any anchors.

The method we suggest to improve the SDP solution is moving every sensor location along the opposite of its gradient direction of the sum of error square function, which will for sure reduce the error function value. The detail and theoretical background of this approach will be given in the next section. We first demonstrate the basic idea of it. Let us begin from (5) and for simplification all σ_{ji} and σ_{jk} are assumed to be the same. The maximum likelihood estimation is an unconstrained optimization problem if all constraints are substituted into the objective function,

$$\begin{aligned} \min_X \ f(X) := \textstyle\sum_{j,k,(j,k)\in N_a} \left(d_{jk} - \sqrt{(x_j - a_k)^T(x_j - a_k)}\right)^2 \\ + \textstyle\sum_{j,i,(j,i)\in N_x} \left(d_{ji} - \sqrt{(x_j - x_i)^T(x_j - x_i)}\right)^2 \end{aligned} \tag{11}$$

Differentiate f with respect to x_j, one can find the gradient ∂f_{x_j} for a certain sensor x_j,

$$\begin{aligned} \partial f_{x_j} \equiv \tfrac{\partial f}{\partial x_j} \\ = \textstyle\sum_{k,(j,k)\in N_a} \left(d_{jk} - \sqrt{(x_j - a_k)^T(x_j - a_k)}\right) \frac{(x_j-a_k)^T}{\sqrt{(x_j-a_k)^T(x_j-a_k)}} \\ + \tfrac{1}{2}\textstyle\sum_{i,(j,i)\in N_x} \left(d_{ji} - \sqrt{(x_j - x_i)^T(x_j - x_i)}\right) \frac{(x_j-x_i)^T}{\sqrt{(x_j-x_i)^T(x_j-x_i)}} \end{aligned} \tag{12}$$

It is important to notice that ∂f_{x_j} only relates to the sensors and anchors that are connected to (within the radio range) x_j, and they are local information, so that ∂f_{x_j} for every sensor x_j can be solved distributedly.

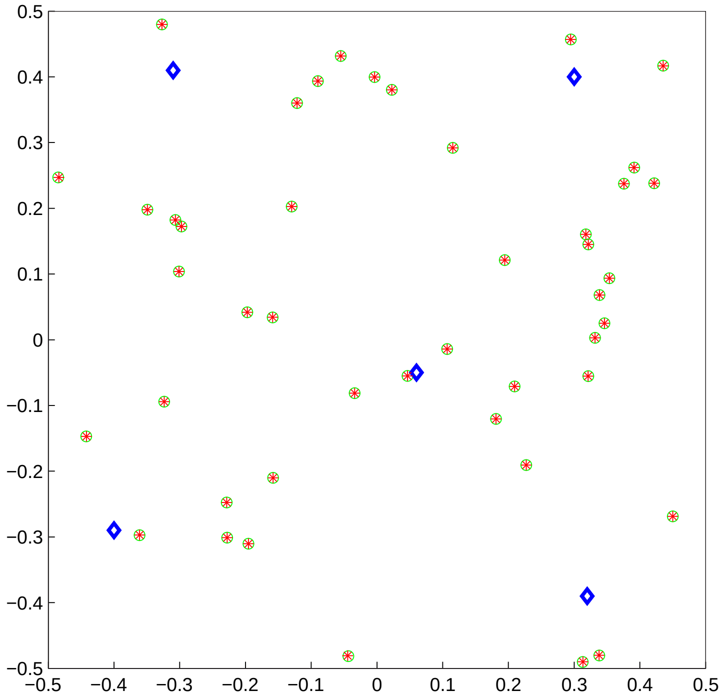

Figure 1: Localization solved by MLE SDP relaxation with no distance measurement error.

