

Convex Euclidean distance embedding for collaborative position localization with NLOS mitigation

Chao Ding and Hou-Duo Qi

the date of receipt and acceptance should be inserted later

Abstract One of the challenging problems in collaborative position localization arises when the distance measurements contain Non-Line-Of-Sight (NLOS) biases. Convex optimization has played a major role in modelling such problems and numerical algorithm developments. One of the successful examples is the Semi-Definite Programming (SDP), which translates Euclidean distances into the constraints of positive semidefinite matrices, leading to a large number of constraints in the case of NLOS biases. In this paper, we propose a new convex optimization model that is built upon the concept of Euclidean Distance Matrix (EDM). The resulting EDM optimization has an advantage that its Lagrangian dual problem is well structured and hence is conducive to algorithm developments. We apply a recently proposed 3-block alternating direction method of multipliers to the dual problem and tested the algorithm on some real as well as simulated data of large scale. In particular, the EDM model significantly outperforms the [existing](#) SDP model and several others.

Keywords Euclidean distance matrix · Collaborative localization · Non-Line of Sight (NLOS) · Augmented Lagrangian · Alternating direction method of multipliers (ADMM)

1 Introduction

One of the challenging problems in source localization arises from the situation where some of the distance measurements are the type of Non-Line-of-Sight (NLOS). The challenges come from two fronts. One is that different localization models appear to be required for different scenarios depending whether prior information is available of the NLOS measurements (e.g., percentage of NLOS measurements and/or the corresponding source nodes). We refer to the excellent paper by Chen et.al. [8] for diverse models in handling different scenarios. The second challenging front is on developing efficient algorithms especially when the number of unknown sources grows bigger. This is the algorithmic scale issue for many known algorithms. The purpose of this paper is to develop a new convex optimization model using the Euclidean Distance Matrix (EDM) to deal with a general

This work is supported in part by Engineering and Physical Science Research Council (UK) project EP/K007645/1.

C. Ding
Institute of Applied Mathematics, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, P.R. China.
E-mail: dingchao@amss.ac.cn
H.D. Qi
School of Mathematics, The University of Southampton, Highfield, Southampton SO17 1BJ, UK.
E-mail: hdqi@soton.ac.uk

scenario where no prior information is assumed of NLOS measurements. We further develop an efficient algorithm that is capable of handling thousands of unknown sources and its localization accuracy on real and simulated data seemed to outperform all the important algorithms that we have tested in the numerical part.

Needless to say that there exist a vast number of papers on source localization problems of various kinds. We do not intend to conduct a survey on them, but rather focus on those that are most relevant to this research and are of range-based localizations. We refer to the (survey) papers by Patwari et. al. [28], Stoica and Li [37], and Guvenc and Chong [18] (and references therein) for detailed documentation on those problems. We are only interested in the situations where there are usually two types of nodes: the anchors whose positions are known and the sensors (or sources) whose positions are to be estimated based on range-based measurements by, such as received signal strength, time-of-arrival or time-difference-of-arrival. Our framework is of the type of the collaborative position localization (also known as cooperative position localization), which determines the sensor positions collaboratively, using both sensors-to-anchors and sensors-to-sensors measurements.

Distance measurements can largely be classified into two types. One is the type of **Line-of-Sight (LOS)**, which yields the Euclidean distances or small perturbation of them among the nodes. The second type is of the NLOS, which refers to the case where the LOS path is blocked due to environment limitations such as the indoor environment depicted in the example of locating Motorola facilities in [27]. NLOS propagation often leads to range-based measurements much larger than the true values. For localizations with all measurements being of LOS, the convex relaxation method of Semi-Definite Programming (SDP) works very well and is probably the most popular method, see Biswas and Ye [5] and some of its follow-up papers [4, 47, 26, 29, 15]. The second-order cone programming by Tseng [41] and the facial reduction SDP approach by Krislock and Wolkowicz [24] can be regarded as variants of the SDP approach, which is capable of handling thousands of sensors. However, without proper modifications, NLOS propagation can significantly degrade the accuracy of the localizations by those SDP methods.

Nevertheless, SDP continues to be a major model for the case of NLOS, but with growing complexity in terms of the constraints involved. For example, for the model that aims for a large number of localizations in [8, (25)], there have as many as $|\mathcal{E}|$ of 2×2 positive semidefinite constraints, $|\mathcal{E}|$ of 4×4 positive semidefinite constraints, and $|\mathcal{E}|$ of linear constraints. Here $|\mathcal{E}|$ denotes the number of observed distances. Even for the case that there is just one unknown source, the SDP model in [43], which aims to simultaneously locate one single source and the NLOS biases, has $(m + 1)$ of 3×3 positive semidefinite constraints and m linear constraints, where m is the number of anchors and it also equals the number of observed distances. When $|\mathcal{E}|$ grows beyond a few hundreds, the complexity of those models would significantly slow down the state-of-art SDP solvers.

As expected, prior information about NLOS distances does help to propose different approaches. In the literature, one approach to mitigate the NLOS propagation is to develop some methods to distinguish NLOS links from LOS links. For example, for the single-source network problem, various identification techniques such as the maximum-likelihood based estimation are proposed in [44, 33]. After identifying NLOS connections, different optimization techniques can be employed to improve the localization performance. For the single-source network, the linear programming method and the weighted linear least squares method are proposed in [45] and [17], respectively (see the nice survey [18] for the overview of various NLOS identification techniques and optimization methods). For the collaborative localization, the convex SDP relaxation model and the distributed iterative parallel projection method are proposed in [42] and [22], respectively.

However, for most applications, it is very difficult to distinguish NLOS links from LOS ones. As pointed out by [18] that misidentification may significantly degrade the localization accuracy of optimization methods. Even worse is that there is often scarce or no information at all about the statistics of the NLOS errors to be used for identification. Consequently, the NLOS mitigation which does not require a priori the NLOS connection status or NLOS error information is necessary

in practical applications. The SDP models discussed above in [8,43] do not require any prior information about NLOS. Some studies such as the recent one by Yousefi et. al. [49] show that without NLOS information, the estimator based on the Huber function [21] is more robust against NLOS errors. However, due to the non-convexity of [the resulting optimization problem](#), a good initial point is essential to achieve a good estimation. In order to produce a faithful initial point, different convex relaxations of the objective function are studied in [1,49]. Although their approach can provide a good initial estimator if the number of NLOS links is small, it is not necessarily the case when the number of NLOS links is large [49].

When the observed distances are all of LOS with/without noises, the classical Multi-Dimensional Scaling (cMDS) and its sometime sophisticated variants have been widely used in source localization as thoroughly demonstrated in the books [9,6]. Here, we only focus on a group of research that are deeply related to cMDS, but are very different in its nature. The central concept in cMDS is the squared EDM. Suppose D is a distance matrix, the cMDS and its associated optimization problems all work with the doubly centralized matrix JDJ , where J is the centralizing matrix. The group of research we are interested in work with the matrix D directly. This results in a class of EDM optimization problems that require iterative methods for them. The key difference between cMDS and EDM optimization as well as their respective benefits have been well documented in Chapter 7 of Dattorro [10] (also see Section 2 (c)).

Interestingly, EDM optimization has long been studied in numerical linear algebra and numerical optimization and further applied to molecular conformation, see [19,12,13,14,30,31]. What makes EDM optimization more interesting is that such problems can be reformulated as SDP as done in [2,4,39]. More mathematical properties of EDM optimization are being revealed through various formulations. For example, for a single source localization problem under LOS, the EDM localization in [32] is equivalent to that obtained by a trust-region method studied in [3].

Despite its recent advances, the application of EDM optimization to the collaborative position localization with NLOS propagation remains, to our knowledge, an open issue. This paper tries to initiate such a study by proposing a convex EDM embedding model for NLOS mitigation. The main contributions of the paper are summarized as follows.

- (a) We develop an EDM model that is a convex conic matrix optimization with the cone being the almost positive semidefinite cone (an interesting geometric subject itself). The model can be regarded as a convex relaxation of the least-trimmed squares (LTS) estimation [35] when applied to our EDM embedding. One of the several relaxation techniques used in this paper is to replace the non-convex ℓ_0 -norm regularization term by its closest convex approximation, the ℓ_1 -norm. This technique is same as the one that is used in Forero and Giannakis [11] to deal with very few of large biases in distance measurements, which were treated as outliers. But our model is convex and their robust MDS model is nonconvex.
- (b) The EDM optimization has such a nice structure that its dual problem has 3 separable block-variables, paving the way for developing an efficient algorithm of Alternating Direction Method of Multipliers (ADMM). The algorithm is derived based on the recent results presented in [38] for large scale SDP problems.
- (c) The numerical comparison on real as well as simulated data demonstrated that our model and algorithm significantly outperform (both in terms of cpu time and solution quality) the best methods available including the SDP method in [8] for NLOS mitigation. Moreover, our algorithm is capable of handling thousands of unknown sources with millions of observed distances. Hence, our model and algorithm are suitable for large network localization where the existing methods have difficulties.

The rest of the paper is organized as follows. Section 2 provides necessary background on Euclidean distance embedding. The proposed convex EDM model is described in Section 3 with more detailed interpretation. Section 4 contains a convergent 3-block ADMM algorithm for our proposed convex EDM model. The performance of the proposed model is compared with some of

the best algorithms in the literature through simulations and real data analysis, and is reported in Section 5. We conclude the paper in Section 6.

2 Background on EDM and notation

In this section, we draw some basic facts from cMDS and emphasize how they will be used in our EDM optimization. There are three elements that have become basics in Euclidean distance embedding. The first one is the definition of the squared Euclidean distance matrix (EDM). The second includes various characterizations of EDMs. And the third one is the Procrustes analysis that produces the actual embedding in a Euclidean space. We briefly describe them one by one. Standard references are [9, 6, 10].

(a) Squared EDM. Let \mathcal{S}^n denote the space of $n \times n$ symmetric matrices equipped with the standard inner product $\langle A, B \rangle = \text{trace}(AB)$ for $A, B \in \mathcal{S}^n$. Let $\|\cdot\|$ denote the induced Frobenius norm. Let \mathcal{S}_+^n denote the cone of positive semidefinite matrices in \mathcal{S}^n (often abbreviated as $X \succeq 0$ for $X \in \mathcal{S}_+^n$). The so-called *hollow* subspace \mathcal{S}_h^n is defined by (“:=” means define)

$$\mathcal{S}_h^n := \{A \in \mathcal{S}^n : \text{diag}(A) = 0\},$$

where $\text{diag}(A)$ is the vector formed by the diagonal elements of A . A matrix D is a (squared) EDM if $D \in \mathcal{S}_h^n$ and there exist points $\mathbf{p}_1, \dots, \mathbf{p}_n$ in \mathbb{R}^r such that $D_{ij} = \|\mathbf{p}_i - \mathbf{p}_j\|^2$ for $i, j = 1, \dots, n$. \mathbb{R}^r is often referred to as the embedding space and r is the embedding dimension when it is the smallest such r . All vectors are treated as column vectors. \mathbf{x}^T is the transpose of the vector \mathbf{x} , hence it is a row vector. We note that D must belong to \mathcal{S}_h^n if it is an EDM.

(b) Characterizations of EDM. It is well-known that a matrix $D \in \mathcal{S}^n$ is an EDM if and only if

$$D \in \mathcal{S}_h^n \quad \text{and} \quad J(-D)J \succeq 0, \quad \text{with } J := I - \mathbf{e}\mathbf{e}^T/n, \quad (1)$$

where I is the identity matrix in \mathcal{S}^n and \mathbf{e} is the vector of all ones in \mathbb{R}^n . The origin of this result can be traced back to Schoenberg [36] and an independent work [48] by Young and Householder.

