

Robust Localization Using Range Measurements with Unknown and Bounded Errors

Xiufang Shi, *Member, IEEE*, Guoqiang Mao, *Senior Member, IEEE*,
 Brian D.O. Anderson, *Life Fellow, IEEE*, Zaiyue Yang, *Member, IEEE*,
 and Jiming Chen, *Senior Member, IEEE*

Abstract—Cooperative geolocation has attracted significant research interests in recent years. A large number of localization algorithms rely on the availability of statistical knowledge of measurement errors, which is often difficult to obtain in practice. Compared with the statistical knowledge of measurement errors, it can often be easier to obtain the measurement error bound. This work investigates a localization problem assuming unknown measurement error distribution except for a bound on the error. We first formulate this localization problem as an optimization problem to minimize the worst-case estimation error, which is shown to be a non-convex optimization problem. Then, relaxation is applied to transform it into a convex one. Furthermore, we propose a distributed algorithm to solve the problem, which will converge in a few iterations. Simulation results show that the proposed algorithms are more robust to large measurement errors than existing algorithms in the literature. Geometrical analysis providing additional insights is also provided.

Index Terms—Cooperative localization, bounded measurement error, worst-case estimation error, Chebyshev center, convex relaxation, semidefinite programming

I. INTRODUCTION

WIRELESS sensor networks (WSNs) play an important role in many applications [1]–[6], such as environmental monitoring, target tracking, energy harvesting, etc. Most of these applications are location-dependent, which require knowledge of the measurement locations, and consequently make localization one of the most important technologies in WSNs. In the last decade, wireless localization technologies have undergone significant developments. Existing localization techniques can be divided into many categories depending on the measurement techniques and the localization algorithms being used. Range-based localization, which uses inter-node range measurements for location estimation, is one of the most widely used localization approaches. The range measurements can be estimated from received signal strength (RSS) [7],

[8], time of arrival (TOA) [9], time difference of arrival (TDOA) [7], [10], and so on [7]. One can also use RSS, TOA, etc. measurements to directly estimate the sensors' positions without first converting these measurements into range measurements [11], [12].

For range-based sensor network localization, the localization problem is usually formulated as an optimization problem to determine sensors' positions, such that they are consistent with the inter-node range measurements. Various localization algorithms are developed to optimize some given objectives. One of the most widely used algorithms is maximum likelihood estimator (MLE), which maximizes the likelihood function of the unknown sensors' positions [13], [14]. MLE relies on statistical knowledge of measurement error. Another widely used algorithm is least squares estimator (LS) [15], [16], which minimizes the squared error between range measurements and range estimates. LS does not need the knowledge of measurement error distribution. When the measurement error follows a zero-mean Gaussian distribution, MLE and LS become the same. Both MLE and LS are non-Bayesian estimators, there are also some Bayesian estimators, e.g., minimum mean squared error (MMSE) estimator and maximum a posteriori (MAP) estimator, which treat the position vector as a random variable with a priori distribution, but the posterior distribution of the position vector is usually hard to describe [17].

Many position estimators, e.g., MLE, are based on the assumption that measurement error follows a specific distribution and its probability density function is known. In practice, the measurement error may deviate from the assumed distribution and vary according to the measurement environment, measurement technique and measurement device. And we often do not have statistical knowledge of measurement error. The performance of these algorithms becomes vulnerable to an inaccurate statistical knowledge of the measurement error. For those algorithms that do not need the statistical knowledge of measurement error, e.g. LS based algorithm, they may perform well when the measurement error is small; however, their performance can degrade significantly when the error becomes very large [18]. The error bound is another and less demanding way to describe the property of measurement error. The bounded error assumption has been widely applied in many areas, e.g., set-theoretic estimation in system and control area [19]–[21], wireless localization [22], [23], etc. Moreover, compared with the statistical distribution of measurement error, it is much easier to obtain the measurement error bound in many situations [24]. Furthermore, there are already techniques to

This work was partially supported by 973 Program under Grant 2015CB352503, National Program for Special Support of Top-Notch Young Professionals, NSFC under Grant 61371159, scholarship funded by China Scholarship Council, National ICT Australia Ltd., the Australian Research Council under Grant DP-110100538 and Grant DP-130103610. The corresponding author of this paper is Jiming Chen.

X. Shi, Z. Yang and J. Chen are with the State Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou, China. (e-mail: xfshi.zju@gmail.com, yangzy@zju.edu.cn, jmchen@ieee.org)

G. Mao is with School of Computing and Communication, University of Technology Sydney, Sydney, NSW 2007, Australia, and Data61-CSIRO, Sydney, NSW 2015, Australia. (e-mail: Guoqiang.Mao@uts.edu.au)

B. D. O. Anderson is with Research School of Engineering, Australian National University, Canberra, ACT 0200, Australia, and Data61-CSIRO, Canberra, ACT 2601, Australia. (e-mail: brian.anderson@anu.edu.au)

estimate the measurement error bound with small sets of data, [23], [24], e.g., using support vector algorithm to find the smallest sphere that the data live on.

Motivated by the above observations, in this paper, we design a localization algorithm that is robust against large measurement errors and does not need the statistical knowledge of measurement error; instead, only a measurement error bound is required. The main contributions of our work are summarized as follows:

- 1) We first design a centralized robust localization algorithm to minimize the worst-case estimation error, which only uses the measurement error bound. Using geometrical analysis, we show that the algorithm has bounded localization error.
- 2) A distributed and iterative localization algorithm is further developed. The convergence of the iterative estimation algorithm is proved. The communication cost and computational complexity of the proposed algorithm are analyzed.
- 3) Extensive simulations show that the proposed algorithms are robust against large measurement errors, and the distributed counterpart can converge in a few iterations.

The remainder of this paper is organized as follows. Section II introduces the related work. Section III gives the problem formulation. Section IV presents the proposed centralized localization algorithm. Section V illustrates a geometrical interpretation of our problem and analyzes the error bound of our algorithm. Section VI proposes a distributed algorithm. Simulation results are presented in Section VII. Conclusions are drawn in Section VIII.

Notations: Throughout the paper, matrices and vectors are denoted by upper boldface and lower boldface letters, respectively; the operator $(\cdot)^T$ denotes the transposition; $\text{Tr}(\cdot)$ denotes the matrix trace; $\|\cdot\|$ stands for the l_2 norm; $|\mathcal{N}|$ denotes the cardinality of \mathcal{N} .

II. RELATED WORK

Range-based sensor network localization seeks to estimate the unknown sensors' positions that are consistent with the inter-node range measurements and the known anchors' positions [25]–[30]. As one of the most widely used localization approaches, range-based localization has attracted substantial research interests.

In practice, the range measurements are usually not error-free. If the statistical knowledge of the measurement error is available *a priori*, MLE is statistically optimal since it maximizes the likelihood function of sensors' positions. However, the optimization problem based on MLE is often non-convex. Many existing methods, e.g., the gradient-based descent method for tackling an MLE problem, require good initialization to reach the global optimum; otherwise, they will fall into a local optimum. One alternative method that deals with the non-convexity of MLE is semidefinite relaxation (SDR), which relaxes the non-convex optimization problem into a convex optimization problem [26], [27]. In [26], an MLE with SDR is proposed for network localization with Gaussian distributed noise. Simonetto and Leus [27] derived

convex relaxation for MLE under different measurement error models.

If the statistical properties of the measurement error are unknown, the localization problem is usually formulated as an optimization problem minimizing some global cost function. To solve such a problem, a number of available algorithms have been proposed. Some of these algorithms are implemented in a centralized way. The semidefinite programming (SDP) based approach in [31] is one of the most widely used algorithms, and it seeks to minimize the error between the squared range measurements and the squared range estimates. Such an optimization problem is however non-convex. In order to obtain a solution, it is common to transform this problem into an SDP problem through SDR. To improve the computation efficiency of SDP in large networks, edge-based SDP (ESDP) and second-order cone programming (SOCP) relaxation approaches were further proposed respectively [29], [30]. Another widely used approach is to use a multidimensional scaling (MDS) algorithm [28], in which, the localization problem is posed as an LS problem. Subject to use of a good initialization, a gradient-based method [32] is also a quick way for position estimation. The above mentioned algorithms also have distributed versions, e.g., distributed SDP [26], distributed SOCP [33], [34], distributed MDS [35], [36], and distributed gradient based methods [26], [32], [35]. These algorithms are essentially minimizing the discrepancy between the range measurements and the range estimates. There are also some other distributed algorithms, e.g., [37], [38]. Khan *et al.* [37] proposed a distributed iterative localization algorithm based on the use of barycentric coordinates, which requires all the unknown sensors to lie in the convex hull of the anchors. Diao *et al.* [38] proposed a more general algorithm, which does not require each sensor to lie inside the convex hull of its neighbors and can guarantee global convergence. Note though that none of these algorithms directly minimizes the position estimation error. Meanwhile, theoretical analysis establishing the mathematical relationship between the above objective functions and the position estimation error is still lacking. Moreover, although these algorithms typically perform well when the measurement error is small, when the error becomes large, their performance cannot be guaranteed.

Regarding the direct minimization of estimation error, Eldar *et al.* [20] investigated a minimax estimator, which minimizes the worst-case estimation error, for parameter estimation in a classical linear regression model. In the problem of [20], the measurement model is linear with bounded error and the true parameter vector is assumed to lie in the intersection of some known ellipsoids. Furthermore, simulations in [20] show the advantage of the proposed minimax estimator over the constrained least squares (CLS), which minimizes the data error. Different from the linear regression problem in [20], in the localization problem, the range measurements are nonlinear functions of sensors' positions and the feasible set of the positions is not convex, which makes the localization problem more challenging. Inspired by the work in [20], we design a position estimator that minimizes the worst-case position estimation error for robust localization.

III. PROBLEM FORMULATION

We consider a static network in two-dimensional space, which consists of n sensors, denoted by $\mathcal{V}_x = \{1, \dots, n\}$, and m anchors, denoted by $\mathcal{V}_a = \{n+1, \dots, n+m\}$. The true position of sensor i is $\mathbf{x}_i = [x_i, y_i]$, $i \in \mathcal{V}_x$, which is unknown and needs to be estimated. Anchor k 's position is known as $\mathbf{a}_k = [a_{kx}, a_{ky}]$, $k \in \mathcal{V}_a$. Due to the communication limitation, a pair of nodes can acquire the range measurement between them only when they are within the sensing range. Let \mathcal{N}_x and \mathcal{N}_a respectively denote the set of sensor-sensor links and the set of sensor-anchor links. All the nodes $\mathcal{V} = \{\mathcal{V}_x, \mathcal{V}_a\}$ and the inter-node links $\mathcal{E} = \{\mathcal{N}_x, \mathcal{N}_a\}$ constitute an undirected graph $G = (\mathcal{V}, \mathcal{E})$. We assume this graph is connected. Furthermore, to guarantee that all the sensors can be uniquely localized, we assume this network is globally rigid¹, and there exist at least three non-collinear anchors in the area [18], [39].

The sensor-sensor range measurement is

$$z_{ij} = d_{ij} + v_{ij}, \quad (i, j) \in \mathcal{N}_x \quad (1)$$

where $d_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|$ is the true distance between sensor i and j , and $|v_{ij}| \leq \gamma$ is the unknown and bounded measurement error. Correspondingly, the sensor-anchor range measurement is

$$z_{ik} = d_{ik} + v_{ik}, \quad (i, k) \in \mathcal{N}_a \quad (2)$$

where $d_{ik} = \|\mathbf{x}_i - \mathbf{a}_k\|$ is the true distance between sensor i and anchor k , and $|v_{ik}| \leq \gamma$ is the unknown and bounded measurement error. We assume the error bound γ is known *a priori* and the measurement errors are independent of each other.

From the constraints on the measurements, sensor i must lie in the following closed set

$$\mathcal{C}_i = \{\mathbf{y}_i : \underline{d}_{ij} \leq \|\mathbf{y}_i - \mathbf{y}_j\| \leq \overline{d}_{ij}, \quad \forall (i, j) \in \mathcal{N}_x \quad (3)$$

$$\underline{d}_{ik} \leq \|\mathbf{y}_i - \mathbf{a}_k\| \leq \overline{d}_{ik}, \quad \forall (i, k) \in \mathcal{N}_a \quad (4)$$

where \mathbf{y}_i denotes an arbitrary point in the feasible set of \mathbf{x}_i , $\underline{d}_{ij} = z_{ij} - \gamma$, $\overline{d}_{ij} = z_{ij} + \gamma$, $\underline{d}_{ik} = z_{ik} - \gamma$ and $\overline{d}_{ik} = z_{ik} + \gamma$. It is possible that \underline{d}_{ij} or \underline{d}_{ik} becomes negative if γ is large or very loose. In such case, we set $\underline{d}_{ij} = 0$, $\underline{d}_{ik} = 0$.