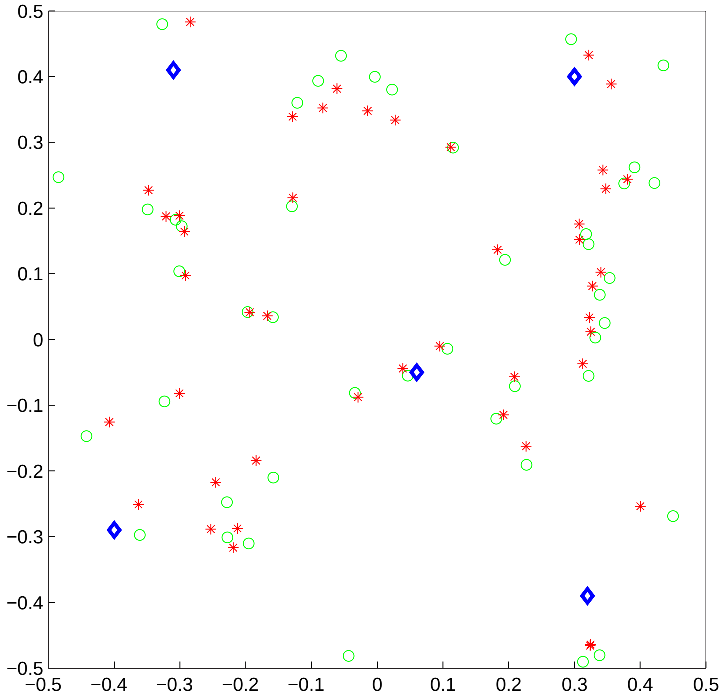

Figure 2: Localization solved by MLE SDP relaxation with 10% distance measurement errors.

The following update rule is applied to improve the iterative solution.

$$x_j \longleftarrow x_j - \alpha \cdot \partial f_{x_j}^T \text{ for } j = 1 \text{ to } n \tag{13}$$

where α is the step size, chosen to be 0.05 for this example. In one gradient step, the method calculates the gradient of each sensor and updates its location by this rule. The SDP localization shown in Figure 2 is used as the initial solution. Figure 3 shows the update trajectories in 50 steps. The red points indicate the new positions of each sensor after each gradient step; contiguous points are connected by blue lines. It can be observed clearly that most sensors are moving toward their actual locations marked by the circles. The final localizations after 50 gradient steps are plotted in Figure 4. Comparing Figure 2 with Figure 4, we see that a much more accurate localization is found.

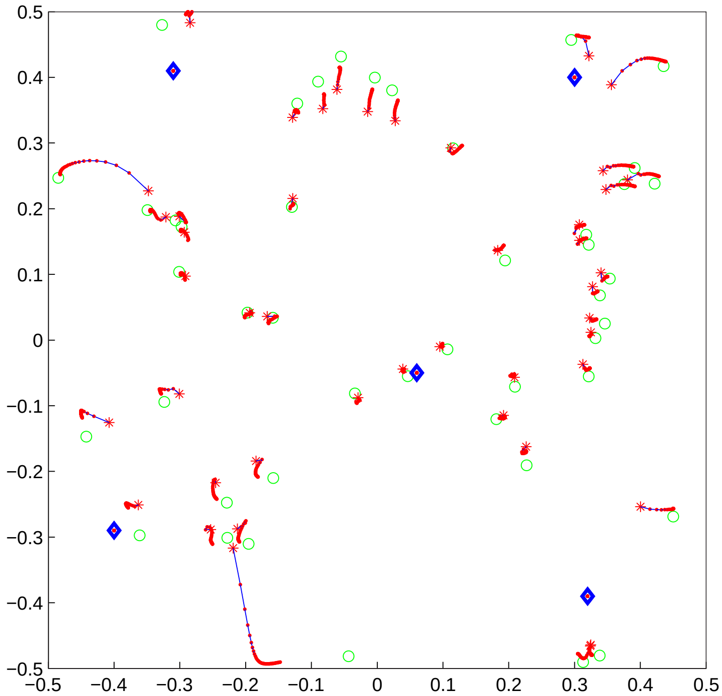

Figure 3: Gradient search trajectories.

To demonstrate that the new localization is indeed better, we can substitute the sensor positions after every gradient step into the objective function

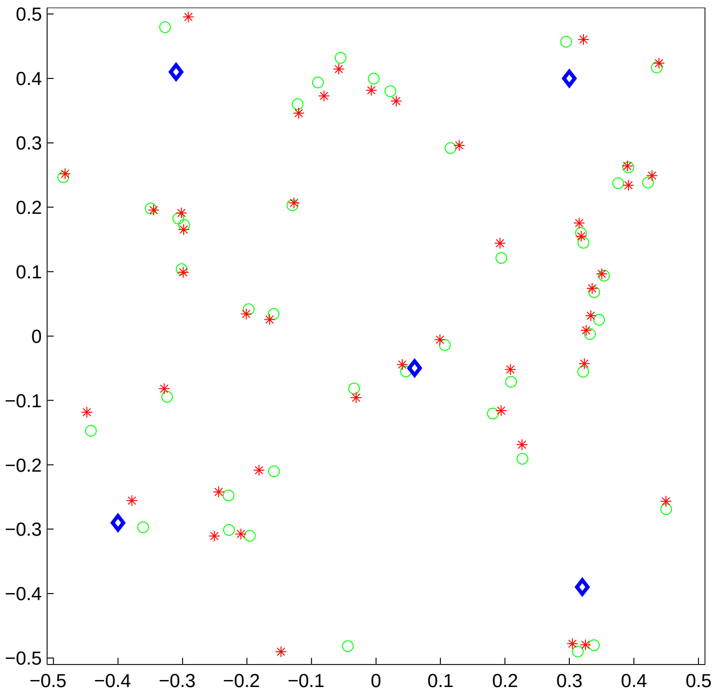

Figure 4: Localization after 50 gradient search steps.