It is noted that the matrix J , when treated as an operator, is the orthogonal projection onto the subspace $\mathbf{e}^\perp := \{x \in \mathbb{R}^n : \mathbf{e}^T x = 0\}$. The characterization (1) simply means that D is an EDM if and only if $D \in \mathcal{S}_h^n$ and D is negative semidefinite on the subspace \mathbf{e}^\perp :

$$-D \in \mathcal{K}_+^n := \left\{A \in \mathcal{S}^n : x^T A x \geq 0, \quad x \in \mathbf{e}^\perp\right\}. \quad (2)$$

\mathcal{K}_+^n is known as the almost positive semidefinite cone. The orthogonal projection onto \mathcal{K}_+^n is given as follows:

$$\Pi_{\mathcal{K}_+^n}(A) = A + \Pi_{\mathcal{S}_+^n}(-JAJ), \quad \forall A \in \mathcal{S}^n, \quad (3)$$

where for a closed convex set $C \in \mathbb{R}^n$, $\Pi_C(\mathbf{x})$ is the orthogonal projection of $\mathbf{x} \in \mathbb{R}^n$ onto C . The polar cone of \mathcal{K}_+^n is defined by

$$\left(\mathcal{K}_+^n\right)^\circ := \left\{X \in \mathcal{S}^n \mid \langle X, A \rangle \leq 0, \quad \forall A \in \mathcal{K}_+^n\right\}.$$

(c) Euclidean embedding. If D is an EDM with embedding dimension r , then $-JDJ \succeq 0$. Let

$$-JDJ/2 = PP^T \quad (4)$$

where $P \in \mathbb{R}^{n \times r}$. Let \mathbf{p}_i denote the i th column of P^T . It is known [9] that $\{\mathbf{p}_1, \dots, \mathbf{p}_n\}$ are the embedding points of D in \mathbb{R}^r , i.e., $D_{ij} = \|\mathbf{p}_i - \mathbf{p}_j\|^2$. We also note that any rotation and shifting of $\{\mathbf{p}_1, \dots, \mathbf{p}_n\}$ would give the same D . In other words, there are infinitely many sets of embedding

points. To find a desired set of embedding points that match positions of the existing anchors, one needs to conduct the Procrustes analysis, which is a simple computational scheme, see [9, Chp. 5]. We omit the details.

cMDS makes use of the fact (4) in the following way. Given a pre-distance matrix \tilde{D} , find an EDM D such that it is the solution of

$$\min \frac{1}{2} \|J(D - \tilde{D})J\|^2, \quad \text{s.t.} \quad -JDJ \succeq 0. \quad (5)$$

In other words, the solution yields the closest distance between JDJ and $J\tilde{D}J$ (see [9, Sect. 2.2.3]).

In contrast, the EDM optimization aims to measure the distance directly between D and \tilde{D} and find the best solution through the following optimization:

$$\min \frac{1}{2} \|D - \tilde{D}\|^2, \quad \text{s.t.} \quad D \text{ is EDM.} \quad (6)$$

The problem does not have a closed-form solution any more and iterative numerical methods are often required. The reward is that one gets a closer (with respect to the Frobenius norm of matrices) distance matrix to \tilde{D} than that obtained by cMDS. It is guaranteed by the optimality of problem (6). This paper will propose one of such EDM optimization problems for NLOS mitigation, which is explained in the next section.

Additional notation: We let \mathbf{e}_i denote the vector whose i th element is 1 and zero elsewhere. For a vector \mathbf{x} , the ℓ_0 norm $\|\mathbf{x}\|_0$ is the number of nonzero elements in \mathbf{x} , whereas the ℓ_1 norm is $\|\mathbf{x}\|_1 = \sum |x_i|$. Its ℓ_2 norm is $\|\mathbf{x}\| = (\sum x_i^2)^{1/2}$. For a positive vector \mathbf{w} , the \mathbf{w} -weighted ℓ_2 norm $\|\mathbf{x}\|_{\mathbf{w}}$ is defined by $\|\mathbf{x}\|_{\mathbf{w}} := (\sum w_i x_i^2)^{1/2}$. For a close convex set $C \subset \mathbb{R}^n$, $\delta_C(\cdot)$ is its indicator function and $\delta_C^*(\cdot)$ is its conjugate function of $\delta_C(\cdot)$ defined by

$$\delta_C^*(\mathbf{y}) := \sup_{\mathbf{x} \in C} \{\langle \mathbf{x}, \mathbf{y} \rangle - \delta_C(\mathbf{x})\}.$$

In particular, when C is a closed convex cone, we have $\delta_C^*(\mathbf{y}) = \delta_{C^\circ}(\mathbf{y})$. For a linear operator $\mathcal{A}: \mathbb{R}^n \mapsto \mathbb{R}^m$, \mathcal{A}^* denotes its adjoint.

3 The problem of NLOS mitigation and EDM model

In this section, we first formally describe the problem of NLOS mitigation. When the least square criterion is applied to the problem, it naturally leads to an optimization problem, which we show is equivalent to the well-known Least-Trimmed Squares (LTS) [35]. Hence, the formulation is intractable even for problems of small scale. Finally, we propose a convex relaxation, which leads to our convex EDM optimization.

3.1 Problem description

It is convenient to state the problem as a network localization problem. Suppose we have a network in \mathbb{R}^r (e.g., $r = 2$ or 3), where there are n_s unknown-position sensors and n_a anchors, whose positions are known. Denote $\mathbf{p}_i \in \mathbb{R}^r$, $i = 1, \dots, n_s$, the coordinates of the unknown sensors and $\mathbf{p}_i \in \mathbb{R}^r$, $i = n_s + 1, \dots, n_s + n_a$, the coordinates of the given anchors, respectively. Denote $n := n_s + n_a$. Let d_{ij} be the Euclidean distance between the i -th sensor and the j -th sensor/anchor, i.e.,

$$d_{ij} := \|\mathbf{p}_i - \mathbf{p}_j\| \quad \forall i \in \{1, \dots, n_s\}, j \in \{1, \dots, n\}.$$

In this paper, we assume that if the distance d_{ij} is less than the physical communication range R , then the i -th and the j -th nodes are able to communicate with each other (i.e., they are connected)

and the (noisy) distance estimation \tilde{d}_{ij} can be obtained based on the signal of time-of-arrival (TOA) measurements. Let \mathcal{E} be the set of index pairs of the connected sensors-sensors/sensors-anchors. We further assume that the distance estimations are symmetric, i.e., $\tilde{d}_{ji} = \tilde{d}_{ij}$ for all $(i, j) \in \mathcal{E}$. The task is to estimate the sensor locations \mathbf{p}_i , $i = 1, \dots, n_s$ from the observation distance information.

We are mainly concerned with the case where some of the observed distances \tilde{d}_{ij} are contaminated with NLOS errors. Let \mathcal{E}_{NLOS} denote the index pairs of NLOS observations and $\mathcal{E}_{LOS} = \mathcal{E} \setminus \mathcal{E}_{NLOS}$ be the index pairs of LOS observations, respectively. Therefore, we may assume that the distance estimations take the form

$$\tilde{d}_{ij} = d_{ij} + \xi_{ij} + \eta_{ij}, \quad (i, j) \in \mathcal{E}, \quad (7)$$

where ξ_{ij} are i.i.d. noise errors which follow the normal distribution $\mathcal{N}(0, \sigma)$ and η is a vector indexed by (i, j) and is defined as follows:

$$\eta_{ij} := \begin{cases} 0 & \text{if } (i, j) \in \mathcal{E}_{LOS} \\ > 0 & \text{if } (i, j) \in \mathcal{E}_{NLOS}. \end{cases}$$

It is commonly assumed that each η_{ij} for $(i, j) \in \mathcal{E}_{NLOS}$ is significantly larger than the magnitude of the corresponding noise error ξ_{ij} .

We emphasize that, in most networks, it is difficult to distinguish NLOS connections from LOS ones. This implies that we do not know the index set \mathcal{E}_{NLOS} in advance. Thus, even all \tilde{d}_{ij} are available and there is no noise, i.e., $\xi_{ij} = 0$ for all $(i, j) \in \mathcal{E}$, solving for d_{ij} and η_{ij} from (7) is an under-determined problem. However, we believe that it is reasonable to assume that there are only a small number (or percentage) of the edges that are NLOS connections (i.e., the cardinality of \mathcal{E}_{NLOS} is small). For example, by employing a variety of communication techniques such as increasing the number of LOS paths, the percentage of NLOS connections can be decreased.

3.2 Least square formulation and LTS

(a) Regularized least square formulation. We now apply the least-square criterion on (7) to get an optimization problem. It follows from (7) that for each $(i, j) \in \mathcal{E}$, we have

$$\begin{aligned} d_{ij}^2 &= \left(\tilde{d}_{ij} - \xi_{ij} - \eta_{ij} \right)^2 = \tilde{d}_{ij}^2 + \eta_{ij}^2 + 2\eta_{ij}(\xi_{ij} - \tilde{d}_{ij}) - 2\tilde{d}_{ij}\xi_{ij} + \xi_{ij}^2 \\ &= \tilde{d}_{ij}^2 + \eta_{ij}^2 - 2\eta_{ij}(\eta_{ij} + d_{ij}) - 2\tilde{d}_{ij}\xi_{ij} + \xi_{ij}^2 \quad (\text{by (7)}) \\ &= \tilde{d}_{ij}^2 - y_{ij} - 2\tilde{d}_{ij}\xi_{ij} + \xi_{ij}^2, \end{aligned}$$

where the new quantity y_{ij} is defined by

$$y_{ij} := \eta_{ij}^2 + 2\eta_{ij}d_{ij} \quad \forall (i, j) \in \mathcal{E}.$$

Since magnitude of ξ_{ij}^2 is much smaller than the term involving ξ_{ij} , we obtain that

$$\frac{d_{ij}^2 + y_{ij} - \tilde{d}_{ij}^2}{2\tilde{d}_{ij}} \approx \xi_{ij}, \quad (i, j) \in \mathcal{E}. \quad (8)$$

Because ξ_{ij} is noise and follows the normal distribution, it is natural to apply the least square to (8) to get an optimization formulation. We would like to point out that the argument led to Equation (8) has been influenced by a similar argument in [43, Section IV] that led to [43, Eq. (10)], where a single source localization under NLOS links was mainly studied.

We also note that for all $(i, j) \in \mathcal{E}$, we have $\eta_{ij} \geq 0$. This implies $y_{ij} \geq 0$ for all $(i, j) \in \mathcal{E}$ and $y_{ij} > 0$ if and only if $\eta_{ij} > 0$. Since the cardinality of \mathcal{E}_{NLOS} is presumably small, we may assume

that the vector $y = (y_{ij})$ is sparse. It has been a well-known strategy that adding ℓ_0 norm of y encourages the sparse recovery of y . Therefore, the natural least-square formulation of (8) takes the following form with the ℓ_0 -norm regularization term:

$$\begin{aligned} \min_{\mathbf{p}_i, \eta_{ij}, y_{ij}} \quad & \sum_{(i,j) \in \mathcal{E}} \frac{1}{2\tilde{d}_{ij}^2} (d_{ij}^2 + y_{ij} - \tilde{d}_{ij}^2)^2 + \rho \|y\|_0 \\ \text{s.t.} \quad & y_{ij} = \eta_{ij}^2 + 2\eta_{ij}d_{ij}, \quad (i,j) \in \mathcal{E} \\ & \|\mathbf{p}_i - \mathbf{p}_j\| = d_{ij}, \quad (i,j) \in \mathcal{E}, \end{aligned} \quad (9)$$

where $\rho > 0$ is a given parameter. In the formulation, we included the embedding points \mathbf{p}_i , which define the actual distance d_{ij} .

(b) Equivalence to LTS. The following interpretation in terms of LTS will put off any attempt to solve the problem (9) directly.

We begin with a simple observation. For any given $y_{ij} \geq 0$ and $d_{ij} > 0$, we can easily find out the corresponding $\eta_{ij} \geq 0$ such that $y_{ij} = \eta_{ij}^2 + 2\eta_{ij}d_{ij}$, $(i,j) \in \mathcal{E}$. Therefore, for any given \mathbf{p}_i , $i = 1, \dots, n$, the optimization problem (9) can be rewritten as the following optimization problem with respect to y_{ij} only (since \mathbf{p}_i , $i = 1, \dots, n$ are assumed be fixed)

$$\begin{aligned} \min \quad & \sum_{(i,j) \in \mathcal{E}} \frac{1}{2\tilde{d}_{ij}^2} (d_{ij}^2 + y_{ij} - \tilde{d}_{ij}^2)^2 + \rho \|y\|_0 \\ \text{s.t.} \quad & y_{ij} \geq 0, \quad (i,j) \in \mathcal{E}. \end{aligned} \quad (10)$$

It is easy to see (cf. e.g., [25, Proposition 2.2]) that the global optimal solution y_{ij}^* of problem (10) has a close form given by

$$y_{ij}^* = \begin{cases} (\tilde{d}_{ij}^2 - d_{ij}^2)_+ & \text{if } ((\tilde{d}_{ij}^2 - d_{ij}^2)_+)^2 > 2\rho\tilde{d}_{ij}^2, \\ 0 & \text{if } ((\tilde{d}_{ij}^2 - d_{ij}^2)_+)^2 \leq 2\rho\tilde{d}_{ij}^2, \end{cases} \quad (11)$$

where $(\tilde{d}_{ij}^2 - d_{ij}^2)_+ := \max\{\tilde{d}_{ij}^2 - d_{ij}^2, 0\}$.

