On geometric upper bounds for positioning algorithms in wireless sensor networks

On Geometric Upper Bounds for Positioning Algorithms in Wireless Sensor Networks

Mohammad Reza Gholami[a,*], Erik G. Ström and Henk Wymeersch[b], Mats Rydström[c]

[a] *ACCESS Linnaeus Center, Electrical Engineering, KTH–Royal Institute of Technology, SE-100 44 Stockholm, Sweden.*
[b] *Division of Communication Systems, Information Theory, and Antennas, Department of Signals and Systems, Chalmers University of Technology*
[c] *Patent professional at Ericsson.*

Abstract

This paper studies the possibility of upper bounding the position error for range-based positioning algorithms in wireless sensor networks. In this study, we argue that in certain situations when the measured distances between sensor nodes have positive errors, e.g., in non-line-of-sight (NLOS) conditions, the target node is confined to a closed bounded convex set (a feasible set) which can be derived from the measurements. Then, we formulate two classes of geometric upper bounds with respect to the feasible set. If an estimate is available, either feasible or infeasible, the position error can be upper bounded as the maximum distance between the estimate and any point in the feasible set (the first bound). Alternatively, if an estimate given by a positioning algorithm is always feasible, the maximum length of the feasible set is an upper bound on position error (the second bound). These bounds are formulated as nonconvex optimization problems. To progress, we relax the nonconvex problems and obtain convex problems, which can be efficiently solved. Simulation results show that the proposed bounds are reasonably tight in many situations, especially for NLOS conditions.

Keywords: Wireless sensor networks, positioning problem, projection onto convex set, convex feasibility problem, semidefinite relaxation, quadratic programming, position error, worst-case position error, non-line-of-sight.

1. Introduction

Recent advances in technology have instigated the use of tiny devices as sensors in large distributed wireless sensor networks (WSNs). A sensor device is capable to sense its environment for monitoring, controlling, or tracking purposes for both civil and military applications [1]. Due to drawbacks in using GPS for WSNs, extracting the position information from the network, also called localization, has been extensively

*Corresponding author
Email addresses: mohrg@kth.se (Mohammad Reza Gholami), {erik.strom, henkw}@chalmers.se) (Erik G. Ström and Henk Wymeersch), jan.mats.ake.rydstrom@gmail.com (Mats Rydström)

studied in the literature [2, 1, 3, 4, 5, 6]. It is commonly assumed that there are a number of fixed reference sensors, also called anchors, whose positions are *a priori* known, e.g., by using GPS receivers [7]. To find the position of other sensor nodes at unknown positions, henceforth called target nodes, it is assumed that there are some types of measurements, e.g., time-of-arrival, angle-of-arrival, or received signal strength, taken between sensor nodes [1].

During the last decades, various positioning algorithms have been proposed in the literature. Different positioning approaches can be categorized based on various factors [8]. For instance, as long as an accurate model of measurements and the statistics of the measurement errors are known, classic estimators, e.g., the maximum likelihood (ML) and the least squares (LS) approaches, can be employed successfully to solve the positioning problem. When the distribution of the measurement errors is unknown or the computational complexity of classic estimators is too high, a number of simple techniques can be applied to the problem. For example, suboptimal algorithms such as semidefinite programming (SDP) [9] or closed-form linear least squares (LLS) [10, 11] have been successfully applied to the positioning problem. In one class of suboptimal algorithms based on a geometric interpretation, the authors of [12, 13] formulated the positioning problem as a convex feasibility problem (CFP) and applied the well-known orthogonal projection onto convex sets (POCS) approach to solve the problem. This method turns out to be robust against non-line-of-sight (NLOS) conditions [14]. POCS was previously studied for the CFP and has found applications in several research fields [15, 16].

Positioning algorithms can be evaluated based on different performance metrics such as complexity, accuracy, and coverage [8]. In the literature one way to assess the positioning algorithms is to evaluate the position error, defined as the Euclidian norm of the difference between the position estimate and the true position. There are a number of techniques to evaluate the performance of an algorithm based on the position error. For instance, a lower bound on the mean square position error is a common metric [17, 18]. There exist a number of such lower bounds for the positioning algorithms in the literature. For example the Cramér-Rao lower bound (CRLB), which gives a lower bound on the variance of any unbiased estimator, can be computed if the probability density function (PDF) of the measurement error is known and satisfies some regularity conditions [19]. Generally, different benchmarks in the literature are used to *statistically* assess a positioning algorithm, which implies that the error in a single position estimate cannot be characterized in a deterministic fashion.