Since the true positions of the sensors are unknown, we cannot minimize the position estimation error directly. Therefore, we *minimize the worst-case estimation error over all the feasible positions*. Let $\hat{\mathbf{x}}_i$ denote sensor i 's position estimate; the worst-case estimation error would be

$$\max_{\mathbf{y}_i} \sum_{i=1}^n \|\mathbf{y}_i - \hat{\mathbf{x}}_i\|^2$$

$$s.t. \quad \underline{d}_{ij} \leq \|\mathbf{y}_i - \mathbf{y}_j\| \leq \overline{d}_{ij}, \quad \forall (i, j) \in \mathcal{N}_x \quad (5a)$$

$$\underline{d}_{ik} \leq \|\mathbf{y}_i - \mathbf{a}_k\| \leq \overline{d}_{ik}, \quad \forall (i, k) \in \mathcal{N}_a \quad (5b)$$

¹A network \mathcal{F}_x at $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{n+m}]^T$ is *globally rigid* if every network \mathcal{F}_p that is equivalent to \mathcal{F}_x is also congruent to it [18], [40], where a) a network \mathcal{F}_x is defined by a graph $G = (\mathcal{V}, \mathcal{E})$ with a map $\Lambda : \mathcal{V} \rightarrow \mathbb{R}^2$, with sensor i 's position $\mathbf{x}_i \in \mathbb{R}^2$, $i \in \mathcal{V}$; b) Two networks \mathcal{F}_x and \mathcal{F}_p are equivalent if $\|\mathbf{x}_i - \mathbf{x}_j\| = \|\mathbf{p}_i - \mathbf{p}_j\|$ for any $\{i, j\} \in \mathcal{E}$; c) Two networks \mathcal{F}_x and \mathcal{F}_p are congruent if $\|\mathbf{x}_i - \mathbf{x}_j\| = \|\mathbf{p}_i - \mathbf{p}_j\|$, for all pairs $\{i, j\}$ not limited by $\{i, j\} \in \mathcal{E}$.

We must choose $\hat{\mathbf{x}}_i$ to minimize this worst-case error. Therefore, we seek to solve:

$$\min_{\hat{\mathbf{x}}_i} \max_{\mathbf{y}_i} \sum_{i=1}^n \|\mathbf{y}_i - \hat{\mathbf{x}}_i\|^2$$

s.t. (5a)(5b) (6)

To facilitate the notation and analysis, we write (6) in a compact matrix form. Let $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n]^T \in \mathbb{R}^{2n}$ be the true position vector, which is unknown and fixed. We also introduce a new position vector $\mathbf{y} = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n]^T \in \mathbb{R}^{2n}$, which denotes an arbitrary possible value of \mathbf{x} in the feasible set. Clearly, $\mathbf{x} \in \mathcal{C}$ below. Let $\hat{\mathbf{x}} = [\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \dots, \hat{\mathbf{x}}_n]^T \in \mathbb{R}^{2n}$ denote the estimate of \mathbf{x} . Then our localization problem can be expressed as

$$\min_{\hat{\mathbf{x}}} \max_{\mathbf{y} \in \mathcal{C}} \text{Tr}((\mathbf{y} - \hat{\mathbf{x}})(\mathbf{y} - \hat{\mathbf{x}})^T) \quad (7)$$

$$\mathcal{C} = \{\mathbf{y} : \underline{d}_{ij}^2 \leq f_{ij}(\mathbf{y}) \leq \overline{d}_{ij}^2, \quad \forall (i, j) \in \mathcal{N}_x, \\ \underline{d}_{ik}^2 \leq f_{ik}(\mathbf{y}) \leq \overline{d}_{ik}^2, \quad \forall (i, k) \in \mathcal{N}_a\} \quad (8)$$

where

$$f_{ij}(\mathbf{y}) = \mathbf{e}_{(2i-1)(2j-1)}^T \mathbf{y} \mathbf{y}^T \mathbf{e}_{(2i-1)(2j-1)} + \mathbf{e}_{(2i)(2j)}^T \mathbf{y} \mathbf{y}^T \mathbf{e}_{(2i)(2j)}$$

$$f_{ik}(\mathbf{y}) = \mathbf{a}_k \mathbf{a}_k^T - 2a_{kx} \mathbf{y}^T \mathbf{e}_{2i-1} - 2a_{ky} \mathbf{y}^T \mathbf{e}_{2i} + \mathbf{e}_{2i-1}^T \mathbf{y} \mathbf{y}^T \mathbf{e}_{2i-1} + \mathbf{e}_{2i}^T \mathbf{y} \mathbf{y}^T \mathbf{e}_{2i}$$

where $\mathbf{e}_i \in \mathbb{R}^{2n}$ is a column vector with 1 at the i th position and 0 elsewhere; $\mathbf{e}_{(i)(j)} \in \mathbb{R}^{2n}$ is a column vector with 1 at the i th position, -1 at the j th position, and 0 elsewhere. Three noncollinear anchors are needed to resolve translation and rotation ambiguities, but a single anchor is sufficient for establishing the boundedness of set \mathcal{C} .

IV. THE RELAXED ESTIMATION

Geometrically, problem (7) is the formulation for computing the Chebyshev center of set \mathcal{C} . The geometrical interpretation and analysis will be given in next section. Problem (7) is a non-convex optimization problem, for which, the convex optimization techniques cannot be directly used. In this section, we will propose a relaxed estimation algorithm. The main idea of our proposed algorithm is as follows: 1) Relax the non-convex optimization problem (7) into a convex optimization problem; 2) Change the order of optimization, which will further simplify the optimization problem; 3) Solve the corresponding Lagrangian dual problem of the simplified problem. In the following, we will introduce each step in detail.

A. Relaxation

Let $\Delta = \mathbf{y} \mathbf{y}^T$; then (7) can be rewritten as

$$\min_{\hat{\mathbf{x}}} \max_{(\mathbf{y}, \Delta) \in \mathcal{G}} \text{Tr}(\Delta - 2\hat{\mathbf{x}} \mathbf{y}^T + \hat{\mathbf{x}} \hat{\mathbf{x}}^T) \quad (9)$$

where \mathcal{G} is the constraint set:

$$\mathcal{G} = \{(\mathbf{y}, \Delta) : \underline{d}_{ij}^2 \leq g_{ij}(\Delta) \leq \overline{d}_{ij}^2, \quad \forall (i, j) \in \mathcal{N}_x, \\ \underline{d}_{ik}^2 \leq g_{ik}(\mathbf{y}, \Delta) \leq \overline{d}_{ik}^2, \quad \forall (i, k) \in \mathcal{N}_a \\ \Delta = \mathbf{y} \mathbf{y}^T\} \quad (10)$$

and

$$\begin{aligned} g_{ij}(\Delta) &= \mathbf{e}_{(2i-1)(2j-1)}^T \Delta \mathbf{e}_{(2i-1)(2j-1)} + \mathbf{e}_{(2i)(2j)}^T \Delta \mathbf{e}_{(2i)(2j)} \\ g_{ik}(\mathbf{y}, \Delta) &= \\ \mathbf{a}_k \mathbf{a}_k^T - 2a_{kx} \mathbf{y}^T \mathbf{e}_{2i-1} - 2a_{ky} \mathbf{y}^T \mathbf{e}_{2i} + \mathbf{e}_{2i-1}^T \Delta \mathbf{e}_{2i-1} + \mathbf{e}_{2i}^T \Delta \mathbf{e}_{2i} \end{aligned}$$

The equality constraint $\Delta = \mathbf{y}\mathbf{y}^T$ in (10) is not affine, which makes \mathcal{G} a non-convex set [41]. The optimization problem cannot be directly solved by convex optimization methods. As commonly done in the field [31], [42], we make the following relaxation:

$$\Delta \succeq \mathbf{y}\mathbf{y}^T \quad (11)$$

where the notation means that $\Delta - \mathbf{y}\mathbf{y}^T$ is a positive semidefinite matrix. To distinguish the relaxed Δ from the original one, we use Δ_r to denote the relaxed Δ . Then the relaxed constraint set, which is now convex, becomes

$$\mathcal{Q} = \{(\mathbf{y}, \Delta_r) : \underline{d}_{ij}^2 \leq g_{ij}(\Delta_r) \leq \overline{d}_{ij}^2, \forall (i, j) \in \mathcal{N}_x, \quad (12a)$$

$$\underline{d}_{ik}^2 \leq g_{ik}(\mathbf{y}, \Delta_r) \leq \overline{d}_{ik}^2, \forall (i, k) \in \mathcal{N}_a \quad (12b)$$

$$\Delta_r \succeq \mathbf{y}\mathbf{y}^T \} \quad (12c)$$

Geometrically, (12a) and (12b) constitute a convex polytope, which is a closed and bounded set. Moreover, the inequality $\Delta_r \succeq \mathbf{y}\mathbf{y}^T$, which is equivalent to $\begin{bmatrix} \Delta_r & \mathbf{y} \\ \mathbf{y}^T & 1 \end{bmatrix} \succeq 0$ [43], defines a positive semidefinite cone (closed but unbounded set) [41]. Set \mathcal{Q} is the intersection of a convex polytope and a positive semidefinite cone. Therefore, \mathcal{Q} is a closed and bounded set.

The relaxed problem becomes a convex optimization problem

$$\min_{\hat{\mathbf{x}}} \max_{(\mathbf{y}, \Delta_r) \in \mathcal{Q}} \text{Tr}(\Delta_r - 2\hat{\mathbf{x}}\mathbf{y}^T + \hat{\mathbf{x}}\hat{\mathbf{x}}^T) \quad (13)$$

B. Change of Optimization Order

In (13), the outer minimization part is an unconstrained optimization problem, which is straightforward, while the inner maximization part is a constrained optimization problem. To simplify the problem, we consider whether we can change the order of these two parts.

In our problem (13), the objective function is continuous, finite, and convex in $\hat{\mathbf{x}}$. Since the objective function is linear with \mathbf{y} and Δ_r , it is concave in (\mathbf{y}, Δ_r) . Both the feasible sets of $\hat{\mathbf{x}}$ and (\mathbf{y}, Δ_r) are closed. Moreover, set \mathcal{Q} is bounded. Consequently, according to Corollary 37.3.2 in [44], we can interchange the order of minimization and maximization. The equivalent optimization problem to (13) becomes

$$\max_{(\mathbf{y}, \Delta_r) \in \mathcal{Q}} \min_{\hat{\mathbf{x}}} \text{Tr}(\Delta_r - 2\hat{\mathbf{x}}\mathbf{y}^T + \hat{\mathbf{x}}\hat{\mathbf{x}}^T) \quad (14)$$

It is straightforward that the optimal solution of the inner minimization problem in (14) is $\hat{\mathbf{x}}(\mathbf{y}) = \mathbf{y}$. Hence, the equivalent optimization problem to (14) becomes

$$\max_{(\mathbf{y}, \Delta_r) \in \mathcal{Q}} \text{Tr}(\Delta_r - \mathbf{y}\mathbf{y}^T) \quad (15)$$

In our problem formulation, the problem of minimizing the worst case error is formulated as a min-max problem. That is, first assuming a given position estimate, since any point in the feasible region can be the true position of the sensor, we find the point in the feasible region that maximizes the difference

between the position estimate and the position of that point (which can be the potential true position). This represents the worst case position estimation error. Secondly, we find the position estimate that minimizes such worst case error. Alternatively, the min-max problem can also be formulated as a max-min problem. That is, assuming the ‘‘true’’ position is fixed (which can be any point in the feasible region), we first find the position estimate that minimizes the difference between the position estimate and the ‘‘true’’ position, i.e., the minimum position estimation error assuming the ‘‘true’’ position is fixed. We recognize that when the true position of the sensor is at a different point of the feasible region, the corresponding minimum position estimation error will be different and some points in the feasible region may deliver more accurate position estimates than some other points. This reflects the fact that other things being equal, some geometric points may be more accurately localized than some other points. Secondly, we find the true position within the feasible region that delivers the worst case minimum position estimation error. It can be shown analytically that the min-max and the max-min problem are equivalent. The maximization problem in (15) corresponds to the last step: i.e., finding the ‘‘true’’ sensor position that delivers the worst case minimum position estimation error.