Let \mathbf{p}_i^* , $i = 1, \dots, n_s$ and y_{ij}^* , $(i,j) \in \mathcal{E}$ be any local optimal solution of (9). Denote $\tau^* := \|y^*\|_0$. Then, we know from (11) that \mathbf{p}_i^* , $i = 1, \dots, n_s$ are also the local optimal solutions [with zero optimal value](#) of the following LTS problem with the trimmed constant $t := |\mathcal{E}| - \tau^*$,

$$\min \sum_{l=1}^t r_l^\uparrow(\mathbf{p}_1, \dots, \mathbf{p}_{n_s}), \quad (12)$$

where $r_1^\uparrow(\mathbf{p}_1, \dots, \mathbf{p}_{n_s}) \leq \dots \leq r_{|\mathcal{E}|}^\uparrow(\mathbf{p}_1, \dots, \mathbf{p}_{n_s})$ are the ordered elements of the following set

$$\left\{ r_{ij}(\mathbf{p}_1, \dots, \mathbf{p}_{n_s}) = \frac{1}{2\tilde{d}_{ij}^2} \left((\tilde{d}_{ij}^2 - \|\mathbf{p}_i - \mathbf{p}_j\|^2)_+ \right)^2 \mid (i,j) \in \mathcal{E} \right\}.$$

To summarize, any local solution of (9) is just the LTS estimate of the corresponding unknown sensors. Similar to a point by Forero and Giannakis [11] concerning the LTS, it is impractical to find its global solution even for the small-scale problem due to the computational complexity of the combinatorial nature of (12).

3.3 Convex EDM model

A plausible approach to solving the problem (9) is to propose a convex relaxation of it. There are three parts to consider in order to achieve this purpose. One part is the ℓ_0 -norm regularization term $\|y\|_0$. As widely adopted, it can be replaced by its convex surrogate function $\|y\|_1$. The second part is the quadratic equation constraints on η_{ij} in (9). We recall the fact that $\eta_{ij} > 0$ implies $y_{ij} > 0$. We simply relax the quadratic equation constraints by

$$y_{ij} \geq 0, \quad (i, j) \in \mathcal{E}. \quad (13)$$

The remaining part is on the constraints between the embedding points $\{\mathbf{p}_i\}$ and the resulting distances d_{ij} . A nice fact is that the squared distances d_{ij}^2 are sufficient in describing those constraints and those squared distances also appear in the objective function in (9). In other words, the third part can be completely characterized by the squared distances, leading to our convex EDM optimization. We detail it below.

Let D be the EDM whose (i, j) -element is d_{ij}^2 , where d_{ij} satisfies the constraints in (9). Then D has the embedding dimension r . That is

$$\text{rank}(JDJ) = r. \quad (14)$$

Moreover, the distances between anchors should be preserved. That is

$$D_{ij} = \|\mathbf{p}_i - \mathbf{p}_j\|^2 = d_{ij}^2, \quad i, j = n_s + 1, \dots, n. \quad (15)$$

It then follows from (4) that a set of embedding points can be generated from this D and the corresponding points to the anchors can be matched to the true positions of the anchors through the Procrustes analysis.

For the matrix D to be an EDM, it follows from (1) and (2) that

$$D \in \mathcal{S}_h^n \quad \text{and} \quad -D \in \mathcal{K}_+^n. \quad (16)$$

For the objective function in (9), we have the following representation. Define three symmetric matrices H , \tilde{D} and Y respectively by

$$H_{ij} = \begin{cases} 1/\tilde{d}_{ij} & \text{if } (i, j) \in \mathcal{E} \\ 0 & \text{otherwise,} \end{cases} \quad \tilde{D}_{ij} = \begin{cases} \tilde{d}_{ij}^2 & \text{if } (i, j) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases} \quad \text{and} \quad Y_{ij} = \begin{cases} y_{ij} & \text{if } (i, j) \in \mathcal{E} \\ 0 & \text{otherwise.} \end{cases}$$

Then the quadratic term in the objective function of (9) is

$$\frac{1}{2} \|H \circ (D + Y - \tilde{D})\|^2. \quad (17)$$

where “ \circ ” denotes the Hadamard product (i.e., componentwise production) between two matrices of same size, and H is the weight matrix.

We put (13), (15), (16) and (17) together, and ignore the rank constraint (14), we obtain the following convex relaxation of (9):

$$\begin{aligned} \min_{D, Y} \quad & \frac{1}{2} \|H \circ (D + Y - \tilde{D})\|^2 + \rho \|y\|_1 \\ \text{s.t.} \quad & D_{ij} = d_{ij}^2, \quad i, j = n_s + 1, \dots, n \\ & D \in \mathcal{S}_h^n, \quad -D \in \mathcal{K}_+^n, \\ & y_{ij} \geq 0, \quad \forall (i, j) \in \mathcal{E}. \end{aligned} \quad (18)$$

It has a convex quadratic objective function with an l_1 penalized term, $n_a(n_a - 1)/2 + n$ equality constraints, $|\mathcal{E}|$ simple non-negativity constraints $y_{ij} \geq 0$, and the cone constraint $-D \in \mathcal{K}_+^n$. We will show that this type of problems can be efficiently solved by an alternating minimization method. We end this section with a remark.

Remark 1 Although we do not mainly focus on the single source problem, it is worth to note that there are two major differences between our proposed convex EDM model (18) and the SDP model proposed by [43], which mainly focuses on the single source case. Firstly, our EDM model has a clear statistical meaning, i.e., it is a convex relaxation of the LTS estimation by ignoring the rank constraint and relaxing the quadratic constraint in (9) to (13). Since the LTS is a well-known robust estimation for random variables with large biases but is computationally intractable, the proposed convex EDM model usually can provide high-quality estimation of unknown sensor locations even for the single source problem (see Section 5-(b) for more discussions). Secondly, the proposed convex EDM model has a nice structure that its dual problem can be solved efficiently by a convergent ADMM algorithm, which will be introduced in the following section. However, as mentioned in Section 1, it seems not clear for us how to solve the SDP model proposed by [43] efficiently in particular for large scale collaborative sensor network problems, since the complexity of those models would significantly slow down the state-of-art SDP solvers.

4 A convergent 3-block ADMM algorithm

In this section, we develop an efficient numerical method for the problem (18). We note that the weights in H corresponding to the indices not in \mathcal{E} are *zero*. They do not contribute to the objective function value. Therefore, we squeeze them out by a simple vectorization reformulation.

4.1 Reformulation by vectorization

To simplify our notation, we assume that there are m pairs of indices (i, j) in \mathcal{E} :

$$\mathcal{E} = \{(i_1, j_1), (i_2, j_2), \dots, (i_m, j_m)\}.$$

For each (i_k, j_k) , we define the symmetric matrix A_k , the scalars y_k and w_k respectively by

$$A_k := \frac{1}{2}(\mathbf{e}_{i_k} \mathbf{e}_{j_k}^T + \mathbf{e}_{j_k} \mathbf{e}_{i_k}^T), \quad y_k := y_{i_k j_k}, \quad w_k := \tilde{d}_{i_k j_k}^2.$$

For a [symmetric](#) matrix X , we have $\langle A_k, X \rangle = X_{i_k j_k}$. That is, A_k simply selects the (i_k, j_k) th element of matrix X when applied to it. We put y_k and w_k in the respective vectors:

$$\mathbf{y} := (y_1, y_2, \dots, y_m)^T \quad \text{and} \quad \mathbf{w} := (w_1, w_2, \dots, w_m)^T.$$

We further define the selection operator $\mathcal{A} : \mathcal{S}^n \rightarrow \mathbb{R}^m$ by

$$\mathcal{A}(X) := (\langle A_1, X \rangle, \dots, \langle A_m, X \rangle)^T, \quad X \in \mathcal{S}^n. \quad (19)$$

Define the inverse \mathbf{w}^{-1} of the weight vector \mathbf{w} to be its componentwise inverse:

$$\mathbf{w}^{-1} := (1/w_1, 1/w_2, \dots, 1/w_m)^T.$$

With the notation defined above, the quadratic term in the objective function in (18) becomes a \mathbf{w}^{-1} -weighted ℓ_2 norm of a vector:

$$\frac{1}{2} \|\mathcal{A}(D) + \mathbf{y} - \mathbf{w}\|_{\mathbf{w}^{-1}}^2.$$

Since the constraints in (18) are all linear, they can be easily vectorized. Define the linear operator $\mathcal{B}_1 : \mathcal{S}^n \mapsto \mathbb{R}^n$ by

$$\mathcal{B}_1(X) = \text{diag}(X).$$

We denote the zero vector in \mathbb{R}^n by \mathbf{b}_1 . We further define the linear operator $\mathcal{B}_2 : \mathcal{S}^n \mapsto \mathbb{R}^{n_a(n_a-1)/2}$ and the vector $\mathbf{b}_2 \in \mathbb{R}^{n_a(n_a-1)/2}$ such that for any pair (i, j) satisfying $n_s + 1 \leq i < j \leq n$ we have

$$\mathcal{B}_2(X) = X_{ij}, \quad (\mathbf{b}_2)_{ij} = d_{ij}^2,$$

where those d_{ij} 's are distances between the known anchors in (15). We finally arrive at the following vectorized form of our problem (18):

$$\begin{aligned} \min_{D, \mathbf{y}} \quad & \frac{1}{2} \|\mathcal{A}(D) + \mathbf{y} - \mathbf{w}\|_{\mathbf{w}^{-1}}^2 + \rho \langle \mathbf{e}, \mathbf{y} \rangle \\ \text{s.t.} \quad & \mathcal{B}(D) = \mathbf{b}, \quad \mathbf{y} \geq 0 \\ & -D \in \mathcal{K}_+^n, \end{aligned} \tag{20}$$

where $\mathcal{B} : \mathcal{S}^n \rightarrow \mathbb{R}^n \times \mathbb{R}^{n_a(n_a-1)/2}$ is the linear mapping defined by

$$\mathcal{B}(D) := (\mathcal{B}_1(D); \mathcal{B}_2(D)), \quad \mathbf{b} := (\mathbf{b}_1; \mathbf{b}_2).$$

4.2 Lagrangian dual problem

For the reasons that will become clear later, we will not solve the problem (20) directly. Instead, we will solve its Lagrangian dual problem and the solution of (20) will be obtained automatically. To derive the Lagrangian dual problem, we introduce the new variable $\mathbf{t} := \mathcal{A}(D) + \mathbf{y} - \mathbf{w}$ so that the problem (20) is equivalent to

$$\begin{aligned} \min_{D, \mathbf{t}, \mathbf{y}} \quad & \frac{1}{2} \|\mathbf{t}\|_{\mathbf{w}^{-1}}^2 + \rho \langle \mathbf{e}, \mathbf{y} \rangle + \delta_{\mathbb{R}_+^m}(\mathbf{y}) + \delta_{\mathcal{K}_+^n}(-D) \\ \text{s.t.} \quad & \mathcal{A}(D) + \mathbf{y} - \mathbf{t} = \mathbf{w} \\ & \mathcal{B}(D) = \mathbf{b}, \end{aligned}$$

where $\delta_{\mathbb{R}_+^m}(\cdot)$ and $\delta_{\mathcal{K}_+^n}(\cdot)$ are indicator functions over the sets \mathbb{R}_+^m and \mathcal{K}_+^n , respectively. Let the Lagrangian function of the above problem be defined by

$$\begin{aligned} L(D, \mathbf{y}, \mathbf{t}; \mathbf{z}_1, \mathbf{z}_2) := & \frac{1}{2} \|\mathbf{t}\|_{\mathbf{w}^{-1}}^2 + \rho \langle \mathbf{e}, \mathbf{y} \rangle + \delta_{\mathbb{R}_+^m}(\mathbf{y}) + \delta_{\mathcal{K}_+^n}(-D) \\ & + \langle \mathcal{A}(D) + \mathbf{y} - \mathbf{w} - \mathbf{t}, \mathbf{z}_1 \rangle + \langle \mathcal{B}(D) - \mathbf{b}, \mathbf{z}_2 \rangle, \end{aligned}$$

where $\mathbf{z}_1 \in \mathbb{R}^m$ and $\mathbf{z}_2 \in \mathbb{R}^{n+n_a(n_a-1)/2}$ are the Lagrangian multipliers corresponding to the two constraints. The Lagrangian dual problem is

$$\max_{\mathbf{z}_1, \mathbf{z}_2} \left\{ \min_{D, \mathbf{t}, \mathbf{y}} L(D, \mathbf{y}, \mathbf{t}; \mathbf{z}_1, \mathbf{z}_2) \right\}. \tag{21}$$