Besides a lower bound on the position error, in some applications it may be useful to know the worst-case behavior of the position error. Such knowledge may be useful not only for evaluation of different services provided by WSNs but also for design and resource management [1, 20]. Similarly in evaluation of the

worst-case position error, we may be interested in assessing a single point estimate. As an example consider Fig. 1, which shows how a nontrivial (i.e., finite) upper bound on position error can be used by a traffic safety application to decrease collisions between vehicles. If an estimate of a vehicle and a nontrivial upper bound on the position error are available, we can define an area in which the vehicle is certainly located, e.g., a disc centered at the position estimate and with a radius equal to the upper bound on the position error. Such an estimate can be obtained in every vehicle, for instance, by measuring the distance between the vehicle and a number of fixed nodes (at known positions) along the road. The estimate of cars' positions and upper bounds on the position errors can be exchanged between vehicles. By this approach, we may be able to decrease the number of collisions between vehicles. In general, computing the position error might be difficult since the true position is unknown, but one may be able to derive an upper bound on the position error. To the best of our knowledge, there is no specific work in the literature on deriving upper bounds on the position error. In this study, we aim at tackling this subject in a geometric framework.

In general, the concept of an upper bound on the position error (or any estimation error) seems to be questionable. In fact, it is not clear if it is meaningful to study upper bounds, since the position error can, in general, be arbitrarily large. In this study, however, we argue that in some practical situations, the position error is finite and can be upper bounded. For instance, if a target node position belongs to a closed bounded set (a feasible set), an upper bound on the position error can be computed from the feasible set. For example, for distance-based positioning, if measurement errors are assumed to be positive, a convex set including the target node can be defined from measurements. The feasible set, in which the target node is located, is the intersection of a number of balls (in a 3-dimensional network) or discs (in a 2-dimensional network) centered at the position of reference nodes [21]. The assumption of positive measurement errors is fulfilled in some scenarios. For instance, in NLOS conditions, the measured distances are often much larger than the actual distances. For practical ranging using UWB, it has been observed that the measurement errors tend to be positive, even for line-of-sight (LOS) scenarios [22]. It should be noted that the measurement error, in general, can be negative as well, meaning the intersection no longer contains the location of the target node and the bounding technique may not work properly. In such scenarios, one can, e.g., modify the measurements and obtain a new set of distance measurements that are larger than the actual distances. For example, if a reasonable lower bound on negative measurement errors is available, then we can enlarge the measurements with the absolute value of the lower bound and obtain a set of measurements with positive errors. Now, assuming a closed bounded (compact) convex set derived from distance measurements having positive errors, a position estimate given by an algorithm can be either feasible or infeasible with respect to the feasible set. If an estimate is available (feasible or infeasible), it is reasonable to define the maximum

distance from the estimate to any point in the feasible region as an upper bound on position error. This idea yields an upper bound on the position error as the solution of a nonconvex optimization problem. Alternatively, a number of positioning algorithms, e.g., POCS, give one feasible point as an estimate. For this type of estimators, we can upper bound the position error as the maximum length[1] of the feasible set. To find the maximum length of the feasible region, we consider an outer-approximation of the feasible set and find the minimum Euclidean ball or the minimum ℓ_∞ ball (minimum bounding box) covering the set. We further relax the nonconvex optimization problem and derive a convex optimization problem. Obviously, if a feasible point is available, the first upper bound, i.e., the maximum distance from the estimate to any point in the feasible region, gives a tighter upper bound compared to the second bound, i.e., the maximum length of the feasible region. Note that the technique introduced in this paper can be applied to every estimation problem when the unknown parameter vector belongs to a compact, finite-volume, convex set.

In summary, the main contributions of this study are:

- introducing the concept of an instantaneous upper bound for a single point position estimate when the distance measurements have positive errors, e.g., in NLOS conditions;
- proposing an upper bound on the position error based on a convex relaxation technique when an estimate of the target position is available (feasible or infeasible);
- proposing three upper bounds for an estimator always giving a feasible point as an estimate (e.g., the POCS estimate) based on the idea of the maximum length of the feasible set or a relaxed feasible set including the target node.

The remainder of the paper is organized as follows. Some preliminary requirements are studied in Section 2. Section 3 explains the signal model considered in this paper. In Section 4, a geometric positioning algorithm is briefly studied, which serves as a basis for obtaining an upper bound. Two types of upper bounds are derived in Section 5. Simulation results are discussed in Section 6. Finally, Section 7 makes come concluding remarks.

[1] By the maximum length of a set, we mean the maximum ℓ_2 norm of the difference between two points (not necessarily a unique pair of points) in the set.

2. preliminaries

2.1. Notation

The following notations are used in this study. Lowercase and bold lowercase letters denote scalar values and vectors, respectively. Matrices are written using bold uppercase letters. By $\mathbf{0}_{n\times n}$ we denote the n by n zero matrix, and we use $\mathbf{0}_n$ as the n-vector of n zeros. $\mathbf{1}_n$ and $\boldsymbol{I}_n$ denote the vector of n ones and the n by n identity matrix, respectively. The operator $\mathrm{tr}(\cdot)$ is used to denote the trace of a square matrix. The ℓ_p norm is denoted by $\|\cdot\|_p$. Given two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$, $\boldsymbol{A} \succ (\succeq) \boldsymbol{B}$ means that $\boldsymbol{A}-\boldsymbol{B}$ is positive (semi)definite. $\mathbb{S}^n$, $\mathbb{R}^n$, and $\mathbb{R}^n_+$ denote the set of all $n\times n$ symmetric matrices, the set of all $n\times 1$ vectors with real values, and the set of all $n\times 1$ vectors with nonnegative real values, respectively.