to see if the sum of error squares is reduced. In Figure 5 the objective function values vs number of gradient steps curve are plotted in a blue line. One can see that in the first 20 steps the objective function value drops rapidly, and then the trajectory tends to be flat. This demonstrated that the gradient search method does improve the overall localization result. A natural question is how good the new localization is. To answer this question we need a lower bound of the objective function value. One trivial lower bound of the objective function value is 0; but a better one is the SDP relaxation objective value, since the SDP problem is a relaxation of the original 2-dimensional localization problem. In this case, the SDP objective value is about 0.094, plotted in Figure 5 in a red dashed line, and the gradient search method finds a 2-dimensional localization with objective function value about 0.12. Thus an error gap 0.026 of the suboptimality is obtained, which is less than 30% of the error lower bound value.

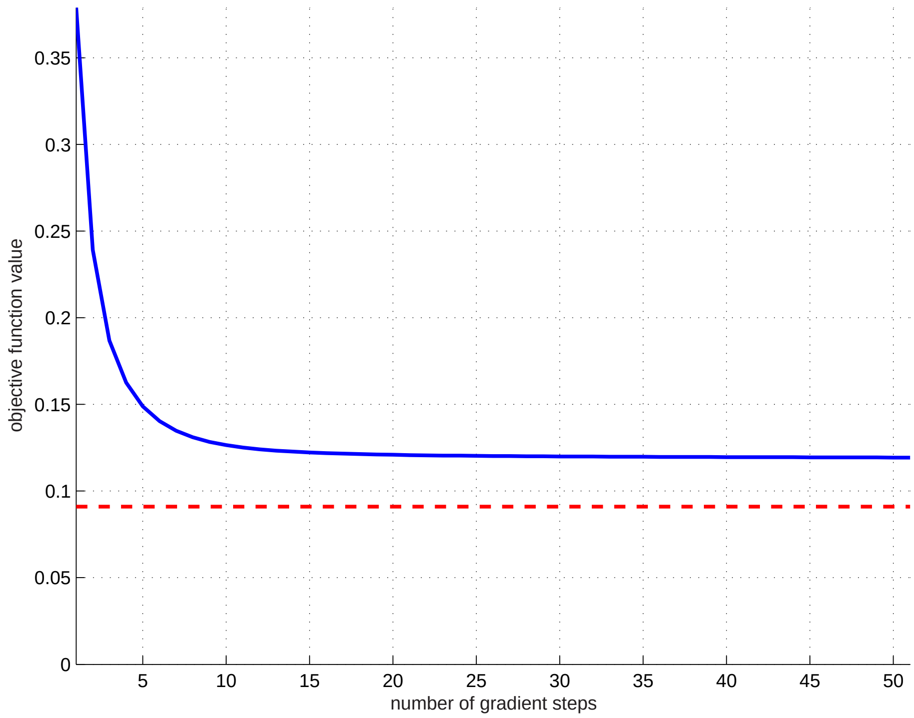

Figure 5: The sum of error squares vs number of gradient search steps.

The gradient-based descent method is a local search method and can be proved to find the global optimal solution only for convex optimization. The ad hoc sensor network localization problem is NOT a convex optimization problem. Hence a pure gradient search method should not work. To see this, another experiment is performed. We use the origin as the initial sensor locations and update them by the same rule (13). The updated trajectories are shown in Figure 6. Most of these sensors do not converge to their actual positions. Comparing it with the result of Figure 4, we can see that the use of the SDP solution as the initial localization makes all the difference.