The insider optimization problem has a simple representation:

$$\begin{aligned} & \min_{D, \mathbf{t}, \mathbf{y}} L(D, \mathbf{y}, \mathbf{t}; \mathbf{z}_1, \mathbf{z}_2) \\ = & \min_D \left\{ \langle \mathcal{A}^*(\mathbf{z}_1) + \mathcal{B}^*(\mathbf{z}_2), D \rangle + \delta_{\mathcal{K}_+^n}(-D) \right\} + \min_{\mathbf{y}} \left\{ \rho \langle \mathbf{e}, \mathbf{y} \rangle + \langle \mathbf{z}_1, \mathbf{y} \rangle + \delta_{\mathbb{R}_+^m}(\mathbf{y}) \right\} \\ & + \min_{\mathbf{t}} \left\{ \frac{1}{2} \|\mathbf{t}\|_{\mathbf{w}^{-1}}^2 - \langle \mathbf{z}_1, \mathbf{t} \rangle \right\} - \langle \mathbf{w}, \mathbf{z}_1 \rangle - \langle \mathbf{b}, \mathbf{z}_2 \rangle. \\ = & -\delta_{\mathcal{K}_+^n}^*(\mathcal{A}^*(\mathbf{z}_1) + \mathcal{B}^*(\mathbf{z}_2)) - \delta_{\mathbb{R}_+^m}^*(-\rho \mathbf{e} - \mathbf{z}_1) - \frac{1}{2} \|\mathbf{z}_1\|_{\mathbf{w}}^2 - \langle \mathbf{w}, \mathbf{z}_1 \rangle - \langle \mathbf{b}, \mathbf{z}_2 \rangle \\ = & -\delta_{(\mathcal{K}_+^n)^\circ}(\mathcal{A}^*(\mathbf{z}_1) + \mathcal{B}^*(\mathbf{z}_2)) - \delta_{\mathbb{R}_+^m}(-\rho \mathbf{e} - \mathbf{z}_1) - \frac{1}{2} \|\mathbf{z}_1\|_{\mathbf{w}}^2 - \langle \mathbf{w}, \mathbf{z}_1 \rangle - \langle \mathbf{b}, \mathbf{z}_2 \rangle. \end{aligned}$$

Introducing

$$Z := \mathcal{A}^*(\mathbf{z}_1) + \mathcal{B}^*(\mathbf{z}_2),$$

the Lagrangian dual problem (21) of (20) takes the form:

$$\begin{aligned} \max & -\frac{1}{2}\|\mathbf{z}_1\|_{\mathbf{w}}^2 - \langle \mathbf{w}, \mathbf{z}_1 \rangle - \langle \mathbf{b}, \mathbf{z}_2 \rangle \\ \text{s.t.} & -\mathcal{A}^*\mathbf{z}_1 - \mathcal{B}^*\mathbf{z}_2 + Z = 0, \\ & \mathbf{z}_1 + \rho\mathbf{e} \geq 0, \quad Z \in (\mathcal{K}_+^n)^\circ. \end{aligned} \quad (22)$$

4.3 An ADMM algorithm

Compared to the primal problem (20), the dual problem (22) is well structured. It has three block-variables \mathbf{z}_1 , \mathbf{z}_2 and Z and they are separable (both in the objective and the constraints) and they are only linked by one equation. A natural algorithm would be to minimize each block-variable in turns until certain convergence is observed. One of the such algorithms is the recently proposed Alternating Direction Method of Multipliers (ADMM) by Sun et. al. [38] in the context of SDP and we found that its adaptation to the problem (22) is numerically very satisfactory. This is the main reason why we do not attempt to solve problem (20) directly. We present this algorithm below.

The algorithm actually *alternately* minimizes the associated Augmented Lagrangian defined by (note: here we cast (22) as a minimization problem)

$$\begin{aligned} L_\beta(\mathbf{z}_1, \mathbf{z}_2, Z; X) &:= \frac{1}{2}\|\mathbf{z}_1\|_{\mathbf{w}}^2 + \langle \mathbf{w}, \mathbf{z}_1 \rangle + \langle \mathbf{b}, \mathbf{z}_2 \rangle + \langle -\mathcal{A}^*\mathbf{z}_1 - \mathcal{B}^*\mathbf{z}_2 + Z, X \rangle \\ &+ \frac{\beta}{2}\|-\mathcal{A}^*\mathbf{z}_1 - \mathcal{B}^*\mathbf{z}_2 + Z\|^2, \end{aligned}$$

where $\beta > 0$ is a given parameter and $X \in \mathcal{S}^n$ is the Lagrangian multiplier corresponding to the equality constraint in (22). The algorithm generates a sequence $\{\mathbf{z}_1^k, \mathbf{z}_2^k, Z^k, X^k\}$ (index by k) according to the following steps.

Algorithm 1 Let $\beta > 0$ and $\alpha > 0$ be given. Choose the starting point $\mathbf{z}_1^0 \in \mathbb{R}^m$ such that $\mathbf{z}_1^0 + \rho\mathbf{e} \geq 0$, $Z^0 \in (\mathcal{K}_+^n)^\circ$, and $X^0 \in \mathcal{S}^n$ such that $\mathcal{B}X^0 = \mathbf{b}$. Set $\mathbf{z}_2^0 = -(\mathcal{B}\mathcal{B}^*)^{-1}\mathcal{B}(\mathcal{A}^*\mathbf{z}_1^0 + Z^0)$. Let $k = 0$. Perform the k -th iteration as follows:

Step 1 Compute

$$\mathbf{z}_1^{k+1} = \arg \min \left\{ L_\beta(\mathbf{z}_1, \mathbf{z}_2^k, Z^k; X^k) : \mathbf{z}_1 \geq -\rho\mathbf{e} \right\}.$$

Step 2 Compute so-called half step for \mathbf{z}_2 by $\mathbf{z}_2^{k+\frac{1}{2}} = \arg \min L_\beta(\mathbf{z}_1^{k+1}, \mathbf{z}_2, Z^k; X^k)$.

Step 3 Compute

$$Z^{k+1} = \arg \min_{Z \in (\mathcal{K}_+^n)^\circ} \left\{ L_\beta(\mathbf{z}_1^{k+1}, \mathbf{z}_2^{k+\frac{1}{2}}, Z; X^k) \right\}.$$

Step 4 Compute the full step for \mathbf{z}_2 by $\mathbf{z}_2^{k+1} = \arg \min L_\beta(\mathbf{z}_1^{k+1}, \mathbf{z}_2, Z^{k+1}; X^k)$.

Step 5 Update X by $X^{k+1} = X^k - \alpha\beta(\mathcal{A}^\mathbf{z}_1^{k+1} + \mathcal{B}^*\mathbf{z}_2^{k+1} - Z^{k+1})$.*

The convergence analysis of Algorithm 1 as well as the necessity of having the half-step update of \mathbf{z}_2 can be found in [38]. We omit the details here for simplicity. However, it is an important fact

to note that the optimization problems from Step 1 and 4 are all projection problems and all have close-form solutions. Actually, for each $k \geq 0$, it is easy to check that

$$\mathbf{z}_1^{k+1} = \max \{ \widehat{W}^{-1} q^k, -\rho \mathbf{e} \}, \quad (23)$$

$$\mathbf{z}_2^{k+1/2} = -(\mathcal{B}\mathcal{B}^*)^{-1} \mathcal{B}(\mathcal{A}^* \mathbf{z}_1^{k+1} - Z^k), \quad (24)$$

$$Z^{k+1} = -\Pi_{S_+^n} \left(J(X^k/\beta - \mathcal{A}^* \mathbf{z}_1^{k+1} - \mathcal{B}^* \mathbf{z}_2^{k+1/2}) J \right) \quad (25)$$

$$\mathbf{z}_2^{k+1} = -(\mathcal{B}\mathcal{B}^*)^{-1} \mathcal{B}(\mathcal{A}^* \mathbf{z}_1^{k+1} - Z^{k+1}), \quad (26)$$

where $\widehat{W} := \text{Diag}(w) + \beta I$ and $q^k := \mathcal{A}(X^k + \beta(Z^k - \mathcal{B}^* \mathbf{z}_2^k)) + \mathbf{w}$. We note that the derivation of (24) and (26) used the fact $\mathcal{B}^*(X^k) = \mathbf{b}$ for all iterates X^k due to the initial choice of X^0 . The formula (25) is due to (3) and the fact that for any $A \in \mathcal{S}^n$,

$$\Pi_{(\mathcal{K}_+^n)^\circ}(A) = A - \Pi_{\mathcal{K}_+^n}(A) = -\Pi_{S_+^n}(-JAJ).$$

Replacing A above by $-(X^k/\beta - \mathcal{A}^* \mathbf{z}_1^{k+1} - \mathcal{B}^* \mathbf{z}_2^{k+1/2})$ gets the solution for Z^{k+1} .

4.4 Computational complexity and stopping criterion

We refer to Subsection 4.1 for the following discussion. Since the matrices in defining the linear operators \mathcal{A} and \mathcal{B} select certain elements X_{ij} when they are applied to the matrix X , the operator $\mathcal{B}\mathcal{B}^*$ is a diagonal operator. The computational complexity in the formulae of (23), (24), and (26) is about $O(n^2)$. The major computation in (25) is the spectral decomposition of a symmetric matrix. Therefore, their implementations are fairly straightforward. The major purpose of this part is to explain when we should terminate the algorithm.

We first note that the primal problem (20) always has an optimal solution. To ensure the dual problem has an optimal solution, we assume the generalized Slater condition holds for the primal problem: there exist $\mathbf{y} > 0$ and $-D \in \text{ri}(\mathcal{K}_+^n)$ such that $\mathcal{B}(D) = \mathbf{b}$, where $\text{ri}(\mathcal{K}_+^n)$ is the relative interior of the cone \mathcal{K}_+^n . Then, since the optimization problem (20) is convex, we know from [34, Corollary 28.2.2 and Corollary 28.3.1] that \bar{X} is the corresponding optimal solution of (20) if and only if the following KKT condition holds at some point $(\bar{X}, \bar{\mathbf{z}}_1, \bar{\mathbf{z}}_2, \bar{Z})$:

$$\begin{cases} 0 \in -\mathcal{A}(X) + \partial f(\mathbf{z}_1), \\ 0 \in X + \partial g(Z), \\ \mathcal{B}(X) - \mathbf{b} = 0, \\ \mathcal{A}^* \mathbf{z}_1 + \mathcal{B}^* \mathbf{z}_2 - Z = 0, \end{cases} \quad (27)$$

where $f(\mathbf{z}_1) := \frac{1}{2} \|\mathbf{z}_1\|_{\mathbf{w}}^2 + \langle \mathbf{w}, \mathbf{z}_1 \rangle + \delta_{\mathbb{R}_+^m}(\mathbf{z}_1 + \rho \mathbf{e})$, $g(Z) := \delta_{(\mathcal{K}_+^n)^\circ}(Z)$, and ∂f , ∂g are the subdifferentials of the convex functions f and g , respectively, at the concerned points.