C. Dual Problem

Since (15) is a convex optimization problem and strictly feasible, strong duality holds [41]. Problem (15) can be solved through its dual problem. The Lagrangian dual function of (15) is

$$\begin{aligned} L(\mathbf{y}, \Delta_r, \alpha_{ij}, \beta_{ij}, \omega_{ik}, \varphi_{ik}, \boldsymbol{\lambda}) &= \\ \text{Tr}(\mathbf{I}_{2n} + \boldsymbol{\lambda})(\Delta_r - \mathbf{y}\mathbf{y}^T) &+ \sum_{(i,j) \in \mathcal{N}_x} \alpha_{ij} (g_{ij}(\Delta_r) - \underline{d}_{ij}^2) + \\ \sum_{(i,j) \in \mathcal{N}_x} \beta_{ij} (-g_{ij}(\Delta_r) + \overline{d}_{ij}^2) &+ \sum_{(i,k) \in \mathcal{N}_a} \omega_{ik} (g_{ik}(\mathbf{y}, \Delta_r) - \underline{d}_{ik}^2) \\ + \sum_{(i,k) \in \mathcal{N}_a} \varphi_{ik} (-g_{ik}(\mathbf{y}, \Delta_r) + \overline{d}_{ik}^2) &\quad (16) \end{aligned}$$

where \mathbf{I}_{2n} denotes a $2n \times 2n$ identity matrix, and the dual variables are $\alpha_{ij} \in \mathbb{R}$, $\beta_{ij} \in \mathbb{R}$, $\omega_{ik} \in \mathbb{R}$, $\varphi_{ik} \in \mathbb{R}$ and $\boldsymbol{\lambda} \in \mathbb{R}^{2n \times 2n}$, which obey the constraints: $\alpha_{ij} \geq 0$, $\beta_{ij} \geq 0$, $\omega_{ik} \geq 0$, $\varphi_{ik} \geq 0$ and $\boldsymbol{\lambda} \succeq \mathbf{0}$. The dual problem is

$$\min_{(\alpha_{ij}, \beta_{ij}, \omega_{ik}, \varphi_{ik}, \boldsymbol{\lambda})} \sup_{(\mathbf{y}, \Delta_r)} L(\mathbf{y}, \Delta_r, \alpha_{ij}, \beta_{ij}, \omega_{ik}, \varphi_{ik}, \boldsymbol{\lambda}) \quad (17)$$

To simplify the notation, let L denote $L(\mathbf{y}, \Delta_r, \alpha_{ij}, \beta_{ij}, \omega_{ik}, \varphi_{ik}, \boldsymbol{\lambda})$. The inner maximization problem can be solved by letting the derivative of L with respect to \mathbf{y} and Δ_r equal to $\mathbf{0}$, i.e., $\frac{\partial L}{\partial \mathbf{y}} = \mathbf{0}$ and $\frac{\partial L}{\partial \Delta_r} = \mathbf{0}$. The optimal value of \mathbf{y} satisfies

$$\hat{\mathbf{y}} = -(\mathbf{I}_{2n} + \boldsymbol{\lambda})^{-1} \sum_{(i,k) \in \mathcal{N}_a} (\omega_{ik} - \varphi_{ik})(a_{kx} \mathbf{e}_{2i-1} + a_{ky} \mathbf{e}_{2i}) \quad (18)$$

and

$$\mathbf{I}_{2n} + \boldsymbol{\lambda} = - \sum_{(i,j) \in \mathcal{N}_x} (\alpha_{ij} - \beta_{ij}) \mathbf{E}_{ij} - \sum_{(i,k) \in \mathcal{N}_a} (\omega_{ik} - \varphi_{ik}) \mathbf{E}_i \quad (19)$$

where

$$\begin{aligned} \mathbf{E}_{ij} &= \mathbf{e}_{(2i-1)(2j-1)} \mathbf{e}_{(2i-1)(2j-1)}^T + \mathbf{e}_{(2i)(2j)} \mathbf{e}_{(2i)(2j)}^T \\ \mathbf{E}_i &= \mathbf{e}_{2i-1} \mathbf{e}_{2i-1}^T + \mathbf{e}_{2i} \mathbf{e}_{2i}^T \end{aligned}$$

By substituting (18) and (19) into (16), the dual function can be obtained as

$$g(\alpha_{ij}, \beta_{ij}, \omega_{ik}, \varphi_{ik}) = \mathbf{f}^T (\mathbf{I}_{2n} + \boldsymbol{\lambda})^{-1} \mathbf{f} + h(\alpha_{ij}, \beta_{ij}, \omega_{ik}, \varphi_{ik}) \quad (20)$$

where

$$\mathbf{f} = \sum_{(i,k) \in \mathcal{N}_a} (\omega_{ik} - \varphi_{ik}) (a_{kx} \mathbf{e}_{2i-1} + a_{ky} \mathbf{e}_{2i})$$

$$\begin{aligned} h(\alpha_{ij}, \beta_{ij}, \omega_{ik}, \varphi_{ik}) &= - \sum_{(i,j) \in \mathcal{N}_x} \alpha_{ij} \underline{d}_{ij}^2 + \sum_{(i,j) \in \mathcal{N}_x} \beta_{ij} \overline{d}_{ij}^2 \\ &+ \sum_{(i,k) \in \mathcal{N}_a} \omega_{ik} (\mathbf{a}_k \mathbf{a}_k^T - \underline{d}_{ik}^2) + \sum_{(i,k) \in \mathcal{N}_a} \varphi_{ik} (\overline{d}_{ik}^2 - \mathbf{a}_k \mathbf{a}_k^T) \end{aligned}$$

The dual optimization problem becomes

$$\min_{\alpha_{ij}, \beta_{ij}, \omega_{ik}, \varphi_{ik}, \boldsymbol{\lambda}} \mathbf{f}^T (\mathbf{I}_{2n} + \boldsymbol{\lambda})^{-1} \mathbf{f} + h(\alpha_{ij}, \beta_{ij}, \omega_{ik}, \varphi_{ik}) \quad (21a)$$

$$s.t. \quad \alpha_{ij}, \beta_{ij}, \omega_{ik}, \varphi_{ik} \geq 0, \quad \boldsymbol{\lambda} \succeq \mathbf{0}, \quad \forall (i,j) \in \mathcal{N}_x, \quad \forall (i,k) \in \mathcal{N}_a \quad (21b)$$

Assume that t is a scalar such that $\mathbf{f}^T (\mathbf{I}_{2n} + \boldsymbol{\lambda})^{-1} \mathbf{f} \leq t$. From the property of Schur complement, we have

$$\begin{bmatrix} \mathbf{I}_{2n} + \boldsymbol{\lambda} & \mathbf{f} \\ \mathbf{f}^T & t \end{bmatrix} \succeq 0 \quad (22)$$

Problem (21) can be transformed into an SDP problem

$$\begin{aligned} \min_{t, \alpha_{ij}, \beta_{ij}, \omega_{ik}, \varphi_{ik}, \boldsymbol{\lambda}} & t + h(\alpha_{ij}, \beta_{ij}, \omega_{ik}, \varphi_{ik}) \\ s.t. & (22)(21b) \end{aligned} \quad (23)$$

The above SDP can be numerically and efficiently solved to any arbitrary accuracy by many existing SDP solvers and toolboxes, e.g., SeDuMi [45], CVX [42], [46].

Let $\hat{\alpha}_{ij}, \hat{\beta}_{ij}, \hat{\omega}_{ik}, \hat{\varphi}_{ik}, \hat{\boldsymbol{\lambda}}$ denote the solution of (23); then the estimate of \mathbf{x} is

$$\mathbf{x}_{est} = -(\mathbf{I}_{2n} + \hat{\boldsymbol{\lambda}})^{-1} \sum_{(i,k) \in \mathcal{N}_a} (\hat{\omega}_{ik} - \hat{\varphi}_{ik}) (a_{kx} \mathbf{e}_{2i-1} + a_{ky} \mathbf{e}_{2i}) \quad (24)$$

In the above SDP, there are $2|\mathcal{N}_x| + 2|\mathcal{N}_a| + n(2n+1) + 1$ scalar variables to be optimized, and the number of scalar equality/inequality constraints is $2|\mathcal{N}_x| + 2|\mathcal{N}_a|$, the number of linear matrix inequality (LMI) constraints is 2, and the size of the LMI is at most $(2n+1) \times (2n+1)$. In (23), the computational complexity of SDP is $\mathcal{O}(n^6)$ [41] and the computational complexity of the matrix inverse operation is $\mathcal{O}(n^3)$. We can conclude that the computational complexity of our estimation algorithm is $\mathcal{O}(n^6)$.

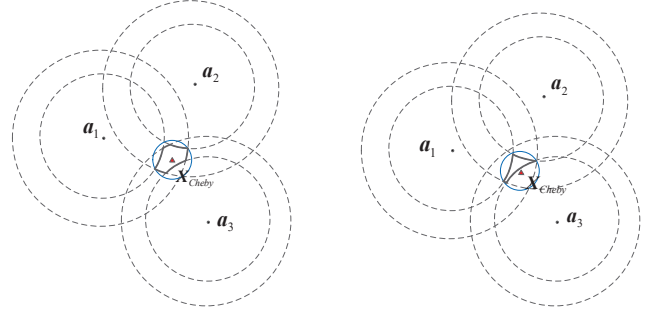
Remark 1

When the measurement errors are larger than the error bound, the constrained set is infeasible, which makes our optimization problem an infeasible problem. In such case, there are some existing techniques, e.g., using Farkas Lemma [47], which can detect such infeasible problem and find an approximated solution.

V. GEOMETRICAL INTERPRETATION

In this section, we will give a geometric interpretation of our problem. Firstly, we will show that the original problem (7) is a standard one, of finding the Chebyshev center of \mathcal{C} [20], [48], [49]. Secondly, we make a relaxation of the original problem, the solution of which may be thought of as a relaxed Chebyshev center.

A. Chebyshev Center



(a) \mathbf{x}_{Cheby} is feasible.

(b) \mathbf{x}_{Cheby} is infeasible.

Fig. 1. Chebyshev center of a non-convex set, which is the intersection of three feasible sets of a single sensor.

Geometrically, our objective is to find the Chebyshev center of \mathcal{C} , i.e., the center of the minimum ball enclosing \mathcal{C} [20]. The equivalent problem to (7) is

$$\min_{\hat{\mathbf{x}}} \left\{ R_c : \|\mathbf{y} - \hat{\mathbf{x}}\|^2 \leq R_c^2, \forall \mathbf{y} \in \mathcal{C} \right\} \quad (25)$$

In our problem, set \mathcal{C} , the feasible set of the position vector \mathbf{y} , is non-convex. This means that the Chebyshev center of \mathcal{C} may not lie in \mathcal{C} . We take a single sensor localization problem as an example to illustrate that. As shown in Fig. 1, $\mathbf{a}_1, \mathbf{a}_2$ and \mathbf{a}_3 are anchors. From their range measurements with bounded errors, we can determine a non-convex feasible set for the target sensor, which is the region surrounded by the solid bold curve. The minimum circle enclosing the feasible set is the smallest solid circle, and its center (the triangle) is the Chebyshev center \mathbf{x}_{Cheby} . In Fig. 1(a), \mathbf{x}_{Cheby} lies in the feasible set, but in Fig. 1(b), it is obvious that \mathbf{x}_{Cheby} does not lie in the feasible set. Even though we cannot be sure whether \mathbf{x}_{Cheby} lies in the feasible set, if we take \mathbf{x}_{Cheby} as the position estimate, we will get the minimum worst-case estimation error.

In practice, sometimes a feasible estimate is preferred over an estimate with the minimum estimation error bound when the latter is outside the feasible set. If the Chebyshev center is infeasible, as shown in Fig. 1(b), one general way to proceed is to take the projection of \mathbf{x}_{Cheby} onto the feasible set as the estimate. We then compare the estimation error bound obtained with Chebyshev center and that obtained with its projection onto the feasible set.