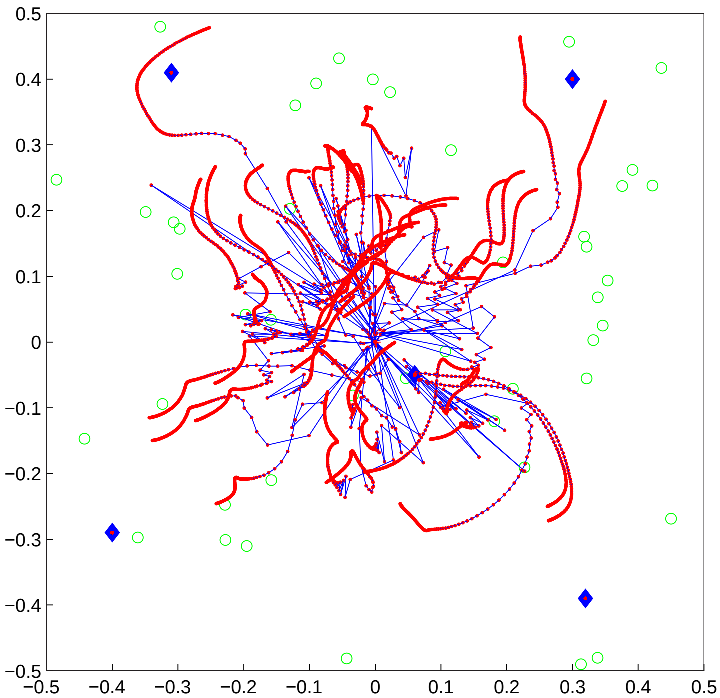

Figure 6: Gradient search trajectories using origin as the initial localization.

3.2 The Gradient Search Method with Exact Line-Search

The example in the previous section shows the basic idea of how gradient search method helps in improving the SDP solution. Several things have to be mentioned. First, because the distance measurements have noises, the true location should not be expected to find. Hence, that some of the trajectories do not converge to their corresponding true locations does not mean that the gradient method is not working. Actually, from the sum of error square function value, we know that the updates work. Secondly, it may not always reduce the objective function value if every sensor moves toward its gradient direction with a fixed step size α.

To furnish the analysis, we will now give an exact line-search method to choose the step size α and prove the convergence based on the WMLE objective function in (9). One reason we use WMLE instead of MLE in the proof is: when performing line search along the gradient direction, MLE does not have an analytical solution while WMLE does. This becomes clear when the step size selection is derived next.

Let us begin with (9). Following the procedure we did last time, we rewrite the WMLE objective function as,

$$\begin{aligned} f(X) := \textstyle\sum_{j,k;(j,k)\in N_a} \left(d_{jk}^2 - (x_j - a_k)^T(x_j - a_k)\right)^2 \\ + \textstyle\sum_{j,i;(j,i)\in N_x} \left(d_{ji}^2 - (x_j - x_i)^T(x_j - x_i)\right)^2 \end{aligned} \tag{14}$$

Define ∂f as

$$\partial f = [\partial {f_{x_1}}^T, \partial {f_{x_2}}^T, \ldots, \partial {f_{x_n}}^T]^T.$$

We have known $\partial f_{x_j}^T$ is a decent direction of the objective function (14). Let

$$X' = X - \alpha \partial f^T \tag{15}$$

Substitute X' into (14), we get

$$\begin{aligned} f(X') = \textstyle\sum_{j,k;(j,k)\in N_a} \left(d_{jk}^2 - (x_j + \alpha\partial f_{x_j}^T - a_k)^T(x_j + \alpha\partial f_{x_j}^T - a_k)\right)^2 \\ + \textstyle\sum_{j,i;(j,i)\in N_x} \left(d_{ji}^2 - (x_j + \alpha\partial f_{x_j}^T - x_i)^T(x_j + \alpha\partial f_{x_j}^T - x_i)\right)^2 \\ = \textstyle\sum_{j,k;(j,k)\in N_a} \left((d_{jk}^2 - \bar{d}_{jk}^2) - 2\alpha\partial f_{x_j}(x_j - a_k) - \alpha^2 \partial f_{x_j}\partial f_{x_j}^T\right)^2 \\ + \textstyle\sum_{j,i;(j,i)\in N_x} \left((d_{ji}^2 - \bar{d}_{ji}^2) - 2\alpha\partial f_{x_j}(x_j - x_i) - \alpha^2 \partial f_{x_j}\partial f_{x_j}^T\right)^2 \end{aligned} \tag{16}$$

where $\bar{d}_{jk}^2 = (x_j - a_k)^T(x_j - a_k)$ and $\bar{d}_{ji}^2 = (x_j - x_i)^T(x_j - x_i)$. Equation (16) is a forth order polynomial of α.

We would like to perform an exact line-search to decide the step size α to minimize $f(X')$ along the ∂f^T direction. To do this, differentiate (16) with respect to α and set it to be zero, we get,

$$\begin{aligned} 0 &= \frac{\partial f(X')}{\partial \alpha} \\ &= 4{c_1}^2\alpha^3 + 6c_1c_2\alpha^2 + 2(c_2^2 + 2c_1c_3)\alpha + 2c_2c_3 \end{aligned} \tag{17}$$

where

$$\begin{aligned}
c_1 &= -\sum_j \partial f_{x_j} \partial f_{x_j}^T \\
c_2 &= -2 \sum_{j,k;(j,k)\in N_a} \partial f_{x_j}(x_j - a_k) - 2 \sum_{j,i;(j,i)\in N_x} \partial f_{x_j}(x_j - x_i) \\
c_3 &= \sum_{j,k;(j,k)\in N_a} (d_{jk}^2 - \bar{d}_{jk}^2) + \sum_{j,i;(j,i)\in N_x} (d_{ji}^2 - \bar{d}_{ji}^2)
\end{aligned}$$

The right hand side of (17) is a third order polynomial of α and the analytical solution of its three roots exists. The positive real one closest to zero is the step size α that we use. It is easy to see using this step size always decreases objective function value, i.e. $f(X') < f(X)$.