We would like to quantify the KKT condition (27) at the k th iteration. Since we always have $\mathcal{B}(X^k) - \mathbf{b} = 0$ for all X^k generated by Algorithm 1 (its proof can be patterned after the same result in [38]), the third condition in (27) is automatically satisfied. The second condition is the complementarity condition is equivalent with $-X \in \mathcal{K}_+^n$, $Z \in (\mathcal{K}_+^n)^\circ$ and $\langle X, Z \rangle = 0$. The first and the last condition in (27) are respectively the primal and dual feasibility. For the first (inclusion) condition, we now quantify the distance from 0 to the set $\mathcal{A}(X) + \partial f(\mathbf{z}_1)$ at $(X^{k+1}, \mathbf{z}_1^{k+1})$. We note that at the k th iteration, Z^{k+1} is the optimal solution of the following convex problem

$$\min f(\mathbf{z}_1) + \langle \mathcal{A}(X^k), \mathbf{z}_1 \rangle + \frac{\beta}{2} \|\mathcal{A}^* \mathbf{z}_1 + \mathcal{B}^* \mathbf{z}_2^k - Z^k\|^2.$$

Thus, we have $0 \in \partial f(\mathbf{z}_1^{k+1}) + \mathcal{A}(X^k) + \beta\mathcal{A}(\mathcal{A}^*\mathbf{z}_1^{k+1} + \mathcal{B}^*\mathbf{z}_2^k - Z^k)$, which implies

$$\mathcal{A}\left(X^{k+1} - X^k - \beta(\mathcal{A}^*\mathbf{z}_1^{k+1} + \mathcal{B}^*\mathbf{z}_2^k - Z^k)\right) \in \mathcal{A}(X^{k+1}) + \partial f(\mathbf{z}_1^{k+1}).$$

Therefore,

$$\text{dist}(0, \mathcal{A}X^{k+1} + \partial f(\mathbf{z}_1^{k+1})) \leq \left\| \mathcal{A}\left(X^{k+1} - X^k - \beta(\mathcal{A}^*\mathbf{z}_1^{k+1} + \mathcal{B}^*\mathbf{z}_2^k - Z^k)\right) \right\|.$$

For the k -th iteration, we define the measurements of the infeasibilities of the primal and dual problems, and the violation of the complementary condition by

$$\begin{cases} R_p := \left\| \mathcal{A}\left(X^{k+1} - X^k - \beta(\mathcal{A}^*\mathbf{z}_1^{k+1} + \mathcal{B}^*\mathbf{z}_2^k - Z^k)\right) \right\|, \\ R_d := \|\mathcal{A}^*\mathbf{z}_1^{k+1} + \mathcal{B}^*\mathbf{z}_2^{k+1} - Z^{k+1}\|, \\ R_c := \max\{|\langle X^{k+1}, Z^{k+1} \rangle|, \| -X^{k+1} - \Pi_{\mathcal{K}_+^n}(-X^{k+1}) \|\}. \end{cases} \quad (28)$$

We terminate the algorithm if the KKT condition are met, i.e.,

$$\max\{R_p, R_d, R_c\} \leq \text{tol}, \quad (29)$$

where tol is the given tolerance.

5 Numerical experiments

In this section, we demonstrate the effectiveness of the proposed convex EDM model (18) by testing Algorithm 1 on both simulations and real-world examples. Note that, in all the tested problems, we do not assume any prior knowledge about the noise error and the NLOS condition such as the noise power, the NLOS status or distribution parameters. For comparison purpose, we aimed to select the best methods available in the literature. Unfortunately, despite the large number of publications in this area, there are few of the methods whose implementations are publicly available. In addition to our own EDM method, we choose to implement 8 of such methods recently proposed in literature. Their descriptions can be found in Table 1. It should be noted that we only use IPPM, ML and ML-EDM as a benchmark for comparison, since we do not know any NLOS information a priori in practice.

Methods	Description
Huber	Huber estimation [20]
RDA-Stage I RDA-Stage II	the 2-stage robust distributed algorithm [49]
SDP	the SDP based embedding [8] with the state-of-art SDP solver SDPT3 [40]
SDP-single	the SDP based embedding in [43] for single source localization
IPPM	the iterative parallel projection method in [22] with EDM solution initialization for LOS links only
ML	Maximum likelihood estimation [23] with true value initialization for LOS links only
ML-EDM	Maximum likelihood estimation with EDM solution initialization for LOS links only

Table 1: Descriptions of the methods tested

As a performance metric, we adopt the commonly used criterion: the Root of Mean Square Distance (RMSD) for the estimated sensor positions $\hat{\mathbf{p}}_1, \dots, \hat{\mathbf{p}}_{n_s}$:

$$\text{RMSD} := \frac{1}{\sqrt{n_s}} \left(\sum_{i=1}^{n_s} \|\hat{\mathbf{p}}_i - \mathbf{p}_i\|^2 \right)^{1/2}.$$

5.1 Parameters settings

All the algorithms are implemented in Matlab (R2014b) and all the numerical experiments were run under a Windows 8.1 64-bit system on an Intel 4 Cores i7 2.4GHz CPU with 8GB memory. The termination tolerance in (29) is chosen to be $\text{tol} = 10^{-4}$ for our EDM method and the initial sensor locations are set at the origin. **In general, the penalty parameter ρ is inversely proportional to the number of NLOS connections.** For the SDP solver, we used the default parameters in SDPT3 (through the CVX toolbox [16]). For **Huber**, we set $K = 2\sigma$ (see [20]). For the 2-stage robust distributed algorithm (**RDA-Stage I & II**), we used the parameters $K_1 = 2\sigma$ and $K_2 = 0.1\sigma$, which are the same as in [49]. The gradient descent algorithm is employed to solve the corresponding unconstrained optimization problems in **Huber** and **RDA-Stage I & II**. Instead of the fixed step size used in [49], we used the Armijo backtracking line search method to compute the suitable step size to ensure a sufficient decrease in the objective function value. The stopping criteria for **Huber** and **RDA-Stage I & II** are the same as those in [49], i.e., the estimators of sensor positions at two consecutive iterations are smaller than the termination tolerance $\text{tol} = 10^{-4}$. However, since the unconstrained optimization problems are non-convex, there is no theoretical convergence guarantee for **Huber** and **RDA-Stage II to obtain a global optimal solution**. We also terminate the algorithms when the maximum iterations (1000 in our implementation) are reached. As in [49], the initial sensor positions for these three algorithms (**Huber** and **RDA-Stage I & II**) are selected as Gaussian random variables with mean equal to the true sensor positions and standard deviation of 10 meters.

5.2 A toy example

In order to understand what our EDM method would achieve, we study an example of small networks, where there are $n_a = 8$ anchors located at the boundary of the square $[0, 10]\text{m} \times [0, 10]\text{m}$ (i.e., $(0, 0)\text{m}$, $(5, 0)\text{m}$, $(10, 0)\text{m}$, $(0, 5)\text{m}$, $(10, 5)\text{m}$, $(0, 10)\text{m}$, $(5, 10)\text{m}$ and $(10, 10)\text{m}$) and there is only one source located randomly in the square. Assume that the source node can communicate with all the anchors. The i.i.d. noise errors ξ_{ij} in (7) follow a zero-mean Gaussian distribution with standard deviation $\sigma = 1\text{m}$. We further assume that there are two sensor-anchor measurements containing the NLOS errors, which are exponential random variables with parameter $\gamma = 10\text{m}$. The RMSD and **empirical cumulative distribution function (CDF)**¹ of different methods are reported in Table 2 and Figure 1, respectively. It can be clearly seen that our EDM method (**with $\rho = 360$**) performs well in terms of RMSD. Most importantly, it is worth to note that with high probability (80 out of 100 MC simulation runs), the nonzero elements in the solution y^* obtained by the EDM method correctly identified the NLOS links. Moreover, if there is only one NLOS link, the probability went up to 93% and RMSD reduces to 1.08m (**with $\rho = 180$**). However, other methods such as the **SDP-single** failed to identify the NLOS links for this example. This may be the partial reason why the proposed method outperforms other techniques in most cases. It can be observed from Table 2 that in both cases, the RMSD of IPPM and ML-EDM equal to ML, which indicates that the EDM provides a good estimation. Note that our proposed method does not need any prior information neither on the measurement noise nor on the NLOS errors.

¹ Let x_1, \dots, x_n be independent, identically distributed real random variables. The corresponding empirical distribution function $F_n(t)$ is defined as $F_n(t) = \frac{1}{n} \sum_{i=1}^n \mathbf{1}_{x_i \leq t}$, where $\mathbf{1}_{x_i \leq t}$ is the indicator of event $x_i \leq t$.

NLOS links	EDM	SDP-single	RDA-Stage I	RDA-Stage II	Huber	IPPM	ML-EDM	ML
2	1.21	2.20	1.62	1.32	2.96	1.06	1.06	1.06
1	1.08	1.32	1.36	1.14	2.97	1.05	1.05	1.05

Table 2: Performance comparison for the toy example in terms of RMSD (in meters).

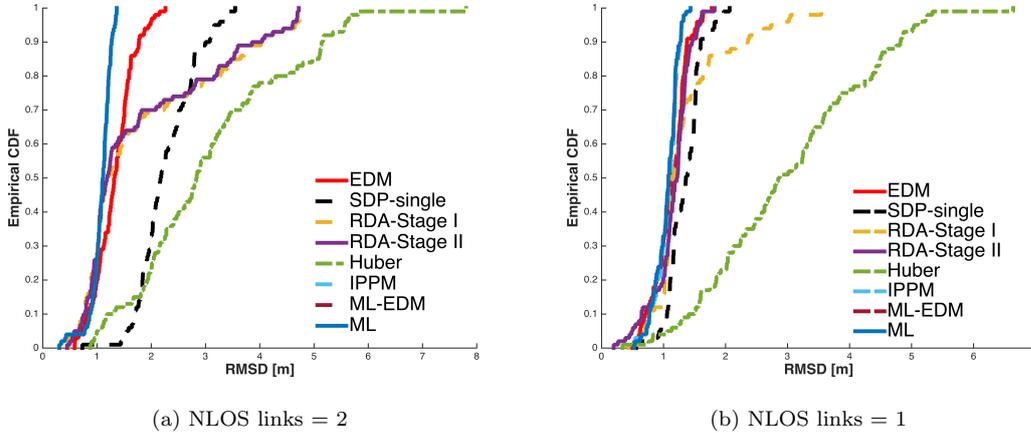


Fig. 1: Performance comparison for the toy example.

5.3 Test problems

We tested three types of problems. The first two examples have been commonly tested and are a class of randomly generated problems. The last one is from real data. All problems are described as follows for easy reference.

Example 1 (T1) This is a class of sensor network localization problems tested in [49] with $n_s = 50$ sensors and $n_a = 4$ anchors. The sensors are randomly generated in the unit square $[0, 10]\text{m} \times [0, 10]\text{m}$ and the anchors are located at the four corners. The i.i.d. noise errors ξ_{ij} in (7) follow a zero-mean Gaussian distribution with standard deviation varying from 0.5m to 3m and the NLOS error η_{ij} in (7) are modelled as an exponential random variable with parameter $\gamma = 10\text{m}$. We let the physical communication range $R = 10\text{m}$. Let \mathbb{P}_{NLOS} be the probability of a link being NLOS. The bigger \mathbb{P}_{NLOS} is, the more there are NLOS links.

Example 2 (T2) This is a class of sensor network localization problems tested in [8, Section IV-C]. The $n_s = 40$ sensors are randomly generated in the unit square $[-20, 20]\text{m} \times [-20, 20]\text{m}$ and the $n_a = 8$ anchors are located at the boundary $(20, 20)\text{m}$, $(-20, 20)\text{m}$, $(20, -20)\text{m}$, $(-20, -20)\text{m}$, $(0, 20)\text{m}$, $(20, 0)\text{m}$, $(0, -20)\text{m}$, $(-20, 0)\text{m}$. The i.i.d. noise errors ξ_{ij} in (7) follow a zero-mean Gaussian distribution with standard deviation varying from 1m to 6m and the NLOS error η_{ij} in (7) are modelled as an exponential random variable with parameter γ varying from 1m to 6m. The physical communication range $R = 25\text{m}$ and the NLOS probability $\mathbb{P}_{NLOS} = 0.3$ are same as those used in [8].

Example 3 (T3) We use the real data obtained by the channel measurement experiment conducted at the Motorola facility in Plantation, which is reported in [27]². The experiment environment is an office area which is partitioned by cubicle walls. Forty-four device locations are identified within a $14\text{m} \times 13\text{m}$ area. Four of the devices ($n_a = 4$) are chosen to be anchors and remaining locations are

² The data can be downloaded from <http://web.eecs.umich.edu/~hero/localize/>.