Proposition 1

Suppose $\mathcal{C} \subset \mathbb{R}^{2n}$ is a closed set, let $\mathbf{x}_{Cheby} \in \mathbb{R}^{2n}$ be the Chebyshev center of \mathcal{C} and suppose $\mathbf{x}_{Cheby} \notin \mathcal{C}$. Let $\mathbf{x}_p = \mathcal{P}_{\mathcal{C}}(\mathbf{x}_{Cheby}) = \arg \min_{\mathbf{y} \in \mathcal{C}} \|\mathbf{x}_{Cheby} - \mathbf{y}\|$, where

$\mathcal{P}_{\mathcal{C}}(\cdot)$ denotes the projection operator onto set \mathcal{C} [41]. Let $R_c = \max_{\mathbf{y} \in \mathcal{C}} \|\mathbf{x}_{Cheby} - \mathbf{y}\|$ and $R_p = \max_{\mathbf{y} \in \mathcal{C}} \|\mathbf{x}_p - \mathbf{y}\|$. Then

$$R_c \leq R_p \leq 2R_c \quad (26)$$

Proof: From the definition of Chebyshev center, we can easily obtain

$$\begin{aligned} R_c^2 &= \max_{\mathbf{y} \in \mathcal{C}} \|\mathbf{x}_{Cheby} - \mathbf{y}\|^2 = \min_{\hat{\mathbf{x}}} \max_{\mathbf{y} \in \mathcal{C}} \|\hat{\mathbf{x}} - \mathbf{y}\|^2 \\ &\leq \max_{\mathbf{y} \in \mathcal{C}} \|\mathbf{x}_p - \mathbf{y}\|^2 = R_p^2 \end{aligned}$$

Apparently, $R_c \leq R_p$. We also note that

$$\begin{aligned} \max_{\mathbf{y} \in \mathcal{C}} \|\mathbf{x}_p - \mathbf{y}\| &= \max_{\mathbf{y} \in \mathcal{C}} \|\mathbf{x}_p - \mathbf{x}_{Cheby} + \mathbf{x}_{Cheby} - \mathbf{y}\| \\ &\leq \max_{\mathbf{y} \in \mathcal{C}} (\|\mathbf{x}_p - \mathbf{x}_{Cheby}\| + \|\mathbf{x}_{Cheby} - \mathbf{y}\|) \\ &\leq \|\mathbf{x}_p - \mathbf{x}_{Cheby}\| + \max_{\mathbf{y} \in \mathcal{C}} \|\mathbf{x}_{Cheby} - \mathbf{y}\| \end{aligned}$$

Since $\mathbf{x}_p \in \mathcal{C}$, then $\|\mathbf{x}_p - \mathbf{x}_{Cheby}\| \leq \max_{\mathbf{y} \in \mathcal{C}} \|\mathbf{x}_{Cheby} - \mathbf{y}\|$. Thus we have

$$\max_{\mathbf{y} \in \mathcal{C}} \|\mathbf{x}_p - \mathbf{y}\| \leq 2 \max_{\mathbf{y} \in \mathcal{C}} \|\mathbf{x}_{Cheby} - \mathbf{y}\|$$

Equivalently, $R_p \leq 2R_c$. The proof is complete. \blacksquare

From Proposition 1, we can see that \mathbf{x}_p is feasible, but the upper bound of the estimation error of \mathbf{x}_p is larger than that of \mathbf{x}_{Cheby} . In practice, we cannot just say one estimate is definitely better than another; such a statement depends on the metric used to evaluate localization performance. If we need a feasible estimate with a small estimation error bound, \mathbf{x}_p would be a good choice. Nevertheless, if we need an estimate with the minimum estimation error bound, \mathbf{x}_{Cheby} is obviously a better choice. In this paper, we only take the estimation error bound as the metric evaluating localization performance, which makes \mathbf{x}_{Cheby} the best choice.

B. Relaxed Chebyshev Center

Finding the Chebyshev center is an NP-hard problem except in some special cases [48]. In our problem, because the constraint set is not convex, it is difficult to obtain \mathbf{x}_{Cheby} . In (24), \mathbf{x}_{est} is a relaxed estimate of the Chebyshev center. It is straightforward that the estimation error bound of \mathbf{x}_{est} is no less than that for \mathbf{x}_{Cheby} ,

$$\max_{\mathbf{y} \in \mathcal{C}} \|\mathbf{x}_{Cheby} - \mathbf{y}\|^2 = \min_{\hat{\mathbf{x}}} \max_{\mathbf{y} \in \mathcal{C}} \|\hat{\mathbf{x}} - \mathbf{y}\|^2 \leq \max_{\mathbf{y} \in \mathcal{C}} \|\mathbf{x}_{est} - \mathbf{y}\|^2$$

Even though we cannot be sure whether \mathbf{x}_{est} lies in the feasible set \mathcal{C} , when we minimize the worst-case estimation error over a relaxed constraint set, the estimation error satisfies

$$\begin{aligned} \|\mathbf{x}_{est} - \mathbf{x}\|^2 &\leq \max_{\mathbf{y} \in \mathcal{C}} \|\mathbf{x}_{est} - \mathbf{y}\|^2 \\ &= \max_{(\mathbf{y}, \Delta) \in \mathcal{G}} \text{Tr}(\Delta - 2\mathbf{x}_{est}\mathbf{y}^T + \mathbf{x}_{est}\mathbf{x}_{est}^T) \\ &\leq \max_{(\mathbf{y}, \Delta_r) \in \mathcal{Q}} \text{Tr}(\Delta_r - 2\mathbf{x}_{est}\mathbf{y}^T + \mathbf{x}_{est}\mathbf{x}_{est}^T) \\ &= \min_{\hat{\mathbf{x}}} \max_{(\mathbf{y}, \Delta_r) \in \mathcal{Q}} \text{Tr}(\Delta_r - 2\hat{\mathbf{x}}\mathbf{y}^T + \hat{\mathbf{x}}\hat{\mathbf{x}}^T) \quad (27) \end{aligned}$$

Then we can say the estimation error is upper bounded by the optimal value of (13).

In Euclidean space, every closed convex bounded set has a unique Chebyshev center [49]. However, our original constraint set \mathcal{C} is not convex, and the Chebyshev center \mathbf{x}_{Cheby} may not be unique. Through relaxation, the relaxed set \mathcal{Q} becomes a closed convex bounded set. Therefore, we can obtain a unique Chebyshev center \mathbf{x}_{est} of the relaxed set \mathcal{Q} .

VI. DISTRIBUTED ESTIMATION

In practice, the network scale is often very large, including hundreds, even thousands of sensors and sometimes only a small number of anchors. If the network is localized in a centralized way, it may result in an extremely high communication burden and computational complexity at the central processor. Therefore, in this section, we consider an implementation of the above relaxed estimation method in a distributed way. In the distributed algorithm, each sensor will first make an initial guess about its position, which is denoted by $\hat{\mathbf{x}}_i(0), i = 1, 2, \dots, n$. Then, each sensor takes its neighbor nodes as ‘anchors’, and iteratively estimates its position.

A. Initial Estimation

1) *Sensor-Anchor Distance Estimation:* Since the sensing range of each sensor is limited, not all the sensors have direct connections with anchors. To make an initial estimation on each sensor’s position, we first estimate the Euclidean distance between each sensor and anchor through information exchange between neighboring nodes.

Similarly to the DV-hop localization algorithm [50], each anchor broadcasts its position with a hop-counter initialized to one through the network. Let n_{ik} denote the hop-count value from anchor k to sensor i . Sensor i would compute the lower and the upper bounds of $\|\mathbf{x}_i - \mathbf{a}_k\|$, and broadcast these bounds along with \mathbf{a}_k and n_{ik} to its neighbors. The procedure of computing the lower and the upper bounds is explained in next paragraph. Each receiving sensor will select the information along the shortest path, measured by the number of hops, to compute the bounds of distance between this sensor and anchor k .

If $(i, k) \in \mathcal{N}_a$, i.e., sensor i has a direct connection with anchor k , i.e., $n_{ik} = 1$, the true distance between i and anchor k satisfies (4). If $(i, k) \notin \mathcal{N}_a$, i.e., sensor i does not have a direct connection with anchor k , $n_{ik} > 1$. For $j \in \mathcal{N}_i$, if $j = \arg \min_{l \in \mathcal{N}_i} n_{lk}$, $n_{ik} = n_{jk} + 1$ and sensor i would use the information from sensor j to estimate \underline{d}_{ik} and \overline{d}_{ik} . The true distance between i and j satisfies (3). Obviously,

$$\begin{aligned} \|\mathbf{x}_i - \mathbf{x}_j\| + \|\mathbf{x}_j - \mathbf{a}_k\| &\leq \overline{d}_{ij} + \overline{d}_{jk} \\ \|\mathbf{x}_i - \mathbf{x}_j\| - \|\mathbf{x}_j - \mathbf{a}_k\| &\geq \underline{d}_{ij} - \underline{d}_{jk} \\ \|\mathbf{x}_j - \mathbf{a}_k\| - \|\mathbf{x}_i - \mathbf{x}_j\| &\geq \underline{d}_{jk} - \overline{d}_{ij} \end{aligned}$$

Since $\|\|\mathbf{x}_i - \mathbf{x}_j\| - \|\mathbf{x}_j - \mathbf{a}_k\|\| \leq \|\mathbf{x}_i - \mathbf{x}_j + \mathbf{x}_j - \mathbf{a}_k\| \leq \|\mathbf{x}_i - \mathbf{x}_j\| + \|\mathbf{x}_j - \mathbf{a}_k\|$, we have $\underline{d}_{ik} = \max\{\underline{d}_{ij} - \underline{d}_{jk}, \underline{d}_{jk} - \overline{d}_{ij}, 0\}$, $\overline{d}_{ik} = \overline{d}_{ij} + \overline{d}_{jk}$ and $n_{ik} = n_{jk} + 1$.

By recursion, it can be shown that, the distance bounds between sensor i and anchor k are

$$\underline{d}_{ik} = \begin{cases} \max\{z_{ik} - \gamma, 0\}, & (i, k) \in \mathcal{N}_a \\ \max\{\underline{d}_{ij} - \underline{d}_{jk}, \underline{d}_{jk} - \overline{d}_{ij}, 0\}, & (i, k) \notin \mathcal{N}_a \end{cases} \quad (28)$$

$$\overline{d_{ik}} = \begin{cases} z_{ik} + \gamma, & (i, k) \in \mathcal{N}_a \\ \frac{z_{ij} + \gamma}{d_{ij} + d_{jk}}, & (i, k) \notin \mathcal{N}_a, \end{cases} \quad (29)$$

where $j = \arg \min_{l \in \mathcal{N}_i} n_{lk}$.

Remark 2

If the network is very large, the initial estimates of sensor-anchor distance bounds are obtained through an n_k -hop path, where n_k might be very large. Consequently, the estimation performance will be degraded. In this case, some approximation strategies can be applied to reduce n_k by dividing such large-scale network into smaller subnetworks. Many existing techniques are available for network division, e.g., the approach applied in [26], clustering in [51], clique extraction in [52], etc.

2) *Initial Position Estimation:* The initial estimate of sensor i 's position is obtained by solving the following optimization problem

$$\begin{aligned} & \min_{\hat{\mathbf{x}}_i} \max_{\mathbf{y}_i} \|\mathbf{y}_i - \hat{\mathbf{x}}_i\|^2 \\ \text{s.t.} \quad & \underline{d_{ik}}^2 \leq \|\mathbf{y}_i - \mathbf{a}_k\|^2 \leq \overline{d_{ik}}^2, \quad \forall k \in \mathcal{V}_a \end{aligned} \quad (30)$$

Let $\Delta_i = \mathbf{y}_i \mathbf{y}_i^T$; an equivalent problem to (30) is

$$\begin{aligned} & \min_{\hat{\mathbf{x}}_i} \max_{(\mathbf{y}_i, \Delta_i) \in \mathcal{G}_i} \{\Delta_i - 2\mathbf{y}_i \hat{\mathbf{x}}_i^T + \hat{\mathbf{x}}_i \hat{\mathbf{x}}_i^T\} \\ \mathcal{G}_i = & \{(\mathbf{y}_i, \Delta_i) : \underline{d_{ik}}^2 \leq \Delta_i - 2\mathbf{y}_i \mathbf{a}_k^T + \mathbf{a}_k \mathbf{a}_k^T \leq \overline{d_{ik}}^2, \\ & \Delta_i = \mathbf{y}_i \mathbf{y}_i^T, \quad \forall k \in \mathcal{V}_a\} \end{aligned} \quad (31)$$

The constraint set \mathcal{G}_i is non-convex, as before, we relax $\Delta_i = \mathbf{y}_i \mathbf{y}_i^T$ into $\Delta_i \geq \mathbf{y}_i \mathbf{y}_i^T$. Let Δ_{ir} denote the relaxed Δ_i ; then problem (31) becomes a convex optimization problem

$$\begin{aligned} & \min_{\hat{\mathbf{x}}_i} \max_{(\mathbf{y}_i, \Delta_{ir}) \in \mathcal{Q}_i} \{\Delta_{ir} - 2\mathbf{y}_i \hat{\mathbf{x}}_i^T + \hat{\mathbf{x}}_i \hat{\mathbf{x}}_i^T\} \\ \mathcal{Q}_i = & \{(\mathbf{y}_i, \Delta_{ir}) : \underline{d_{ik}}^2 \leq \Delta_{ir} - 2\mathbf{y}_i \mathbf{a}_k^T + \mathbf{a}_k \mathbf{a}_k^T \leq \overline{d_{ik}}^2, \\ & \Delta_{ir} \geq \mathbf{y}_i \mathbf{y}_i^T, \quad \forall k \in \mathcal{V}_a\} \end{aligned} \quad (32)$$