It is clear now why we use the WMLE relaxation instead of MLE one to derive the step size rule: using MLE we need to solve a rational polynomial equation to find the step size that reduce the objective function value, and in general no analytical solution exists. One disadvantage of the exact step size is that the global network information is required. The cost of computing exactly best α could be too high when solving extremely large scale problems.

4 Numerical Examples

In this section, several examples are used to demonstrate the potential of the combination of the SDP relaxation and the gradient search method in solving the localization problem for ad hoc sensor networks.

We have shown in an experiment that the gradient search method can dramatically improve the localization accuracy when the distance measurements are noisy. It would be interesting to know how this method helps when the distance measurements are exact but the network is not uniquely localizable. We select a simple example from [24] to demonstrate this point. We will also use this example to compare the difference between the exact line-search and the constant step size gradient method.

Example 2 *Three anchors and two sensors are located at* $a_1 = [-\sqrt{3}/2, -1/2]^T$, $a_2 = [-\sqrt{3}/2, 1/2]^T$, $a_3 = [0,1]^T$, $x_1 = [0,0]^T$, *and* $x_2 = [0.77, 0.2]^T$, *respectively. The connection of sensor/sensor and sensor/anchor are shown in Figure 7 by solid lines. Exact distance measurements of the connections are available. This example is interesting because it is not uniquely localizable* *but has a unique localization in the original 2-dimensional space. We use both MLE SDP relaxation with fixed step size gradient method and WMLE SDP relaxation with exact line-search gradient method to solve it. The localization and gradient update trajectories are shown in Figure 7.*

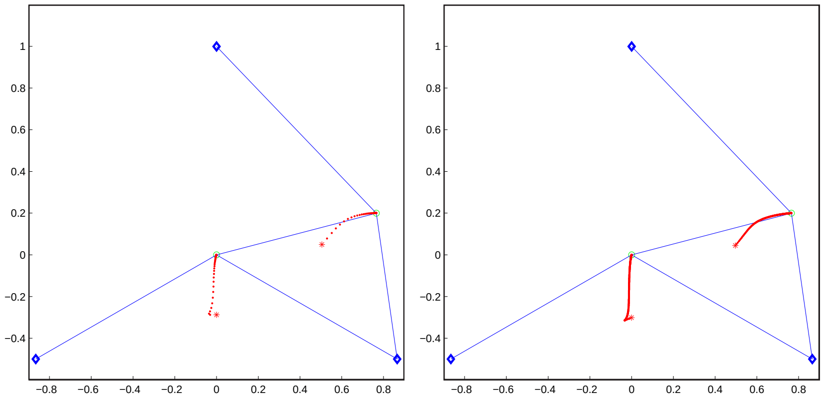

Figure 7: Left: WMLE SDP localization and update trajectory with exact line-search. Right: MLE SDP localization and update trajectory with fixed step size.

Both WMLE and MLE SDP relaxations in this example give wrong solutions. However, after applying gradient search methods, they both converge to the correct ones. This indicates that our approach can apply to at least some of the distance geometry problems which are not uniquely localizable.

The gradient update stops when the objective values between two steps is less than $1e-5$*. It takes* 61 *search steps for the gradient method with exact line-search and* 410 *search steps for the gradient method with a fixed step size* $\alpha = 0.05$ *to converge in Figure 7 left and right, respectively. One can observe that the exact line-search can significantly reduce the number of update steps. Nevertheless, if we change the step size to be* $\alpha = 0.5$*, it needs only* 51 *search steps to converge. This says a better chosen constant step size may accelerate the convergence rate as well, though no general rules can tell how to choose it.*

We have proved that with the exact line-search, the gradient method can reduce the objective function value and find a suboptimal solution. In section 3 we also demonstrate that a lower bound of the objective function value can be obtained by the SDP relaxation. The gap between these two values indicates how good the suboptimal solution is. In the following we would like to illustrate how this gap changes as the noise level varies.