T1	\mathbb{P}_{NLOS}	σ	EDM	SDP	RDA-Stage I	RDA-Stage II	Huber	IPPM	ML-EDM	ML
RMSD (in m)	0.05	0.5	0.30	1.09	0.86	0.43	1.91	0.32	0.21	0.21
		1	0.54	1.30	1.52	0.95	2.20	0.75	0.35	0.35
		1.5	1.02	1.36	2.04	1.34	2.90	1.19	0.52	0.51
		2	1.61	1.72	2.50	1.82	3.20	1.56	0.70	0.69
		2.5	1.79	2.20	2.79	2.50	3.88	2.05	0.88	0.87
		3	2.34	2.53	3.13	2.87	4.14	2.70	1.05	1.03
	0.5	0.5	0.47	3.85	1.57	1.25	5.67	0.58	0.24	0.24
		1	0.67	3.95	2.40	2.16	6.04	1.14	0.49	0.48
		1.5	1.14	4.07	2.72	2.68	6.25	1.77	0.76	0.74
		2	1.72	4.21	2.52	2.25	6.51	2.42	0.98	0.95
		2.5	2.11	4.51	3.29	3.15	7.63	2.69	1.56	1.41
		3	2.63	4.73	3.58	3.48	7.01	3.66	1.61	1.47
	0.95	0.5	1.35	5.85	3.86	4.04	9.35	1.64	1.34	0.93
		1	2.70	5.96	3.96	4.61	9.40	3.06	2.56	1.59
		1.5	2.80	5.79	3.95	5.27	9.59	3.38	2.96	2.07
		2	2.91	5.85	4.16	4.93	9.46	3.70	2.97	2.36
		2.5	2.52	5.92	4.09	5.47	9.71	3.90	3.41	2.61
		3	2.92	6.13	4.19	5.61	9.91	4.47	3.77	2.70
CPU (in s)	0.05	0.5	0.14	11.39	0.33	0.62	0.61	0.97	0.01	0.01
		1	0.43	12.88	0.42	0.65	0.64	1.26	0.03	0.03
		1.5	0.65	12.59	0.40	0.64	0.64	1.39	0.05	0.03
		2	0.83	12.29	0.39	0.62	0.51	1.64	0.05	0.04
		2.5	0.14	12.29	0.39	0.60	0.67	2.10	0.05	0.05
		3	0.16	11.80	0.38	0.53	0.49	2.44	0.06	0.07
	0.5	0.5	0.09	12.38	0.40	0.62	0.64	1.15	0.02	0.01
		1	0.18	12.49	0.43	0.65	0.65	1.43	0.03	0.02
		1.5	0.29	12.61	0.41	0.65	0.65	1.70	0.04	0.03
		2	0.39	12.22	0.40	0.62	0.64	1.97	0.05	0.05
		2.5	0.40	13.21	0.50	0.57	0.60	2.21	0.15	0.16
		3	0.52	11.60	0.48	0.56	0.50	2.10	0.06	0.05
	0.95	0.5	0.11	12.11	0.45	0.65	0.65	0.31	0.04	0.04
		1	0.19	12.46	0.42	0.66	0.66	0.34	0.05	0.05
		1.5	0.22	12.36	0.41	0.66	0.66	0.33	0.05	0.04
		2	0.27	12.40	0.43	0.65	0.65	0.33	0.05	0.05
		2.5	0.21	11.94	0.48	0.63	0.63	0.35	0.06	0.05
		3	0.26	12.34	0.39	0.61	0.61	0.36	0.06	0.05

Table 3: Performance comparison for Example 1 with different NLOS probability \mathbb{P}_{NLOS} and different noise standard deviation σ .

unknown. In this experiment, each node can communicate with all other nodes. We use the original time-of-arrival (TOA) to obtain the pairwise range measurements. This means that almost all of the measurements have large positive NLOS errors due to the environment limitation. For comparison, we consider three scenarios tested in [49], i.e., (i) the raw measurements; (ii) the moderate NLOS propagation; (iii) the small NLOS propagation.

5.3.1 Performance comparison.

In this part, we make detailed comparison between all the algorithms that we have tested based on our extensive numerical experiments on the three test problems. We describe them below.

Comparison on T1. We set $n_a = 4$ and $n_s = 50$ in Example 1. The penalty parameter ρ in the EDM method is chosen as follows: $\rho = 9 \times 10^4$ for $\mathbb{P}_{NLOS} = 0.05$, $\rho = 2 \times 10^4$ for $\mathbb{P}_{NLOS} = 0.5$ and $\rho = 10^4$ for $\mathbb{P}_{NLOS} = 0.95$. The performance in terms of RMSD and CPU for all the methods is summarized in Table 3 and Figure 2. As can be observed from Table 3 and Figure 2, the EDM method works best in terms of RMSD. It is worth to notice that the distributed techniques such as the RDA-Stage I, RDA-Stage II, IPPM and Huber methods have an important feature that

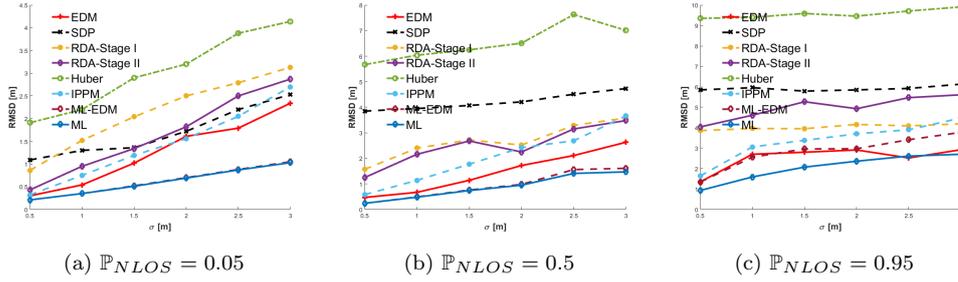


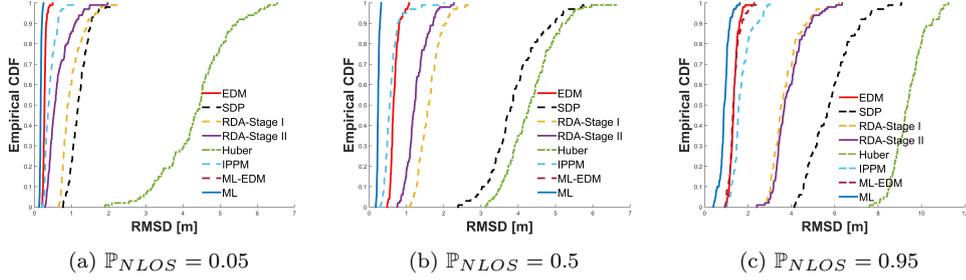
Fig. 2: Performance comparison for Example 1 with different σ in terms of RMSD (in meters).

they can be implemented in a parallel manner, which means that the computational complexity of those methods should be roughly the total time divided by the number of sensors plus the total communication time. As would have been expected, RMSD of all methods gets worse as the NLOS contamination level and the noise standard deviation increase. It is interesting to note from Table 3 that if the NLOS probability \mathbb{P}_{NLOS} is small, the ML-EDM and ML have very similar RMSD, which indicates that the proposed EDM method can provide a good estimator. An amazing observation is that this nice property still holds at 50% level of contamination, and even at 95% of NLOS contamination, our EDM still returns a reasonable RMSD = 1.35m (vs 5.85m by SDP) when the noise σ is small ($= 0.5$), which means that it can remove those NLOS biases across nearly all the NLOS measurements. This is rather surprising (and indeed a welcoming feature) as our model is based on the assumption that only a small percentage of the measurements are of NLOS.

As observed by [43], the second stage of RDA might not improve the localization performance for the high NLOS contamination. This is confirmed in our test. For example, in the cases $\mathbb{P}_{NLOS} = 0.95$ and $\sigma = 0.5\text{m}$ (the same case considered in [43]), we see from Table 3 that RMSD respectively increased from 3.86m to 4.04m. The CDF of different methods in this case are also plotted in Figure 3. We also tested Example 1 with different physical communication range R . In particular, for the case that $\sigma = 0.5\text{m}$ and $\mathbb{P}_{NLOS} = 0.5$, we let R decrease from 10m to 2m, while the average node connectivity decreases from 26.4% to 2.1%. The average RMSD of different methods after 100 MC simulation runs are reported in Table 4. It can be observed that the RMSD of all methods gets worse as the node connectivity decreases. The EDM still outperforms other methods even for the low connectivity case.

Finally, we consider the following modifications of Example 1. Let the anchors be located in the middle of the square, i.e., (3, 3)m, (7, 3)m, (7, 7)m and (3, 7)m with $\sigma = 1\text{m}$, $\gamma = 10\text{m}$, $R = 10\text{m}$ and $\mathbb{P}_{NLOS} = 0.05$. Consider Case (1): some sensors are outside the convex hull of anchors (see Figure 4a); and Case (2): all sensors are outside the convex hull of anchors (see Figure 4c). The RMSD of different methods for these two examples are summarized in Table 5 and CDF of different methods can be found from Figure 4b and Figure 4d, respectively. By comparing with the RMSD of Example 1 listed in Table 3, we can see that unlike other methods, our proposed EDM method seems not effected by the geometric conditions of the problem. For both cases, EDM works well just like the original Example 1. Overall, it can be clearly observed that, in all scenarios with different NLOS contamination levels and different noise levels, the estimation errors of our convex EDM method are much smaller than the others.

Comparison on T2. The main purpose of testing Example 2 is to see if we can reproduce the best results reported by other methods so as to ensure that our implementations of those methods are up to a high standard. Therefore, we tested the same scenarios as in [8]. we first fix the NLOS error mean $\gamma = 4\text{m}$ but let the standard deviation of measurement noise σ vary from 1m to 6m (see Table 6 and Figure 5a). It can be observed that we are able to reproduce the results similar to that reported in [8, Fig. 9]. We also observe that the performances of SDP, RDA-Stage II and

Fig. 3: CDF of different methods for Example 1 with $\sigma = 0.5\text{m}$.

R	connectivity (%)	EDM	SDP	RDA-Stage I	RDA-Stage II	Huber	IPPM	ML-EDM	ML
10	26.4	0.47	3.85	1.57	1.25	5.67	0.58	0.24	0.24
8	20.9	0.88	4.37	3.30	3.20	6.73	0.90	0.30	0.29
6	14.4	1.07	4.82	3.95	3.95	6.69	1.16	0.48	0.37
4	8.0	2.13	5.52	4.97	5.01	7.47	2.24	1.54	0.63
2	2.1	6.29	9.04	9.40	9.42	10.32	6.28	6.14	0.54

Table 4: Performance comparison for Example 1 with different communication range R (in meters) in terms of RMSD (in meters).

Problem T1	EDM	SDP	RDA-Stage I	RDA-Stage II	Huber	IPPM	ML-EDM	ML
Case (1)	0.57	1.61	2.47	1.64	3.40	0.89	0.37	0.37
Case (2)	0.59	1.71	3.16	2.13	3.88	0.91	0.37	0.37

Table 5: Performance comparison for the two cases in Figure 4 in terms of RMSD (in meters).

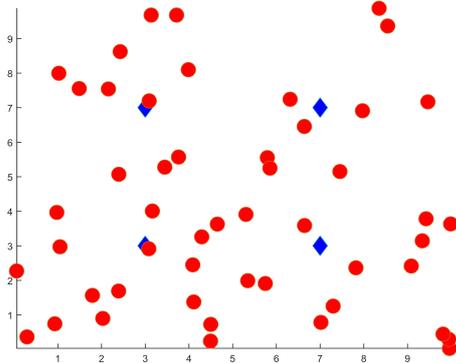
σ	EDM	SDP	RDA-Stage I	RDA-Stage II	Huber	IPPM	ML-EDM	ML
1	0.87	1.54	2.06	1.65	2.28	1.04	0.52	0.51
2	1.41	2.06	2.87	2.20	3.79	2.02	1.03	1.03
3	2.17	2.73	3.51	2.69	5.13	2.93	1.56	1.55
4	2.88	3.53	4.39	3.44	6.44	4.14	2.12	2.10
5	3.87	4.34	4.86	3.99	7.55	5.12	2.61	2.59
6	4.91	5.15	5.64	5.37	8.84	6.10	3.19	3.10

Table 6: Performance comparison for Example 2 with different σ (in meters) in terms of RMSD (in meters) .

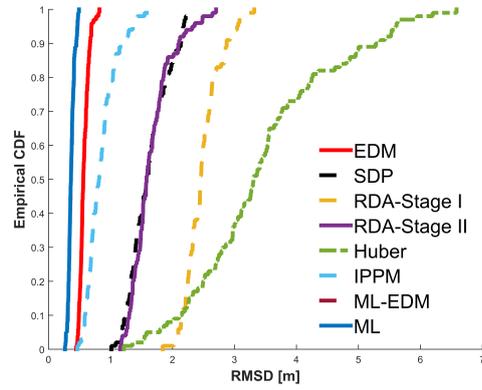
IPPM are at a same level for this case and our EDM (with $\rho = 8 \times 10^4$) produced better results than them. Obviously, ML and ML-EDM produced the best results as they used the information of which links being of NLOS.

We further compared different methods by fixing the standard deviation of measurement noise $\sigma = 4\text{m}$ but varying the mean of NLOS error from 1m to 6m (see Table 7 and Figure 5b). It is interesting to notice that for all methods, the localization accuracy does not change linearly with the change of γ , which is also reported in [8, Section IV-C].