Similarly to the centralized estimation, we change the order of optimization and the solution of the minimization part is $\hat{\mathbf{x}}_i = \mathbf{y}_i$. Problem (32) is simplified as

$$\max_{(\mathbf{y}_i, \Delta_{ir}) \in \mathcal{Q}_i} \{\Delta_{ir} - \mathbf{y}_i \mathbf{y}_i^T\} \quad (33)$$

Its Lagrangian dual problem is also an SDP problem:

$$\begin{aligned} & \min_{\omega_k, \varphi_k} t + \sum_{k \in \mathcal{V}_a} \omega_k (\mathbf{a}_k \mathbf{a}_k^T - \underline{d_{ik}}^2) + \sum_{k \in \mathcal{V}_a} \varphi_k (\overline{d_{ik}}^2 - \mathbf{a}_k \mathbf{a}_k^T) \\ \text{s.t.} \quad & \begin{bmatrix} -\sum_{k \in \mathcal{V}_a} (\omega_k - \varphi_k) I_2 & \sum_{k \in \mathcal{V}_a} (\omega_k - \varphi_k) \mathbf{a}_k^T \\ \sum_{k \in \mathcal{V}_a} (\omega_k - \varphi_k) \mathbf{a}_k & t \end{bmatrix} \succeq 0 \\ & -\sum_{k \in \mathcal{V}_a} (\omega_k - \varphi_k) \geq 1 \\ & \omega_k, \varphi_k \geq 0, \quad k \in \mathcal{V}_a \end{aligned} \quad (34)$$

where ω_k and φ_k are the dual variables. By setting the derivative of the Lagrangian function with respect to \mathbf{y}_i equal to 0, the estimate of \mathbf{y}_i becomes a function of ω_k and φ_k ,

whose optimal values $\hat{\omega}_k$ and $\hat{\varphi}_k$ are obtained by solving (34). Then, the initial position estimate of sensor i is

$$\hat{\mathbf{x}}_i(0) = \frac{\sum_{k \in \mathcal{V}_a} (\hat{\omega}_k - \hat{\varphi}_k) \mathbf{a}_k}{\sum_{k \in \mathcal{V}_a} (\hat{\omega}_k - \hat{\varphi}_k)} \quad (35)$$

From (27),

$$\|\mathbf{x}_i - \hat{\mathbf{x}}_i(0)\|^2 \leq \min_{\hat{\mathbf{x}}_i} \max_{(\mathbf{y}_i, \Delta_{ir}) \in \mathcal{Q}_i} \{\Delta_{ir} - 2\mathbf{y}_i \hat{\mathbf{x}}_i^T + \hat{\mathbf{x}}_i \hat{\mathbf{x}}_i^T\} = R_i^2(0)$$

Since strong duality holds, the value of $R_i^2(0)$ equals to the optimal value of (34).

B. Iterative Estimation

After initial estimation, in the following iterative estimation algorithm, each sensor takes its neighbor nodes as ‘anchors’, and iteratively estimates its position utilizing its neighbor nodes’ position estimates and the corresponding range measurements. Suppose sensor i 's position estimate at the τ -th iteration is $\hat{\mathbf{x}}_i(\tau)$, and $\|\mathbf{x}_i - \hat{\mathbf{x}}_i(\tau)\|^2 \leq R_i(\tau)^2$. Then the updated position at the $(\tau + 1)$ -th iteration can be obtained by solving the following optimization problem

$$\min_{\hat{\mathbf{x}}_i} \max_{\mathbf{y}_i \in \mathcal{C}_i(\tau)} \|\mathbf{y}_i - \hat{\mathbf{x}}_i\|^2 \quad (36)$$

where \mathbf{y}_i denotes the feasible value of \mathbf{x}_i in $\mathcal{C}_i(\tau)$ and

$$\begin{aligned} \mathcal{C}_i(\tau) = & \{\mathbf{y}_i : \|\mathbf{y}_i - \hat{\mathbf{x}}_i(\tau)\|^2 \leq R_i(\tau)^2 \\ & \underline{d_{ij}}^2 \leq \|\mathbf{y}_i - \hat{\mathbf{x}}_j(\tau)\|^2 \leq \overline{d_{ij}}^2, \quad \forall j \in \mathcal{N}_i\} \end{aligned} \quad (37)$$

Let $\Delta_i = \mathbf{y}_i \mathbf{y}_i^T$; problem (36) can be rewritten as

$$\begin{aligned} & \min_{\hat{\mathbf{x}}_i} \max_{(\mathbf{y}_i, \Delta_i) \in \mathcal{G}_i(\tau)} \{\Delta_i - 2\mathbf{y}_i \hat{\mathbf{x}}_i^T + \hat{\mathbf{x}}_i \hat{\mathbf{x}}_i^T\} \\ \mathcal{G}_i(\tau) = & \{(\mathbf{y}_i, \Delta_i) : \|\mathbf{y}_i - \hat{\mathbf{x}}_i(\tau)\|^2 \leq R_i(\tau)^2 \end{aligned} \quad (38)$$

$$\underline{d_{ij}}^2 \leq g_{ij}(\tau) \leq \overline{d_{ij}}^2, \quad \forall j \in \mathcal{N}_i \quad (39)$$

$$\Delta_i = \mathbf{y}_i \mathbf{y}_i^T \quad (40)$$

where $g_{ij}(\tau) = \Delta_i - 2\mathbf{y}_i \hat{\mathbf{x}}_j(\tau)^T + \hat{\mathbf{x}}_j(\tau) \hat{\mathbf{x}}_j(\tau)^T$.

Through a similar process to that used in equations (32),(33),(34), i.e., relaxation, change of optimization order, and dual problem transformation, the position update of sensor i at the $(\tau + 1)$ -th iteration becomes

$$\hat{\mathbf{x}}_i(\tau + 1) = \frac{\hat{\alpha}(\tau + 1) \hat{\mathbf{x}}_i(\tau) + \sum_{j \in \mathcal{N}_i} (\hat{\phi}_j(\tau + 1) - \hat{\psi}_j(\tau + 1)) \hat{\mathbf{x}}_j(\tau)}{\hat{\alpha}(\tau + 1) + \sum_{j \in \mathcal{N}_i} (\hat{\phi}_j(\tau + 1) - \hat{\psi}_j(\tau + 1))} \quad (41)$$

where $\hat{\alpha}(\tau + 1)$, $\hat{\phi}_j(\tau + 1)$ and $\hat{\psi}_j(\tau + 1)$ are the optimal values of α , ϕ_j and ψ_j in the following SDP problem:

$$\begin{aligned} & \min_{t_i, \alpha, \phi_j, \psi_j} f_i(t_i, \alpha, \phi_j, \psi_j, \tau + 1) \\ \text{s.t.} \quad & \begin{bmatrix} (\alpha - \sum_{j \in \mathcal{N}_i} (\phi_j - \psi_j)) I_2 & (\alpha - \sum_{j \in \mathcal{N}_i} (\phi_j - \psi_j)) \hat{\mathbf{x}}_i(\tau)^T \\ (\alpha - \sum_{j \in \mathcal{N}_i} (\phi_j - \psi_j)) \hat{\mathbf{x}}_i(\tau) & t_i \end{bmatrix} \succeq 0, \\ & \alpha - \sum_{j \in \mathcal{N}_i} (\phi_j - \psi_j) \geq 1, \\ & \alpha, \phi_j, \psi_j \geq 0, \quad j \in \mathcal{N}_i \end{aligned} \quad (42)$$

where

$$f_i(t_i, \alpha, \phi_j, \psi_j, \tau + 1) = t_i + \alpha(R_i(\tau)^2 - \hat{\mathbf{x}}_i(\tau)\hat{\mathbf{x}}_i(\tau)^T) + \sum_{j \in \mathcal{N}_i} \phi_j(\hat{\mathbf{x}}_j(\tau)\hat{\mathbf{x}}_j(\tau)^T - \underline{d}_{ij}^2) + \sum_{j \in \mathcal{N}_i} \psi_j(\overline{d}_{ij}^2 - \hat{\mathbf{x}}_j(\tau)\hat{\mathbf{x}}_j(\tau)^T)$$

and α is the dual variable associated with the inequality (38) in the Lagrange function, ϕ_j and ψ_j are the dual variables associated with the inequalities (39) in the Lagrange function.

Let $R_i^2(\tau + 1)$ denote the upper bound of the squared position estimation error of sensor i at the $(\tau + 1)$ -th iteration, i.e.,

$$\| \mathbf{x}_i - \hat{\mathbf{x}}_i(\tau + 1) \|^2 \leq R_i^2(\tau + 1) \quad (43)$$

where $R_i^2(\tau + 1)$ equals to the optimal value of $f_i(t_i, \alpha, \phi_j, \psi_j, \tau + 1)$. If $\| R_i^2(\tau + 1) - R_i^2(\tau) \| \leq \epsilon$, where ϵ is a very small constant, we regard the position estimate of sensor i as having converged to a steady state, and mark it as ‘localized’. The localization of sensor i will be terminated. The network localization will be terminated when all the sensors are ‘localized’. The procedures of the distributed estimation including initial estimation are illustrated in Algorithm 1.

Algorithm 1 : Distributed Algorithm

- 1: **for** $i = 1$ to n **do**
 - 2: **Initial distance bound estimation:**
 - 3: Compute \underline{d}_{ik} and \overline{d}_{ik} , $\forall k \in \mathcal{V}_a$, as (28) and (29).
 - 4: **Initial position estimation:**
 - 5: Compute $\hat{\mathbf{x}}_i(0)$ using (35).
 - 6: **end for**
 - 7: **Iterative position estimation:**
 - 8: **repeat**
 - 9: **for all** $i \in \mathcal{V}_x$ [in parallel] **do**
 - 10: Update sensor i 's position $\hat{\mathbf{x}}_i(\tau)$ as (41).
 - 11: **if** $\| R_i^2(\tau + 1) - R_i^2(\tau) \| \leq \epsilon$ **then**
 - 12: Mark sensor i as ‘localized’.
 - 13: **end if**
 - 14: **end for**
 - 15: $\tau = \tau + 1$
 - 16: **until** All sensors are ‘localized’
-

C. Convergence Analysis

A key convergence property of Algorithm 1 is provided by the following theorem.