Example 3 *An ad hoc sensor network with the same five anchors of the inspiring example and 45 sensors being randomly deployed each time in* $[-0.5, 0.5] \times [-0.5, 0.5] \in \Re^2$. *Radio range is set to be 0.35. This problem is repeatedly solved 100 times with noisy factors randomly chosen from* $[0, 0.3]$. *The solver we use is MLE SDP relaxation combing with gradient search method. In each solution, The SDP objective function value, i.e. the lower bound, and the best objective function value found by the gradient search method are recorded. Figure 8 shows the result.*

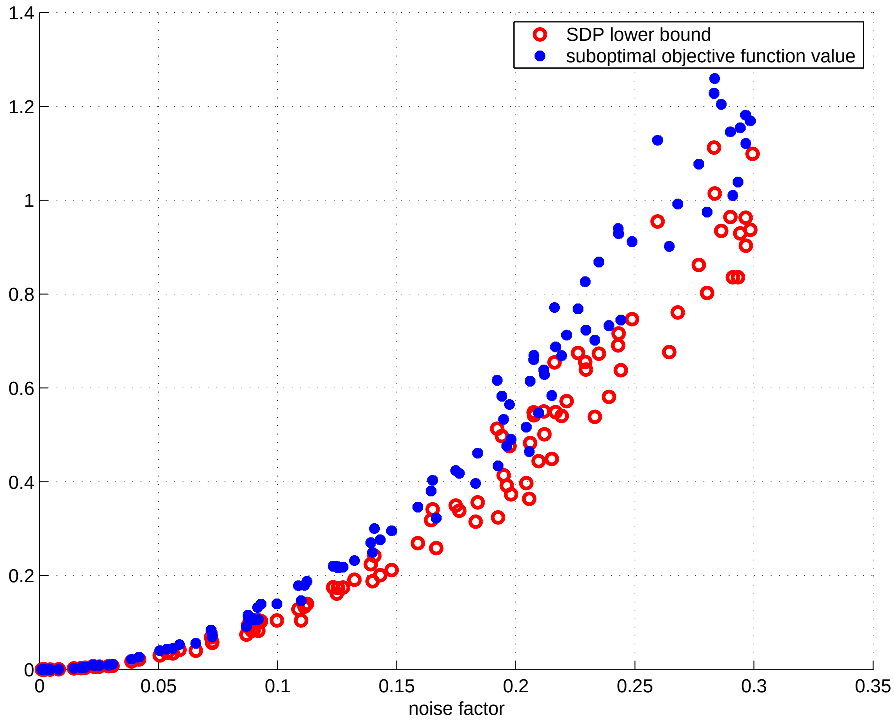

Figure 8: SDP lower bound and suboptimal objective function value vs noisy factor

In this figure we see that when the noisy factor is larger, both the lower bound and the suboptimal objective function value are growing but keeping close—the gap is roughly 30% of the lower bound value at all different noisy levels. This implies that not only the lower bound is a tight one but also the suboptimal solution is very close to optimal.

In [6] a distributed strategy to solve large size sensor network localization problem is suggested. It has been shown that when the distance measurements are accurate, this strategy works as well as the original one. However, in the case with measurement noises, this method needs more iterations and much longer time to converge and the result is not quite acceptable. We would like to re-examine the same example, and show that the SDP solution combining with the gradient search method will give us a much better localization within a much shorter time.

We would like to first describe the basic idea of the decomposition strategy. The detail can be found in [6]. This strategy decomposes the whole sensor network into several smaller domains based on the anchor distribution. Then by their connections to sensors each small domain forms a small localization problem. In one iteration, these small problems can be solved in parallel or sequentially. After solving them, the newly solved sensor locations are sent and received among the boundaries of domains. Newly localized sensors serve as anchors in the next iteration. Then after several iterations more and more sensors become anchors and finally the whole sensor network is localized. The difficulty is for the problem with measurement uncertainty, unless using all the connection information, we cannot tell a sensor is localized unbiasedly. Therefore propagating the localized sensors also propagates the errors. This algorithm spends lots of time in finding which sensors are localized correctly and which are not. For those that are judged incorrectly they need to be solved again with more connection information in the next iteration.

In our approach, we stop the SDP iteration earlier. Then, the solution produced by the decomposed SDP strategy is served as the starting solution for the gradient-based search method to minimize the objective function of the entire sensor network. Since the computation complexity of the gradient vector is very low so that the gradient-based method, although applied to the entire network without decomposition, can be completed extremely fast.

Example 4 *Consider the same problem in [6] with 1800 sensors, 200 anchors, radio range 0.05 and a larger noisy factor 0.1, where the sensor network is decomposed into 36 equal-sized domains. Figure 9 shows the SDP solution after 3 iterations.*

One can see that the SDP algorithm fails to find accurate solution in most small areas. But after 50 gradient-based search steps, the final localization, shown in Figure 10, is greatly improved.