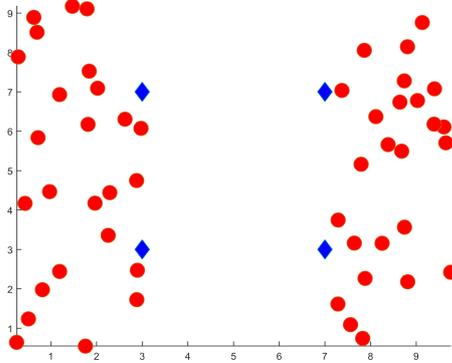
Comparison on T3. For Example 3, we compare our proposed EDM method with the SDP and RDA-Stage I & II under different NLOS propagation levels. Firstly, the performance comparison for Example 3 with the raw measurements can be seen from the localization plots in Figure 6. Since the different choices of initial sensor positions may lead to different estimation results for RDA-Stage I & II, the MC simulations are implemented under 100 runs. The corresponding CDF of the estimation error of all the methods are plotted in Figure 7a. From the results, we observe that both the proposed convex EDM method (with $\rho = 10^4$) and the SDP based method [8] are



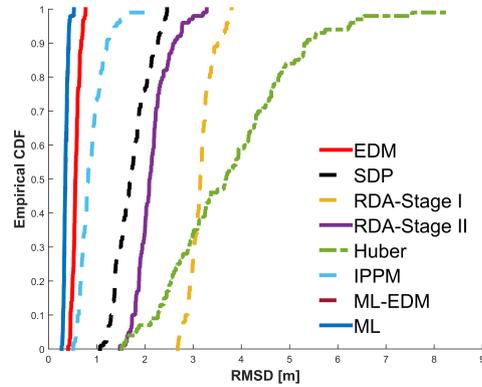
(a) True positions of sensors: some sensors are outside of the convex hull of 4 anchors (diamonds in blue)



(b) CDF of different methods for the data in (a).



(c) True positions of sensors: all sensors are outside of the convex hull of 4 anchors (diamonds in blue)



(d) CDF of different methods for the data in (c).

Fig. 4: Performance comparison for the modifications of Example 1 with the 4 anchors in blue inside the square $[0, 10]\text{m} \times [0, 10]\text{m}$.

γ	EDM	SDP	RDA-Stage I	RDA-Stage II	Huber	IPPM	ML-EDM	ML
1	2.78	3.57	4.04	3.06	6.36	4.10	2.09	2.08
2	2.76	3.51	4.11	3.18	6.36	4.15	2.08	2.06
3	2.86	3.51	4.21	3.32	6.41	4.05	2.10	2.09
4	2.88	3.53	4.39	3.44	6.44	4.14	2.12	2.10
5	2.89	3.62	4.37	3.55	6.56	4.04	2.10	2.07
6	2.92	3.71	4.37	3.56	6.63	4.01	2.09	2.07

Table 7: Performance comparison for Example 2 with different γ (in meters) in terms of RMSD (in meters) .

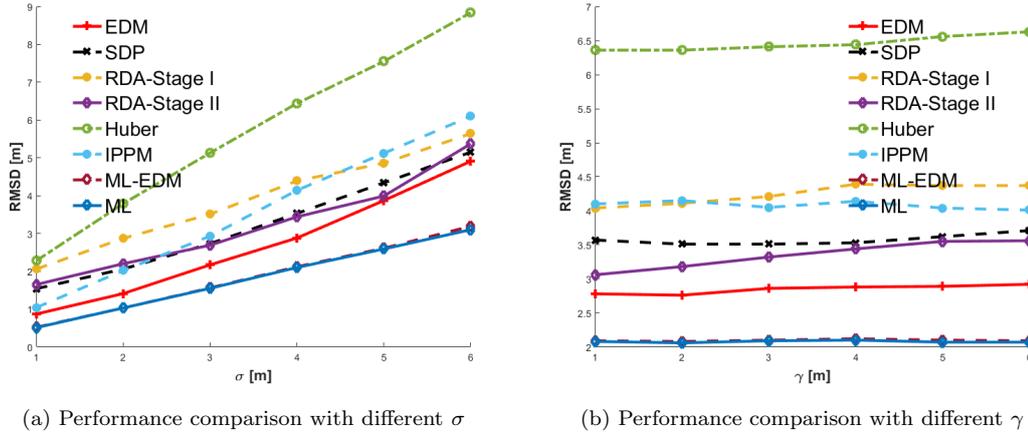


Fig. 5: Performance comparison for Example 2 in terms of RMSD (in meters) .

NLOS level	EDM	SDP	RDA-Stage I	RDA-Stage II
raw (large)	1.38	1.54	1.95	1.90
moderate	1.35	1.36	1.79	1.44
small	0.94	1.07	1.75	1.14

Table 8: Performance comparison for Example 3 with different NLOS propagation levels in terms of RMSD (in meters).

able to obtain high-quality estimation of the unknown sensor locations. Moreover, the localization quality obtained by the proposed method (EDM) is better than that obtained by SDP in terms of the estimation errors and is among the best for this widely tested problem.

We further tested the other two scenarios (the moderate and small NLOS propagations), which were also tested by [49]. We plot the CDF of all methods for the two cases in Figure 7b and 7c. The corresponding RMSD are listed in Table 8. From Figure 6 and Table 8, we can see that EDM and SDP returned similar results with EDM (with $\rho = 3 \times 10^4$ for the moderate NLOS propagation case and $\rho = 9 \times 10^4$ for the small NLOS propagation case) being slightly better, and both returned better results than RDA-Stage II and RDA-Stage I. However, RDA-Stage II improved RDA-Stage I in all three scenarios (large, moderate and small NLOS propagations).

5.3.2 Performance on large scale networks.

This final part is to test the capability of our model on large scale problems that are beyond those methods tested before. Such large networks are generated from Example 1 with growing number of unknown sensors. The large number of links presents a great deal of challenge for the SDP model and SDPT3 solver simply run out of memory for such tested problems.

In the test, we consider 5 different scenarios where the sensor number is chosen to be 100, 500, 1000, 1500 and 2000. The probability of a link being NLOS is 0.05 and the physical communication range $R = 10\text{m}$. For example, when $n_s = 2000$, the number of edges $|\mathcal{E}|$ is nearly 2 millions. The numerical performance of the proposed EDM is reported in Table 9. We also include two plots of localizations of 500 and 1000 sensors in Figure 8. We note that for such large scale problems, the primal and dual feasibility violations are very small at 10^{-5} and the RMSD is also very reasonable. Such features have been well reflected in the two plots in Figure 8, where the true locations (indicated as \circ) match well with the computed locations (indicated by $*$). Furthermore, the computing

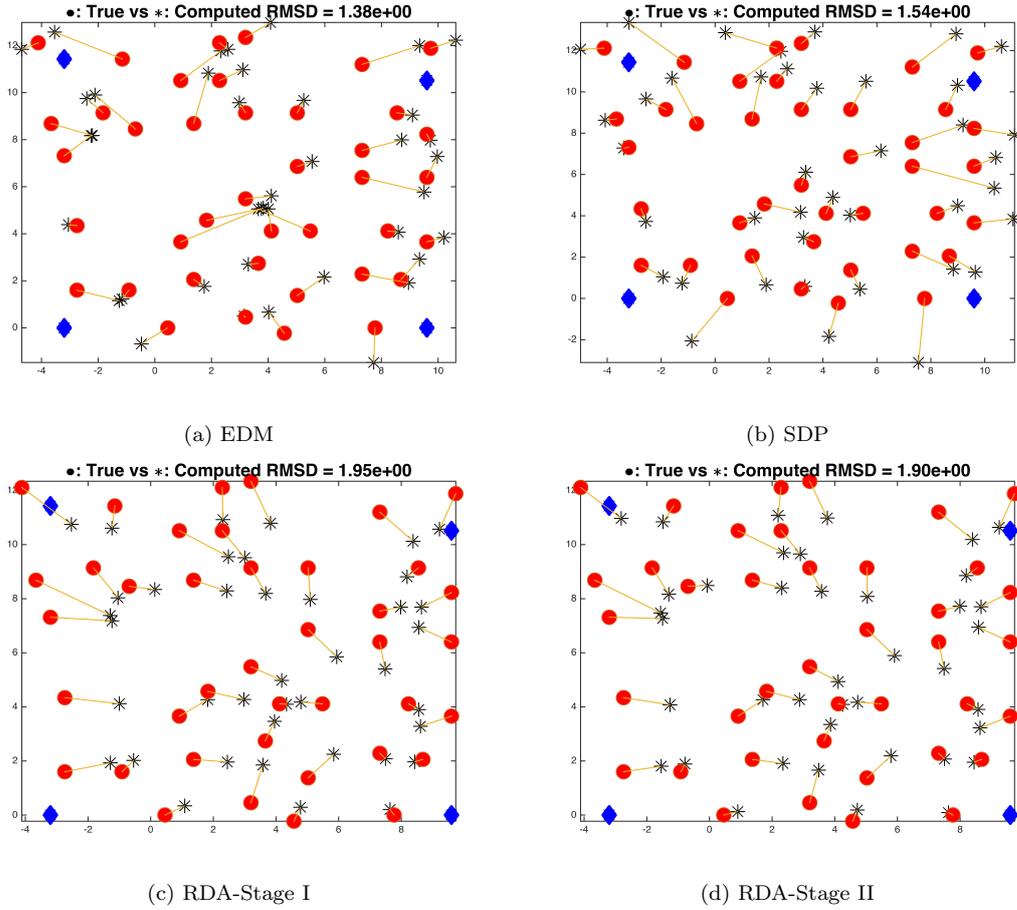


Fig. 6: Performance comparison for Example 3 using the raw measurements with the 4 anchors in blue.

n_a	n_s	$ \mathcal{E} $	R_p	R_d	R_c	RMSD	cpu(s)
4	100	5145	7.95e-05	9.27e-05	9.90e-05	0.2106	0.71
	500	122820	9.98e-05	7.02e-05	9.28e-05	0.0985	15.53
	1000	488481	9.99e-05	9.46e-05	8.05e-05	0.0695	91.41
	1500	1100475	9.94e-05	9.53e-05	6.91e-05	0.0606	242.11
	2000	1949639	9.95e-05	7.95e-05	6.49e-05	0.0578	504.31

Table 9: Numerical performance of EDM for large-scale problems.

time is very promising. For example, for the largest problem $n_s = 2000$, it took less than 9 minutes to reach the required stopping accuracy.

If the communication range R decrease, then the performance of EDM gets worse as the node connectivity decreases. The detailed numerical results for the case that $n_s = 1000$ with different communication range R (from 10m to 2m) are reported in Table 10. Similar with Example 1 (see

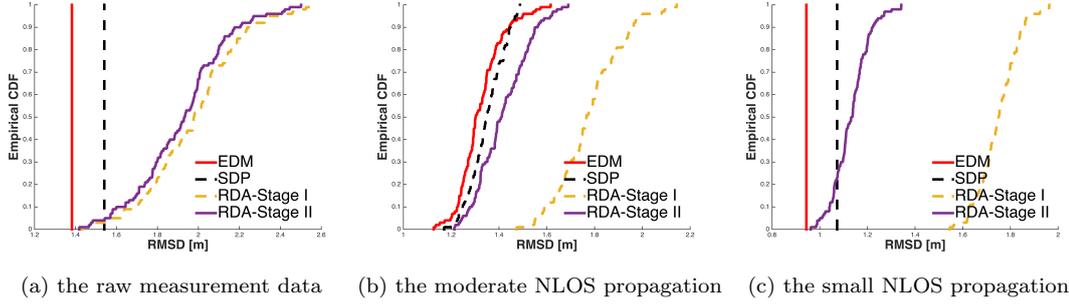


Fig. 7: CDF of different methods for Example 3 with different NLOS propagation levels.

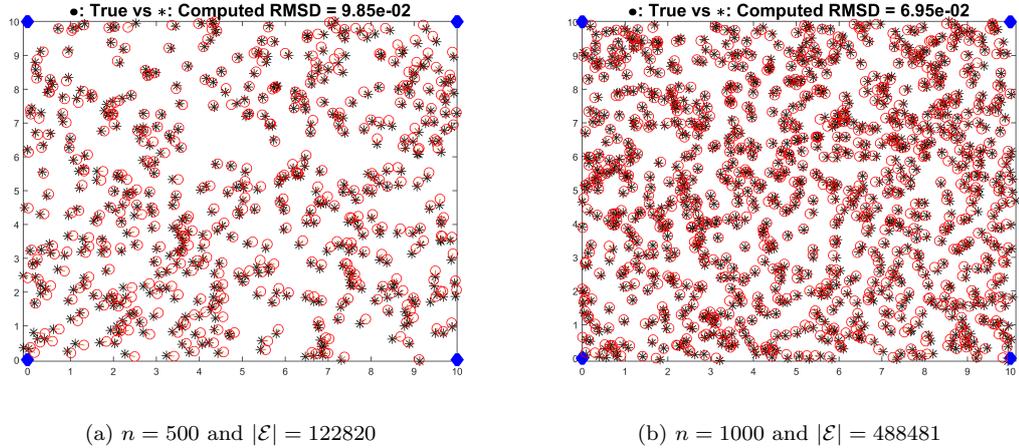


Fig. 8: Performance comparison for large scale networks with $P_{NLOS} = 5\%$ with the 4 anchors in blue at the 4 corners.

n_a	n_s	R	connectivity (%)	$ \mathcal{E} $	RMSD	cpu(s)
4	1000	10	48.5	488481	0.0695	91.41
		8	42.6	429820	0.0742	95.42
		6	31.2	314756	0.0862	100.89
		4	16.8	168813	0.1785	124.64
		2	5.3	53078	0.8046	196.27

Table 10: Performance of EDM for large scale networks with different communication range R (in meters).