Theorem 1

Let $\{\hat{\mathbf{x}}_i(\tau)\}_{\tau=0}^\infty, i \in \mathcal{V}_x$ be the sequence of sensor i 's position estimates generated by Algorithm 1 and let the corresponding upper bounds of the squared position estimation errors be $\{R_i^2(\tau)\}_{\tau=0}^\infty$. Then for every τ , we have

$$R_i^2(\tau + 1) \leq R_i^2(\tau) \quad (44)$$

and

$$\lim_{\tau \rightarrow \infty} (R_i^2(\tau + 1) - R_i^2(\tau)) = 0 \quad (45)$$

and

$$\lim_{\tau \rightarrow \infty} \| \hat{\mathbf{x}}_i(\tau + 1) - \hat{\mathbf{x}}_i(\tau) \| = 0 \quad (46)$$

Proof: For sensor i , at the $(\tau + 1)$ -th iteration, we have

$$R_i^2(\tau + 1) = \min_{\hat{\mathbf{x}}_i} \max_{(\mathbf{y}_i, \Delta_i) \in \mathcal{Q}_i(\tau)} \{ \Delta_i - 2\mathbf{y}_i\hat{\mathbf{x}}_i^T + \hat{\mathbf{x}}_i\hat{\mathbf{x}}_i^T \} \quad (47)$$

where

$$\mathcal{Q}_i(\tau) = \{ (\mathbf{y}_i, \Delta_i) : \Delta_i - 2\mathbf{y}_i\hat{\mathbf{x}}_i(\tau)^T + \hat{\mathbf{x}}_i(\tau)\hat{\mathbf{x}}_i(\tau)^T \leq R_i(\tau)^2 \\ \underline{d}_{ij}^2 \leq g_{ij}(\tau) \leq \overline{d}_{ij}^2, \forall j \in \mathcal{N}_i \\ \Delta_i \geq \mathbf{y}_i\mathbf{y}_i^T \}$$

By changing the order of optimization and solving the minimization part, in a similar way with (33), the maximization part becomes

$$R_i^2(\tau + 1) = \max_{(\mathbf{y}_i, \Delta_i) \in \mathcal{Q}_i(\tau)} (\Delta_i - \mathbf{y}_i\mathbf{y}_i^T) \quad (48)$$

The optimal value of \mathbf{y}_i is $\hat{\mathbf{x}}_i(\tau + 1)$ in (41). Then, we have

$$R_i^2(\tau + 1) = \max_{\Delta_i \in \mathcal{S}_i(\tau)} (\Delta_i - \hat{\mathbf{x}}_i(\tau + 1)\hat{\mathbf{x}}_i(\tau + 1)^T) \quad (49)$$

where

$$\mathcal{S}_i(\tau) = \{ \Delta_i : \Delta_i - 2\hat{\mathbf{x}}_i(\tau + 1)\hat{\mathbf{x}}_i(\tau)^T + \hat{\mathbf{x}}_i(\tau)\hat{\mathbf{x}}_i(\tau)^T \leq R_i(\tau)^2 \\ \underline{d}_{ij}^2 \leq \Delta_i - 2\hat{\mathbf{x}}_i(\tau + 1)\hat{\mathbf{x}}_j(\tau)^T + \hat{\mathbf{x}}_j(\tau)\hat{\mathbf{x}}_j(\tau)^T \leq \overline{d}_{ij}^2 \\ \Delta_i \geq \hat{\mathbf{x}}_i(\tau + 1)\hat{\mathbf{x}}_i(\tau + 1)^T, \forall j \in \mathcal{N}_i \}$$

Let $u_i(\tau) = R_i(\tau)^2 + 2\hat{\mathbf{x}}_i(\tau + 1)\hat{\mathbf{x}}_i(\tau)^T - \hat{\mathbf{x}}_i(\tau)\hat{\mathbf{x}}_i(\tau)^T$ and $u_{ij}(\tau) = \overline{d}_{ij}^2 + 2\hat{\mathbf{x}}_i(\tau + 1)\hat{\mathbf{x}}_j(\tau)^T - \hat{\mathbf{x}}_j(\tau)\hat{\mathbf{x}}_j(\tau)^T$; then the value of $R_i^2(\tau + 1)$ is

$$R_i^2(\tau + 1) = \min \{ u_i(\tau), u_{ij}(\tau), \forall j \in \mathcal{N}_i \} \\ - \hat{\mathbf{x}}_i(\tau + 1)\hat{\mathbf{x}}_i(\tau + 1)^T \\ \leq u_i(\tau) - \hat{\mathbf{x}}_i(\tau + 1)\hat{\mathbf{x}}_i(\tau + 1)^T \\ = R_i^2(\tau) - \| \hat{\mathbf{x}}_i(\tau + 1) - \hat{\mathbf{x}}_i(\tau) \|^2 \quad (50)$$

Straightforwardly, we can easily obtain (44). By recursion, we have

$$R_i^2(\tau + 1) \leq R_i^2(\tau) - \| \hat{\mathbf{x}}_i(\tau + 1) - \hat{\mathbf{x}}_i(\tau) \|^2 \leq \dots \\ \leq R_i^2(0) - \sum_{l=0}^{\tau} \| \hat{\mathbf{x}}_i(l + 1) - \hat{\mathbf{x}}_i(l) \|^2$$

Since $R_i^2(\tau + 1) \geq 0$, we have $\sum_{l=0}^{\infty} \| \hat{\mathbf{x}}_i(l + 1) - \hat{\mathbf{x}}_i(l) \|^2 \leq R_i^2(0)$, which means an infinite sum of non-negative values is bounded. Therefore, (46) must hold.

From (50), $\forall j \in \mathcal{N}_i$, if $\lim_{\tau \rightarrow \infty} u_{ij}(\tau) \leq \lim_{\tau \rightarrow \infty} u_i(\tau)$,

$$\lim_{\tau \rightarrow \infty} R_i^2(\tau + 1) = \lim_{\tau \rightarrow \infty} \min_{j \in \mathcal{N}_i} u_{ij}(\tau + 1) - \hat{\mathbf{x}}_i(\tau + 1)\hat{\mathbf{x}}_i(\tau + 1)^T \\ = \lim_{\tau \rightarrow \infty} \min_{j \in \mathcal{N}_i} (\overline{d}_{ij}^2 - \| \hat{\mathbf{x}}_i(\tau + 1) - \hat{\mathbf{x}}_j(\tau) \|^2) \quad (51)$$

If $\lim_{\tau \rightarrow \infty} u_{ij}(\tau) > \lim_{\tau \rightarrow \infty} u_i(\tau)$,

$$\lim_{\tau \rightarrow \infty} R_i^2(\tau + 1) = \lim_{\tau \rightarrow \infty} u_i(\tau) - \hat{\mathbf{x}}_i(\tau + 1)\hat{\mathbf{x}}_i(\tau + 1)^T \\ = \lim_{\tau \rightarrow \infty} R_i^2(\tau) - \| \hat{\mathbf{x}}_i(\tau + 1) - \hat{\mathbf{x}}_i(\tau) \|^2 \quad (52)$$

Since $\lim_{\tau \rightarrow \infty} \mathbf{x}_i(\tau + 1) = \lim_{\tau \rightarrow \infty} \mathbf{x}_i(\tau), \forall i \in \mathcal{V}_x$, from (51) and (52), we always have $\lim_{\tau \rightarrow \infty} R_i^2(\tau + 1) = \lim_{\tau \rightarrow \infty} R_i^2(\tau)$. Thus (45) holds. ■

Theorem 1 shows that upper bounds of the estimation errors generated by Algorithm 1, i.e., $\{R_i^2(\tau)\}_{\tau=0}^\infty, \forall i \in \mathcal{V}_x$, are non-increasing positive sequences. When $\tau \rightarrow \infty$, $R_i^2(\tau)$ will converge to a fixed value, not necessarily 0, and $\mathbf{x}_i(\tau)$ satisfies $\lim_{\tau \rightarrow \infty} \| \mathbf{x}_i(\tau) - \mathbf{x}_i \|^2 \leq \lim_{\tau \rightarrow \infty} R_i^2(\tau)$. The converged value of $R_i^2(\tau)$ is determined by the network configuration and measurement errors.

D. Computational Complexity and Communication Cost

In the initial estimation, a sensor only needs to broadcast the necessary information once. The information that sensor i broadcasts to its neighbors is $\{d_{ik}, \bar{d}_{ik}, n_{ik}\}, \forall k \in \mathcal{N}_a$. Therefore, the communication cost for sensor i is $3m |\mathcal{N}_{ij}|$. The computational cost comes from the SDP problem in (34), in which there are $2m + 1$ scalar variables to be optimized. The computational complexity of initial estimation is $\mathcal{O}(m^3)$.

In the iterative estimation, at each iteration, for sensor i , the computational cost comes from the SDP problem in (42), in which there are $2|\mathcal{N}_i| + 2$ scalar variables to be optimized, and the number of scalar inequality constraints is $2|\mathcal{N}_i| + 2$, the number of LMI constraints is 1, the size of the LMI is at most 3×3 . Therefore, the computational complexity of (42) is $\mathcal{O}(|\mathcal{N}_i|^3)$ [41]. The communication cost at each iteration for sensor i comes from the information exchange. Sensor i needs to send its position estimate at each iteration to its neighbor nodes. For 2-D localization, the communication cost for sensor i is proportional to $2|\mathcal{N}_i|$.

VII. SIMULATION RESULTS

In this section, we conduct extensive simulations to illustrate the performance of the relaxed estimation in Section IV, denoted by MinMax-SDP, and the distributed algorithm in Section VI, denoted by Dis-MinMax.

A. Performance of Centralized Algorithm

To investigate the performance of our proposed centralized algorithm, we first compare MinMax-SDP with three centralized algorithms when the measurement errors follow two well-known distributions, i.e., uniform distribution and Gaussian distribution [27]; second, we consider a mixture error model with Gaussian distributed errors as inliers and uniformly distributed errors as outliers; third, we test the performance of MinMax-SDP using the experimental data.

Our simulations are conducted in a unit square area. In this area, 50 sensors with unknown positions are randomly deployed using a uniform distribution, along with four anchors whose positions are known as $(-0.3, -0.3)$, $(0.3, -0.3)$, $(-0.3, 0.3)$ and $(0.3, 0.3)$. To guarantee the unique localizability of this network, we set the sensing range of each node as $R = 0.5$.

1) *Measurement error follows uniform or Gaussian distribution:* In this part we compare MinMax-SDP with three centralized algorithms: a) SDP in [31], which minimizes the norm of the error between the squared range measurements and the squared range estimates; b) ESDP in [29], which involves a further relaxation of SDP, and is of less computational complexity than SDP; c) E-ML in [27], which is an MLE with ESDP relaxation.

In order to show the robustness to measurement error distributions, we consider two scenarios: 1) the measurement errors are uniformly distributed; 2) the measurement errors are Gaussian distributed, which can be considered approximately bounded. That is, the actual measurement errors are determined in accord with these statistics, but our algorithm

runs under the assumption that the errors are limited by a fixed known bound.

Fig. 2 shows the localization results when the measurement errors are uniformly distributed in $[-\gamma, \gamma]$ and $\gamma = 0.02$. We can see that the localization errors of the four algorithms are all very small. The root mean squared error (RMSE) of MinMax-SDP is slightly higher than SDP, but smaller than ESDP and E-ML. However, when γ increases to 0.1, from Fig. 3, all of SDP, ESDP, and E-ML collapse, because the projection of high-dimension solution onto a lower dimension results in points getting ‘crowded’ together when measurement errors become large [31]. MinMax-SDP can still work with increased localization error. The reason behind this phenomenon is that when the measurement errors are large, our proposed algorithm can guarantee that the estimation error is bounded as shown in (27). However, SDP, ESDP and E-ML are known to collapse when the measurement errors become large [18]. When we set a larger measurement error bound, the worst-case measurement error would be larger and the probability for getting larger measurement errors increases. Therefore, our proposed algorithm outperforms SDP, ESDP and E-ML when the measurement error bound becomes large.

To compare the statistical performance of these algorithms, we conduct 50 Monte Carlo trials. Fig. 4 shows the RMSEs of these four algorithms under different measurement error bounds, from which, we can see, MinMax-SDP performs the best, and the advantage is much more obvious when the measurement error bound becomes larger. When the measurement errors follow Gaussian distribution, with zero-mean and variance as σ^2 , the measurement error bound utilized in MinMax-SDP is taken as $\gamma = 3\sigma$. The only information about the measurement errors for MinMax-SDP, SDP and ESDP is the bound, while E-ML has full knowledge of the measurement error distribution. From the RMSEs shown in Fig. 5, we can see that the performance of MinMax-SDP is comparable with that of E-ML, and much better than that of SDP and ESDP.

By comparing Fig. 5 with Fig. 4, we find that, MinMax-SDP performs better than E-ML when the measurement errors are uniformly distributed, whereas it performs worse than E-ML when the measurement errors are Gaussian distributed. At the first sight, this result appears to be counter-intuitive because E-ML, which has accurate knowledge of the measurement error distribution, should deliver better performance than the proposed scheme in both cases. We offer the following explanation for the observed result. When the measurement errors are uniformly distributed, all positions within the feasible region \mathcal{C} occur with equal probability. An ML-based estimator cannot differentiate these positions and therefore may return any position within this feasible region as the position estimate. In comparison, our algorithm will only deliver the best position estimate that minimizes the worst-case estimation error, i.e. the one resembling the Chebyshev center. Therefore when the measurement errors are uniformly distributed, the proposed algorithm delivers better performance than an ML-based estimator. When the measurement errors are Gaussian distributed, all positions within the feasible region occur with different probabilities. In this situation, the accurate

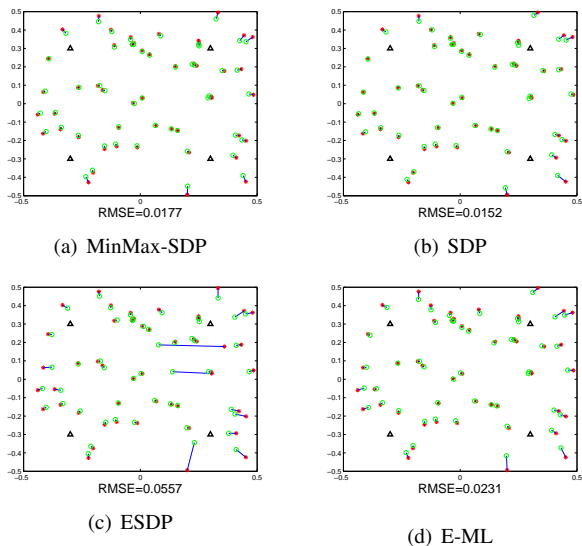


Fig. 2. Localization results under uniformly distributed measurement errors, where the triangles denote the anchors, the stars denote the true sensors' positions and circles denote the estimated sensors' positions. Measurement error bound $\gamma = 0.02$.