Our program is implemented with Matlab and it uses SEDUMI [25] or DSDP2 [3] as the SDP solver. It costs 165 seconds of SEDUMI or 63 seconds of DSDP2 to get the SDP localization in Figure 9 on a PC. Then, after 50 gradient search steps, the objective function is reduced from 12.81 to 0.230.

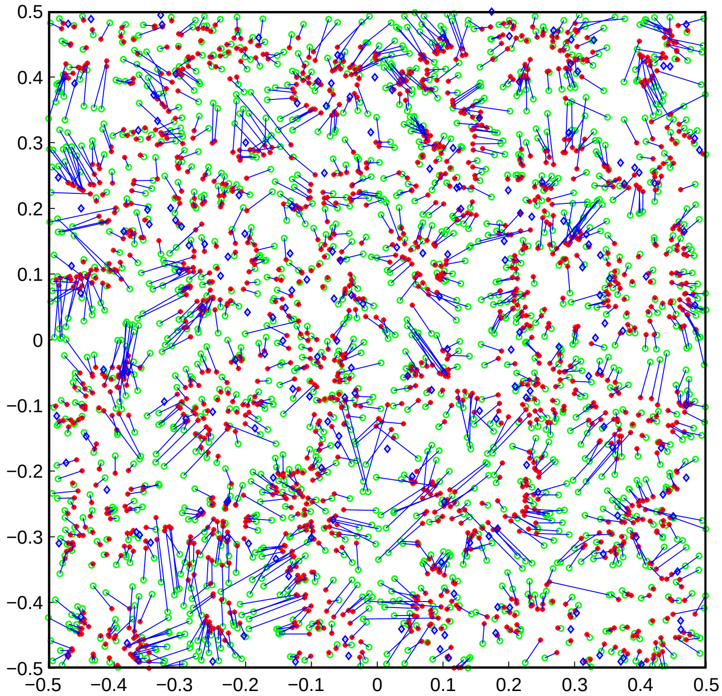

Figure 9: SDP localization

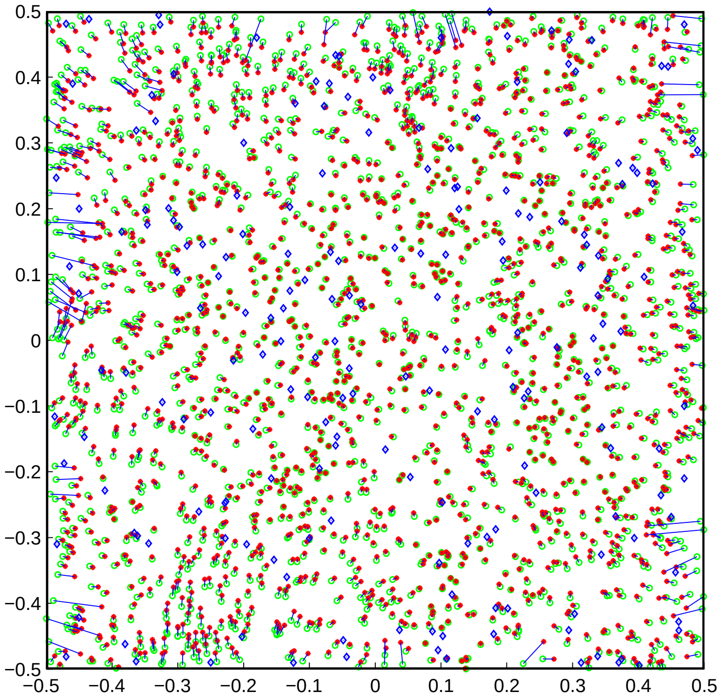

Figure 10: Localization after 50 gradient search steps

It can be seen from Figure 10 that most of the sensors are located very close to their true positions, although few of them, most of which are close to the boundary of the network, are solved inaccurately. We see, from this example, the dramatic solution improvement and the cost efficiency of the combination of the SDP relaxation and the gradient search methods.

5 Discussion and Conclusion

In this report we have developed a new SDP rounding technique, based on the gradient-based search method, to find a suboptimal solution for the ad hoc sensor network localization problem. The technique utilizes the projection of the SDP relaxation solution as the initial localization and improves it iteratively by using merely (local) gradient information. The result is quite satisfactory even when the distance measurement is highly noise. The gradient-based search method has extremely low computational complexity and can be done independently for each sensor, that is, its computation is totally distributable and scalable. Combining with the distributed SDP algorithm suggested in [6], we have found an extremely effective and efficient localization algorithm.

Since in all cases we observe that the gradient-based search have reduced the objective function value close to its SDP lower bound, one would expect that, for some of the localization problems, the combined algorithm has actually found the global optimal solution. Further research has to be done to explain this observation and to develop even more powerful methods.