Table 4), the average RMSD of EDM increases from 0.0695m to 0.8046m, while the average node connectivity decreases from 48.5% to 5.3%.

To conclude our numerical section, we would like to emphasize the three reasons which may have contributed to the strong performance of our method. The first is that our model is robust in handling NLOS links in the sense that it seems less sensitive to the number of NLOS links. For example, our EDM model can still produce a reasonably good solution even when the probability of NLOS links grows to 95% in the test problem T1 (see Table 3). The second reason is that our model does not rely on the graph rigidity theory for the underlying networks. It merely uses all

the distances information to find the best networks, which yield the least error from the observed distances, See Remark 1 at the end of Section II for our model being a convex relaxation of the LTS. The third reason is that the stopping criterion used in our implementation is the true KKT condition of the problem. However, we noticed that other methods used other stopping criteria that may lead to terminations at undesirable points. Without using the KKT condition (or its equivalence) as stopping criterion, one would never know whether the constrained optimization problem is truly solved or not.

6 Conclusions

This paper aimed to tackle a challenging problem in source localization where the distance measurements may contain biases of NLOS-type. The least-square formulation of such problem is equivalent to the well-known least-trimmed squares and hence is extremely difficult to solve. Convex relaxation seems to be a sensible approach as demonstrated by the existing SDP relaxations, which often have a large number of conic constraints.

With the awareness of the shortcomings of the SDP approach, we proposed a new convex optimization relaxation for the problem and our approach relies on the concept of EDM. In particular, we employ the almost positive semidefinite cone \mathcal{K}_+^n . This results in the convex optimization model (18), whose dual problem has a nice structure. We apply the recently developed 3-block ADMM to the dual problem and tested it on some real as well as simulated data. The numerical comparison with several popular methods including the SDP demonstrates the efficiency of our method and the localization quality is also very satisfactory even for some large-scale networks. We hope that the strong performance of our model and its algorithm will motivate other researchers to apply the EDM tool to various localization problems with large measurement errors.

One commonly used technique, which is also included in the real data [27], is to remove a common bias (a constant distance) from all NLOS measurements. This involves two issues here. One is to detect the NLOS links and the other is to accurately estimate the common bias. This remains a challenge for all existing models and thus will be our future research regarding the EDM optimization models. Another interesting topic is to see whether the EDM tool can be used in a recently investigated robust convex approximation for the localization under NLOS conditions by Wang, So and Li [46], where only one unknown source is studied.

Finally we note that algorithms in the form of ADMM can often be put in a distributed fashion so as for them to be applied to problems of very large size [7]. However, for our ADMM algorithm, the bottleneck lies with the solution of (25) where it is hard to distribute the calculation of the matrix projection. This issue is certainly worth exploring in future.

Acknowledgement. We would like to thank the two referees for their constructive comments that have helped to improve the quality of the paper.

References

1. A. Abramo, F. Blanchini, L. Geretti, and C. Savorgnan, A mixed convex/nonconvex distributed localization approach for the deployment of indoor positioning services, *IEEE Trans. on Mobile Comput.*, 7 (2008), pp. 1325-1337.
2. A. Y. Alfakih, A. Khandani, and H. Wolkowicz, Solving Euclidean distance matrix completion problems via semidefinite programming, *Comput Optim Appl*, 12 (1999), pp. 13-30.
3. A. Beck, P. Stoica, and J. Li, Exact and approximate solutions of source localization problems, *IEEE Tran. Sign. Proces.*, 56 (2008), pp. 1770-1778.
4. P. Biswas, T.-C. Lian, T.-C. Wang, and Y. Ye, Semidefinite programming based algorithms for sensor network localization, *ACM Transactions on Sensor Networks (TOSN)*, 2 (2006), pp. 188-220.
5. P. Biswas and Y. Ye, Semidefinite programming for ad hoc wireless sensor network localization, *Information Processing in Sensor Networks*, 2004, pp. 46-54.

6. I. Borg and P.J.F. Groenen, *Modern Multidimensional Scaling: Theory and Applications* (2nd ed.) Springer Series in Statistics, Springer, 2005.
7. S. Boyd, N. Parikh, E. Chu, B. Peleato and J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers, *Foundations and Trends in Machine Learning*, 3 (2010), pp. 1-122.
8. H. Chen, G. Wang, Z. Wang, H. C. So, and H. V. Poor, Non-Line-of-Sight node localization based on semi-definite programming in wireless sensor networks, *IEEE Trans. Wireless Commun.*, 11 (2012), pp. 108-116.
9. T.F. Cox and M.A.A. Cox, *Multidimensional Scaling*, 2nd Ed, Chapman and Hall/CRC, 2001.
10. J. Dattorro, *Convex Optimization & Euclidean Distance Geometry*. Meboo Publishing USA 2005.
11. P.A. Forero and G.B. Giannakis, Sparsity-exploiting robust multidimensional scaling. *IEEE Tran. Signal Processing*, 60 (2012), 4118-4134.
12. N. Gaffke and R. Mathar, A cyclic projection algorithm via duality, *Metrika*, 36 (1989), pp. 29-54.
13. W. Glunt, T. L. Hayden, S. Hong, and J. Wells, An alternating projection algorithm for computing the nearest Euclidean distance matrix, *SIAM J. Matrix Anal. Appl.*, 11 (1990), pp. 589-600.
14. W. Glunt, T.L. Hayden, and R. Raydan, Molecular conformations from distance matrices, *J. Comput. Chem.*, 14 (1993), pp. 114-120.
15. J. Gouveia and T.K. Pong, Comparing SOS and SDP relaxations of sensor network localization, *Comput. Optim. Appl.*, 52 (2012), pp. 609627.
16. M. Grant and S. Boyd, *CVX: Matlab software for disciplined convex programming*, Version 2.1, <http://cvxr.com/cvx>, January 2015.
17. I. Guvenc, C.-C. Chong, F. Watanabe, and H. Inamura, NLOS identification and weighted least-squares localization for UWB systems using multipath channel statistics, *EURASIP J Adv Signal Process*, volume 2008, pp. 271984-14.
18. I. Guvenc and C.-C. Chong, A survey on TOA based wireless localization and NLOS mitigation techniques, *IEEE Commun. Surv. Tutorials*, 11 (2009), pp. 107-124.
19. T. L. Hayden and J. Wells, Approximation by matrices positive semidefinite on a subspace, *Linear Algebra and its Applications*, 109 (1988), pp. 115-130.
20. P. J. Huber, Robust estimation of a location parameter, *The Annals of Mathematical Statistics*, 35 (1964), pp. 73-101.
21. P. J. Huber and E. M. Ronchetti, *Robust Statistics*. John Wiley & Sons, 2011.
22. T. Jia and R. M. Buehrer, Collaborative position location with NLOS mitigation, *Personal, Indoor and Mobile Radio Communications Workshops (PIMRC Workshops)*, 2010, pp. 267-271.
23. S.M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Upper Saddle River, NJ: Prentice-Hall, 1993.
24. N. Krislock and H. Wolkowicz, Explicit sensor network localization using semidefinite representations and facial reductions, *SIAM J. Optim.* 20 (2010), pp. 2679-2708.
25. Z.-S. Lu and Y. Zhang, Penalty decomposition methods for l_0 -norm minimization, Technical Report, http://www.optimization-online.org/DB_FILE/2010/08/2719.pdf, (2010).
26. J.W. Nie, Sum of squares method for sensor network localization, *Comput. Optim. Appl.*, 43 (2009), pp. 151-179.
27. N. Patwari, A. O. Hero, M. Perkins, N. S. Correal, and R. J. O'Dea, Relative location estimation in wireless sensor networks, *IEEE Tran. Signal Processing*, 51 (2003), pp. 2137-2148.
28. N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero, R. L. Moses, and N. S. Correal, Locating the nodes: cooperative localization in wireless sensor networks, *IEEE Signal Processing Magazine*, 22 (2005), pp. 5469.
29. T.K. Pong, Edge-based semidefinite programming relaxation of sensor network localization with lower bound constraints, *Comput. Optim. Appl.*, 53 (2012), pp. 2344.
30. H.-D. Qi, A semismooth Newton method for the nearest Euclidean distance matrix problem, *SIAM J. Matrix Anal. Appl.*, 34 (2013), pp. 67-93.
31. H.-D. Qi and X. Yuan, Computing the nearest Euclidean distance matrix with low embedding dimensions, *Mathematical programming*, 147 (2014), pp. 351-389.
32. H.-D. Qi, N.H. Xiu, and X.M. Yuan, A Lagrangian dual approach to the single source localization problem, *IEEE Tran. Signal Processing*, 61 (2013), 3815-3826.
33. J. Riba and A. Urruela, A non-line-of-sight mitigation technique based on ML-detection, *ICASSP*, 2 (2004), pp. 153-156.
34. R. Rockafellar, *Convex Analysis*. Princeton University Press, 1970.
35. P.J. Rousseeuw and A.M. Leroy, *Robust regression and outlier detection*. John Wiley & Sons, Inc., New York, Hoboken, NJ, USA (1987).
36. I.J. Schoenberg, Remarks to Maurice Fréchet's article "Sur la définition axiomatique d'une classe d'espace distanciés vectoriellement applicable sur l'espace de Hilbert". *Ann. of Math. (2)*. 36 (1935), pp. 724-732.
37. P. Stoica and J. Li, Source localization from range-difference measurements, *IEEE Signal Processing Mag.*, 23 (2006), pp. 63-69.
38. D. Sun, K.-C. Toh, and L. Yang, A convergent proximal alternating direction method of multipliers for conic programming with 4-Block constraints. To appear in *SIAM J. Optimization*.

39. K.-C. Toh, An inexact primal-dual path following algorithm for convex quadratic SDP, *Math. Program.*, 112 (2008), pp. 221-254.
40. K.-C. Toh, M. J. Todd, and R. H. Tütüncü, SDPT3 A Matlab software package for semidefinite programming, Version 1.3, *Optimization Methods and Software*, 11 (1999), pp. 545-581.
41. P. Tseng, Secondorder cone programming relaxation of sensor network localization, *SIAM J. Optim.*, 18 (2007), pp. 1561-185.
42. R.M. Vaghefi and R.M. Buehrer, Cooperative sensor localization with NLOS mitigation using semidefinite programming, *WPNC (2012)*, pp. 13-18.
43. R.M. Vaghefi, J. Schloemann, and R.M. Buehrer, NLOS mitigation in TOA-based localization using semidefinite programming, *Positioning Navigation and Communication (WPNC)*, 2013, pp. 1-6.
44. S. Venkatesh and R. M. Buehrer, Non-line-of-sight identification in ultra-wideband systems based on received signal statistics, *IET Microw. Antennas Propag.*, 1 (2007), pp. 1120-11.
45. S. Venkatesh and R. M. Buehrer, NLOS mitigation using linear programming in ultrawideband location-aware networks, *IEEE Trans. Veh. Technol.*, 56 (2007), pp. 3182-3198.
46. G. Wang, A. M-C. So and Y. Li, Robust convex approximation methods for TDOA-based localization under NLOS conditions, Technical report, Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, 2014.
47. Z. Wang, S. Zheng, Y. Ye and S. Boyd, Further relaxations of the semidefinite programming approach to sensor network localization. *SIAM J. Optim.* 19 (2008), 655-673.
48. G. Young and A.S. Householder, Discussion of a set of points in terms of their mutual distances. *Psychometrika*. 3, 19-22 (1938).
49. S. Yousefi, X.-W. Chang, and B. Champagne, Distributed cooperative localization in wireless sensor networks without NLOS identification, *Positioning, Navigation and Communication (WPNC)*, March 2014, pp. 1-6.