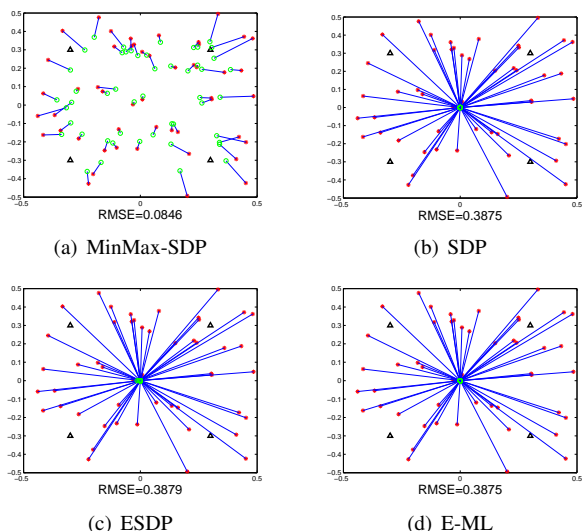


Fig. 3. Localization results under uniformly distributed measurement errors, where the triangles denote the anchors, the stars denote the true sensors' positions and circles denote the estimated sensors' positions. Measurement error bound $\gamma = 0.1$.

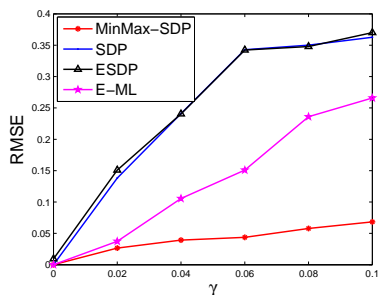


Fig. 4. RMSEs under uniformly distributed measurement errors with different bounds in 50 trials.

knowledge of the measurement error distribution, which forms the basis of E-ML estimator, can be exploited to deliver better performance than the proposed scheme, which does not rely

on such knowledge.

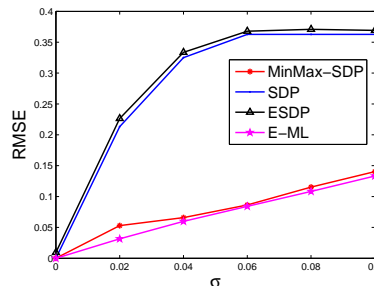


Fig. 5. RMSEs under Gaussian distributed measurement errors with different standard deviations in 50 trials.

2) *Mixture measurement error model*: In practice, the measurement errors may not perfectly fit a statistical distribution, among which, usually there exist some outliers. In the simulation, the inliers follow a zero-mean Gaussian distribution with standard deviation σ , and the outliers are uniformly distributed in $[-3\sigma, 3\sigma]$. Fig. 6 compares the performance of MinMax-SDP algorithm with E-ML. In MinMax-SDP, the error bound $\gamma = 3\sigma$. In E-ML, the measurement errors are taken as Gaussian distributed with zero-mean and standard deviation σ . The ratio denotes the ratio between the number of uniformly distributed errors and the number of Gaussian distributed errors. We can see, when the number of the outliers is small, i.e., *ratio* = 0.1 in our figure, E-ML is better than MinMax-SDP when $\sigma < 0.08$, but worse than MinMax-SDP when $\sigma > 0.08$. When the ratio increases to 0.5, we can see the performance of MinMax-SDP is slightly changed, but the performance of E-ML is much worse than that when *ratio* = 0.1. We can conclude that MinMax-SDP is more robust to the changes of the measurement error distributions than E-ML.

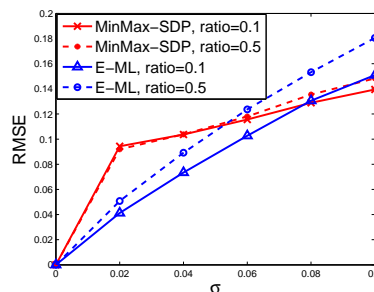


Fig. 6. Localization results with mixture measurement error model, i.e., Gaussian distributed errors as inliers and uniformly distributed errors as outliers.

3) *Experimental evaluation*: In this part, we use experimental data provided in [13] to test the performance of MinMax-SDP. In the experiment, there are four devices placed near the corners in the area and 40 devices to be localized. The measurements utilized for localization are TOA measurements, of which, the measurement errors are zero-mean Gaussian distributed with standard deviation $\sigma_T = 6.1ns$. Since the signal transmission speed v is known, the range measurements can be easily obtained. In MinMax-SDP, the range measurement error bound is $3\sigma_T v$. We assume that

the sensing range of each device is $R = 5m$, and build up the corresponding connectivity matrix. Only the measurements between the connected devices are used for localization. The localization results of MinMax-SDP and E-ML are shown in Fig. 7(a) and Fig. 7(b) respectively. Fig. 7(c) compares the CDF of the position estimation errors of MinMax-SDP and E-ML. We can see the performance of MinMax-SDP, which only uses the measurement error bound, is comparable with that of E-ML, which takes advantage of the statistical distribution of the measurement errors.

B. Performance of Distributed Algorithm

In the simulations about the performance of Dis-MinMax, we let four anchors be placed at $(-0.5, -0.5)$, $(0.5, -0.5)$, $(-0.5, 0.5)$, $(0.5, 0.5)$, and the sensors with unknown positions be randomly deployed using a uniform distribution in a unit square area. Three scenarios are considered: a) $n = 50$ and $R = 0.5$; b) $n = 100$ and $R = 0.3$; c) $n = 200$ and $R = 0.2$. The measurement errors follow a zero-mean Gaussian distribution, with standard deviation as $\sigma = 0.02$. The error bound is set as $\gamma = 3\sigma$. Fig. 8(a) compares the RMSEs of the position estimates obtained by MinMax-SDP and Dis-MinMax. We can see the RMSEs obtained by Dis-MinMax converge very quickly and the converged value is slightly smaller than that obtained by the centralized algorithm. Though MinMax-SDP and Dis-MinMax are both minimizing the worst-case estimation error, they are tackled in different ways, which results in the difference between the converged value obtained by the distributed algorithm and the RMSEs obtained by the centralized algorithm. Fig. 8(a) also illustrates the iteration process of the upper bound for the RMSE, which is defined as $\sqrt{\frac{\sum_{i=1}^n R_i(\tau)^2}{n}}$. From Fig. 8(a), we can see that the upper bound for the RMSE also quickly converges. The localization results when the number of sensors increases to 100 and 200 are shown in Fig.8(b) and Fig.8(c), from which, we can observe similar performance with that shown in Fig. 8(a). Therefore, Dis-MinMax is scalable to a larger number of nodes.

We then compare the localization performance of Dis-MinMax with other existing distributed algorithms: a) E-ML with ADMM in [27], which is a distributed implementation of the aforementioned E-ML and can converge with a sublinear rate; b) ECHO in [38], which uses barycentric coordinates to express the positions in a linear form and can converge to the true positions in error-free case; c) SNLRS in [51], which divides the entire network into several overlapping subnetworks, in which the sensors are localized via SDP, respectively. The global coordinates of sensor position estimates in each subnetwork are obtained through rigid registration. SNLRS is a variant of the distributed SDP. Table I compares the computational complexities and communication costs **per iteration** of different distributed algorithms. In this table, $|\mathcal{N}_i|$ denotes the number of sensor i 's neighboring nodes, $|\mathcal{N}_{ij}|$ denotes the number of sensors among sensor i 's neighboring nodes, and $|\mathcal{N}_{ia}|$ denotes the number of anchors among sensor i 's neighboring nodes. We can find the computational complexity of ECHO is smaller than that of Dis-MinMax and

E-ML with ADMM. The computational complexities of Dis-MinMax and E-ML with ADMM are of the same order. The communication costs of Dis-MinMax and ECHO are equal and smaller than that of E-ML with ADMM. It should be noted that in SNLRS, the sensors in each subnetwork are actually locally localized in a centralized way. With regard to SNLRS, table I only shows the computational complexities and communication costs **per iteration** during the position refinement using gradient-based search after the global rigid registration. Fig. 9 compares the localization results of these distributed algorithms. In Fig. 9, both Dis-MinMax and E-ML with ADMM can converge very fast, SNLRS converges in more than 100 iterations and ECHO converges in more than 10^5 iterations. Moreover, the converged value of RMSE obtained by Dis-MinMax is much lower than those obtained by other three algorithms. We can conclude that the performance of Dis-MinMax is the best among these four algorithms.

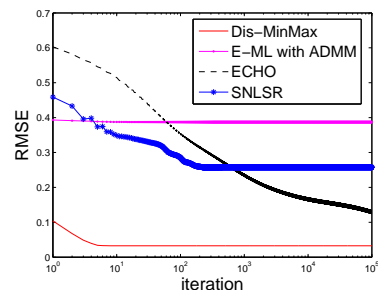


Fig. 9. Comparison of RMSEs obtained by Dis-MinMax, E-ML with ADMM, ECHO and SNLRS under Gaussian distributed measurement errors, where $\sigma = 0.02$, $n = 100$ and $R = 0.3$.

VIII. CONCLUSIONS

In this paper, we investigate a network localization problem with unknown and bounded measurement errors. We formulate this problem as a non-convex optimization problem to minimize the worst-case localization error. Through relaxation, we transform the non-convex optimization problem into a convex optimization problem, whose dual problem can be solved through semidefinite programming. We give a geometrical interpretation of our problem and prove that the localization error of our proposed algorithm is upper bounded. Furthermore, we propose a distributed algorithm, along with an initial estimation algorithm. The convergence of the distributed algorithm is also proved. Extensive simulations show that both the centralized algorithm and the distributed algorithm can perform very well without the statistical knowledge of measurement errors.

In this paper, we only consider the localization problem with bounded range measurements. One interesting extension of this work would be studying the localization problem with other forms of measurements, e.g., AOA, RSS, TDOA, etc., or hybrid measurements with unknown and bounded measurement errors.

REFERENCES

- [1] X. Cao, J. Chen, Y. Zhang, and Y. Sun, "Development of an integrated wireless sensor network micro-environmental monitoring system," *ISA trans.*, vol. 47, no. 3, pp. 247–255, 2008.

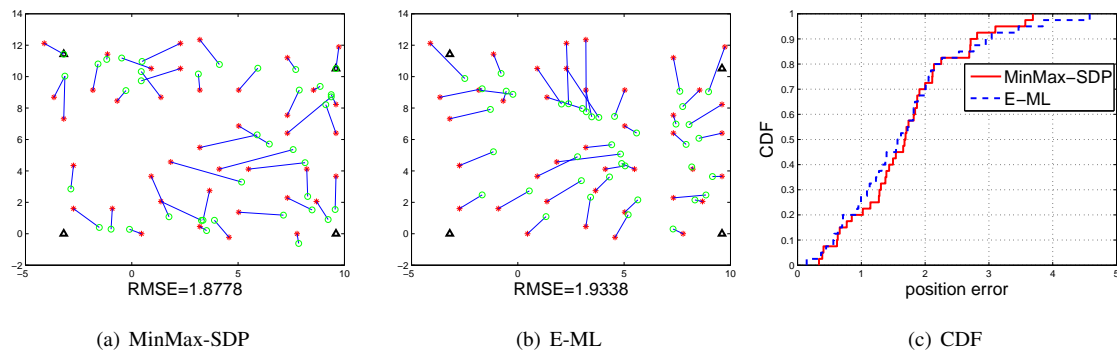


Fig. 7. Experimental localization results under Gaussian distributed measurement errors, where the triangles denote the anchors, the stars denote the true positions of the sensors and the circles denote the estimated positions.