We would like to thank Eunji Lim for sharing her code on the exact line search with us.

References

[1] A. Y. Alfakih, A. Khandani, and H. Wolkowicz. Solving Euclidean distance matrix completion problems via semidefinite programming. Comput. Optim. Appl., 12(1-3):13–30, November 1999.

[2] A. I. Barvinok. Problems of Distance Geometry and Convex Properties of Quadratic Maps. Discrete and Computational Geometry, 13:189–202, 1995.

[3] S. J. Benson, Y. Ye, and X. Zhang. Solving large-scale sparse semidefinite programs for combinatorial optimization. *SIAM Journal on Optimization*, 10(2):443–461, 2000.

[4] J. Beutel. Geolocation in a PicoRadio Environment, MS Thesis, ETH Zurich, December, 1999.

[5] P. Biswas and Y. Ye. Semidefinite Programming for Ad Hoc Wireless Sensor Network Localization. Technical report, Dept of Management Science and Engineering, Stanford University, September 2003.

[6] P. Biswas and Y. Ye. A distributed method for solving semidefinite programming arising from ad hoc wireless sensor network localization. Technical report, Dept of Management Science and Engineering, Stanford University, Octiber 2003.

[7] S. Boyd, L. E. Ghaoui, E. Feron, and V. Balakrishnan. Linear Matrix Inequalities in System and Control Theory. SIAM, 1994.

[8] N. Bulusu, J. Heidemann, and D. Estrin. GPS-less Low-cost Outdoor Localization for Very Small Devices, IEEE Personal Communications, 7(5):28V34, Oct 2000.

[9] J. Caffery and G. Stuber. Subscriber Location in CDMA Cellular Networks, IEEE Transactions on Vehicular Technology, Vol. 47 No.2, pages 406-416, May 1998

[10] L. Doherty, L. E. Ghaoui, and S. J. Pister. Convex position estimation in wireless sensor networks. In IEEE Infocom, Vol. 3, pages 1655 –1663, April 2001.

[11] T. Eren, D. K. Goldenberg, W. Whiteley, Y. R. Yang, A. S. Moore, B. D. O. Anderson, P. N. Belhumeur. Rigidity, Computation, and Randomization in Network Localization. IEEE INFOCOM, 2004.

[12] FCC. Revision of the commissions rules to ensure compatibility with enhanced 911 emergency calling systems, RM-8143, CC Docket 94-102, FCC, Washington, DC, July 26, 1996.

[13] J. Hightower and G. Borriello. Location systems for ubiquitous computing. IEEE Computer, 34(8):57–66, August 2001.

[14] A. Howard, M. Mataric, and G. Sukhatme. Relaxation on a mesh: a formalism for generalized localization. In IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems, Vol. 3, pages 1055–1060, October 2001.

[15] B. Jackson, T. Jordán. Connected Rigidity Matroids and Unique Realizations of Graphs. Preprint, 2003.

[16] E. Kaplan. Understanding GPS Principles and Applications Artech House, 1996

[17] M. Laurent. Matrix completion problems. The Encyclopedia of Optimization., 3:221–229, 2001.

[18] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler and J. Anderson. Wireless Sensor Networks for Habitat Monitoring, WSNA02, September 28, 2002, Atlanta, Georgia, USA.

[19] J. Moré and Z. Wu. Global continuation for distance geometry problems. SIAM Journal on Optimization, 7:814–836, 1997.

[20] D. Niculescu and B. Nath. Ad hoc positioning system (APS). In IEEE GLOBECOM (1), pages 2926–2931, 2001.

[21] C. Savarese, J. Rabay, and K. Langendoen. Robust positioning algorithms for distributed ad-hoc wireless sensor networks. USENIX Technical Annual Conference, June 2002.

[22] A. Savvides, C.-C. Han, and M. B. Srivastava. Dynamic fine-grained localization in ad-hoc networks of sensors. Mobile Computing and Networking, pages 166-179, 2001.

[23] Y. Shang, W. Ruml, Y. Zhang, and M. P. J. Fromherz. Localization from mere connectivity. In Proceedings of the 4th ACM international symposium on Mobile Ad Hoc Networking & Computing, pages 201–212. ACM Press, 2003.

[24] Anthony Man-Cho So and Yinyu Ye. Theory of Semidefinite Programming for Sensor Network Localization. Technical report, Dept of Management Science and Engineering, Stanford University, April 2004.

[25] J. F. Sturm. Let sedumi seduce you, October 2001.

[26] T.-C. Wang and S. Lall. Decentralized Sensor Localization via Convex Optimization,Dept of Aeronautics and Astronautics, Stanford University, March 2004.