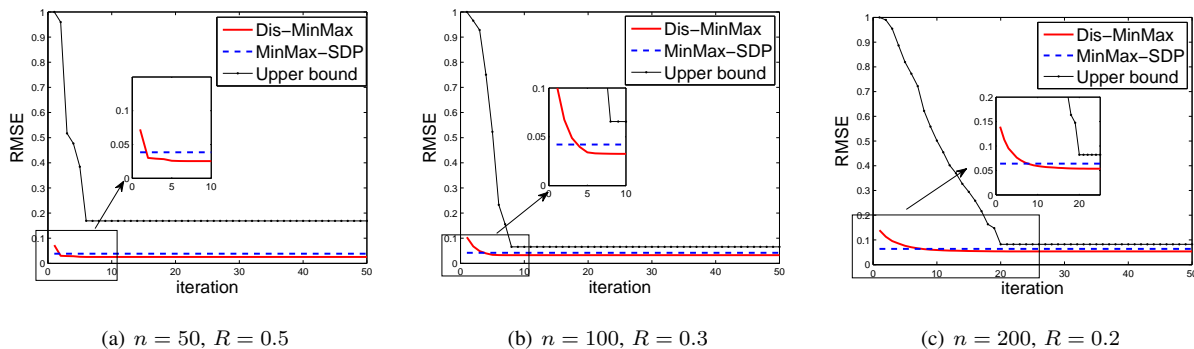


Fig. 8. Localization performance of Dis-MinMax under Gaussian distributed measurement errors

TABLE I
ANALYTICAL COMPARISON OF DIFFERENT DISTRIBUTED ALGORITHMS

	Dis-MinMax	E-ML with ADMM	ECHO	SNLRS
Size of convex problem	$2 \mathcal{N}_i + 2$	$7 \mathcal{N}_{ij} + 2 \mathcal{N}_{ia} + 3$	not applicable	not applicable
Computational complexity	$\mathcal{O}(\mathcal{N}_i ^3)$	$\mathcal{O}(\mathcal{N}_{ij} ^3)$	$\mathcal{O}(\mathcal{N}_i)$	$\mathcal{O}(\mathcal{N}_i)$
Communication cost	$2 \mathcal{N}_{ij} $	$9 \mathcal{N}_{ij} $	$2 \mathcal{N}_{ij} $	$ \mathcal{N}_{ij} $

- [2] R. Mao and G. Mao, "Road traffic density estimation in vehicular networks," in *Proc. IEEE WCNC*, 2013, pp. 4653–4658.
- [3] Y. Zhang, S. He, J. Chen, Y. Sun, and X. S. Shen, "Distributed sampling rate control for rechargeable sensor nodes with limited battery capacity," *IEEE Trans. Wireless Commun.*, vol. 12, no. 6, pp. 3096–3106, 2013.
- [4] Z. Yang, X. Shi, and J. Chen, "Optimal coordination of mobile sensors for target tracking under additive and multiplicative noises," *IEEE Trans. Ind. Electron.*, vol. 61, no. 7, pp. 3459–3468, 2014.
- [5] G. Mao and B. D. Anderson, "Graph theoretic models and tools for the analysis of dynamic wireless multihop networks," in *Proc. IEEE WCNC*, 2009, pp. 1–6.
- [6] X. Ge, S. Tu, T. Han, Q. Li, and G. Mao, "Energy efficiency of small cell backhaul networks based on Gauss–Markov mobile models," *IET Networks*, vol. 4, no. 2, pp. 158–167, 2015.
- [7] G. Mao, B. Fidan, and B. D. Anderson, "Wireless sensor network localization techniques," *Comput. Netw.*, vol. 51, no. 10, pp. 2529–2553, 2007.
- [8] G. Mao, B. D. Anderson, and B. Fidan, "Online calibration of path loss exponent in wireless sensor networks," in *Proc. IEEE GLOBECOM*, 2006, pp. 1–6.
- [9] C.-H. Park and J.-H. Chang, "Closed-form localization for distributed MIMO radar systems using time delay measurements," *IEEE Trans. Wireless Commun.*, vol. 15, no. 2, pp. 1480–1490, 2016.
- [10] K. Yang, G. Wang, and Z.-Q. Luo, "Efficient convex relaxation methods for robust target localization by a sensor network using time differences of arrivals," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2775–2784, 2009.
- [11] C. Liu, D. Fang, Z. Yang, H. Jiang, X. Chen, W. Wang, T. Xing, and L. Cai, "RSS distribution-based passive localization and its application in sensor networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 4, pp. 2883–2895, 2016.
- [12] Y. Zhang, Y. Lou, Y. Hong, and L. Xie, "Distributed projection-based algorithms for source localization in wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 14, no. 6, pp. 3131–3142, 2015.
- [13] N. Patwari, A. O. Hero III, M. Perkins, N. S. Correal, and R. J. O’dea, "Relative location estimation in wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 51, no. 8, pp. 2137–2148, 2003.
- [14] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero III, R. L. Moses, and N. S. Correal, "Locating the nodes: cooperative localization in wireless sensor networks," *IEEE Signal Process. Mag.*, vol. 22, no. 4, pp. 54–69, 2005.
- [15] T. Van Nguyen, Y. Jeong, H. Shin, and M. Z. Win, "Least square cooperative localization," *IEEE Trans. Veh. Technol.*, vol. 64, no. 4, pp. 1318–1330, 2015.
- [16] S. Zhu and Z. Ding, "Distributed cooperative localization of wireless sensor networks with convex hull constraint," *IEEE Trans. Wireless Commun.*, vol. 10, no. 7, pp. 2150–2161, 2011.
- [17] H. Wymeersch, J. Lien, and M. Z. Win, "Cooperative localization in wireless networks," *Proc. IEEE*, vol. 97, no. 2, pp. 427–450, 2009.
- [18] B. D. Anderson, I. Shames, G. Mao, and B. Fidan, "Formal theory of noisy sensor network localization," *SIAM J. Discrete Math.*, vol. 24, no. 2, pp. 684–698, 2010.
- [19] F. C. Schweppe, "Recursive state estimation: unknown but bounded errors and system inputs," *IEEE Trans. Autom. Control*, vol. 13, no. 1, pp. 22–28, 1968.
- [20] Y. C. Eldar, A. Beck, and M. Teboulle, "A minimax Chebyshev estimator

- for bounded error estimation," *IEEE Trans. Signal Process.*, vol. 56, no. 4, pp. 1388–1397, 2008.
- [21] M. Milanese and A. Vicino, "Optimal estimation theory for dynamic systems with set membership uncertainty: an overview," in *Bounding Approaches to System Identification*. Springer, 1996, pp. 5–27.
- [22] M. R. Gholami, E. G. Ström, H. Wymeersch, and S. Gezici, "Upper bounds on position error of a single location estimate in wireless sensor networks," *EURASIP J. Adv. Signal Process.*, vol. 2014, no. 1, pp. 1–14, 2014.
- [23] G. Wang, A. M.-C. So, and Y. Li, "Robust convex approximation methods for TDOA-based localization under NLOS conditions," *IEEE Trans. Signal Process.*, vol. 64, no. 13, pp. 3281–3296, 2016.
- [24] G. Belforte, B. Bona, and V. Cerone, "Bounded measurement error estimates: Their properties and their use for small sets of data," *Measurement*, vol. 5, no. 4, pp. 167–175, 1987.
- [25] A. A. Kannan, B. Fidan, and G. Mao, "Robust distributed sensor network localization based on analysis of flip ambiguities," in *Proc. IEEE GLOBECOM*, 2008, pp. 1–6.
- [26] P. Biswas, T.-C. Lian, T.-C. Wang, and Y. Ye, "Semidefinite programming based algorithms for sensor network localization," *ACM Trans. Sens. Netw.*, vol. 2, no. 2, pp. 188–220, 2006.
- [27] A. Simonetto and G. Leus, "Distributed maximum likelihood sensor network localization," *IEEE Trans. Signal Process.*, vol. 62, no. 6, pp. 1424–1437, 2014.
- [28] Y. Shang, W. Rumi, Y. Zhang, and M. Fromherz, "Localization from connectivity in sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 15, no. 11, pp. 961–974, 2004.
- [29] Z. Wang, S. Zheng, S. Boyd, and Y. Ye, "Further relaxations of the SDP approach to sensor network localization," Stanford University, Tech. Rep., Tech. Rep., 2006.
- [30] P. Tseng, "Second-order cone programming relaxation of sensor network localization," *SIAM J. Opt.*, vol. 18, no. 1, pp. 156–185, 2007.
- [31] P. Biswas, T.-C. Liang, K.-C. Toh, Y. Ye, and T.-C. Wang, "Semidefinite programming approaches for sensor network localization with noisy distance measurements," *IEEE Trans. Autom. Sci. Eng.*, vol. 3, no. 4, pp. 360–371, 2006.
- [32] G. C. Calafiore, L. Carlone, and M. Wei, "Distributed optimization techniques for range localization in networked systems," in *Proc. IEEE CDC*, 2010, pp. 2221–2226.
- [33] S. Srirangarajan, A. H. Tewfik, and Z.-Q. Luo, "Distributed sensor network localization using SOCP relaxation," *IEEE Trans. Wireless Commun.*, vol. 7, no. 12, pp. 4886–4895, 2008.
- [34] Q. Shi, C. He, H. Chen, and L. Jiang, "Distributed wireless sensor network localization via sequential greedy optimization algorithm," *IEEE Trans. Signal Process.*, vol. 58, no. 6, pp. 3328–3340, 2010.
- [35] M. Wei, R. Aragues, C. Sagues, and G. C. Calafiore, "Noisy range network localization based on distributed multidimensional scaling," *IEEE Sensors J.*, vol. 15, no. 3, pp. 1872–1883, 2015.
- [36] J. A. Costa, N. Patwari, and A. O. Hero III, "Distributed weighted-multidimensional scaling for node localization in sensor networks," *ACM Trans. Sens. Netw.*, vol. 2, no. 1, pp. 39–64, 2006.
- [37] U. Khan, S. Kar, J. M. Moura *et al.*, "Distributed sensor localization in random environments using minimal number of anchor nodes," *IEEE Trans. Signal Process.*, vol. 57, no. 5, pp. 2000–2016, 2009.
- [38] Y. Diao, Z. Lin, and M. Fu, "A barycentric coordinate based distributed localization algorithm for sensor networks," *IEEE Trans. Signal Process.*, vol. 62, no. 18, pp. 4760–4771, 2014.
- [39] T. Eren, D. K. Goldenberg, W. Whiteley, Y. R. Yang, A. S. Morse, B. D. Anderson, and P. Belhumeur, "Rigidity, computation, and randomization in network localization," in *Proc. IEEE INFOCOM 2014*.
- [40] M. Deghat, I. Shames, B. D. Anderson, and J. M. Moura, "Distributed localization via barycentric coordinates: Finite-time convergence," *Proc. IFAC*, vol. 44, no. 1, pp. 7824–7829, 2011.
- [41] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.
- [42] Z.-Q. Luo, W.-k. Ma, A. M.-C. So, Y. Ye, and S. Zhang, "Semidefinite relaxation of quadratic optimization problems," *IEEE Signal Process. Mag.*, vol. 27, no. 3, pp. 20–34, 2010.
- [43] A. M.-C. So and Y. Ye, "Theory of semidefinite programming for sensor network localization," *Math. Program.*, vol. 109, no. 2-3, pp. 367–384, 2007.
- [44] R. T. Rockafellar, *Convex analysis*. Princeton university press, 2015.
- [45] J. F. Sturm, "Using sedumi 1.02, a MATLAB toolbox for optimization over symmetric cones," *Opt. Methods Softw.*, vol. 11, no. 1-4, pp. 625–653, 1999.
- [46] M. Grant, S. Boyd, and Y. Ye, "CVX: Matlab software for disciplined convex programming," 2008.
- [47] E. D. Anderson, "How to use Farkas's lemma to say something important about linear infeasible problems," *MOSEK Technical Report TR-2011-1*, 2011.
- [48] D. Wu, J. Zhou, and A. Hu, "A new approximate algorithm for the Chebyshev center," *Automatica*, vol. 49, no. 8, pp. 2483–2488, 2013.
- [49] Y. Y. Druzhinin, "Existence of a Lipschitz selection of the Chebyshev-centre map," *Sbornik: Mathematics*, vol. 204, no. 5, p. 641, 2013.
- [50] D. Niculescu and B. Nath, "DV based positioning in ad hoc networks," *Telecommun. Syst.*, vol. 22, no. 1-4, pp. 267–280, 2003.
- [51] K. N. Chaudhury, Y. Khoo, and A. Singer, "Large-scale sensor network localization via rigid subnetwork registration," in *Proc. IEEE ICASSP*, 2015, pp. 2849–2853.
- [52] R. Sanyal and K. N. Chaudhury, "Scalable and accurate sensor network localization using cliques," in *Proc. IEEE SPCOM*, 2016, pp. 2849–2853.