2.2. Quadratically constrained quadratic programming

Let us consider a quadratically constrained quadratic program (QCQP) as

$$\begin{aligned} &\underset{\boldsymbol{x}\in\mathbb{R}^n}{\text{maximize}} \;\; \boldsymbol{x}^T\boldsymbol{A}_0\boldsymbol{x} + 2\boldsymbol{b}_0^T\boldsymbol{x} + c_0 \\ &\text{subject to } \boldsymbol{x}^T\boldsymbol{A}_i\boldsymbol{x} + 2\boldsymbol{b}_i^T\boldsymbol{x} + c_i \le 0, \quad i=1,\ldots,N \end{aligned} \tag{1}$$

for $\boldsymbol{A}_i \in \mathbb{S}^n$, $\boldsymbol{b}_i \in \mathbb{R}^n$, and $c_i \in \mathbb{R}$. For nonconvex QCQP in (1), we can employ a relaxation technique and obtain a semidefinite programming problem (SDP) as

$$\begin{aligned} &\underset{\boldsymbol{x}\in\mathbb{R}^n,\; \boldsymbol{Z}\in\mathbb{S}^n}{\text{maximize}} \; \mathrm{tr}(\boldsymbol{A}_0\boldsymbol{Z}) + 2\boldsymbol{b}_0^T\boldsymbol{x} + c_0 \\ &\text{subject to } \mathrm{tr}(\boldsymbol{A}_i\boldsymbol{Z}) + 2\boldsymbol{b}_i^T\boldsymbol{x} + c_i \le 0, \qquad i=1,\ldots,N \\ &\qquad\qquad \begin{bmatrix} \boldsymbol{Z} & \boldsymbol{x} \\ \boldsymbol{x}^T & 1 \end{bmatrix} \succeq 0. \end{aligned} \tag{2}$$

For details of the relaxation technique, see, e.g., [23, 24]. To refer to the QCQP formulated in (1) throughout this paper, we use $\mathrm{QP}\{\boldsymbol{A}_i, \boldsymbol{b}_i, c_i\}_{i=0}^N$. Similarly, to refer to the SDP relaxation (2) of the QCQP in (1), we use $\mathrm{SDP}\{\boldsymbol{A}_i, \boldsymbol{b}_i, c_i\}_{i=0}^N$. For the optimal values of the objective function of the QCQP and the corresponding SDP relaxation in (1) and in (2), we use $v_{\mathrm{qp}}\{\boldsymbol{A}_i, \boldsymbol{b}_i, c_i\}_{i=0}^N$ and $v_{\mathrm{sdp}}\{\boldsymbol{A}_i, \boldsymbol{b}_i, c_i\}_{i=0}^N$, respectively. By adopting the relaxation in (2), we expand the feasible set, therefore, the objective function in (2) is maximized over a larger set than in (1), thus

$$v_{\mathrm{qp}}\{\boldsymbol{A}_i, \boldsymbol{b}_i, c_i\}_{i=0}^N \le v_{\mathrm{sdp}}\{\boldsymbol{A}_i, \boldsymbol{b}_i, c_i\}_{i=0}^N. \tag{3}$$

That is, the optimal value in (2) gives an upper bound on the optimal value in (1). Further steps can be taken to improve the accuracy of the solution, e.g., based on a rank-1 approximation [23] or linearization technique [25].

2.3. Bounds on estimation errors given a realization of the measurement vector

Consider an unknown parameter vector $\boldsymbol{x} \in \mathbb{R}^n$. Regardless if we model $\boldsymbol{x}$ as random or unknown deterministic, we can define the set of the possible values of $\boldsymbol{x}$ as

$$\mathcal{X} \triangleq \{\text{possible values of } \boldsymbol{x}\} \subseteq \mathbb{R}^n$$

Suppose $\boldsymbol{m}$ is the observed realization of the (random) measurement vector $\mathbb{M}$. Given the event $\mathbb{M} = \boldsymbol{m}$, the set of possible values of $\boldsymbol{x}$ changes to

$$\mathcal{X}(\boldsymbol{m}) \triangleq \{\text{possible values of } \boldsymbol{x} : \mathbb{M} = \boldsymbol{m}\} \subseteq \mathcal{X}.$$

The estimate of $\boldsymbol{x}$, denoted by $\hat{\boldsymbol{x}}(\boldsymbol{m}, \boldsymbol{f}) \in \mathbb{R}^n$, is a function of the observed data $\boldsymbol{m}$ and some algorithm tuning parameters, e.g., initialization, step size, termination criterion, etc., which are collected in the vector $\boldsymbol{f}$. The $\boldsymbol{f}$-vector is chosen, possibly randomly, from the set $\mathcal{F}$. In other words, $\boldsymbol{f} \in \mathcal{F}$ completely determines how the estimator maps the observed data $\boldsymbol{m}$ to the estimate $\hat{\boldsymbol{x}}$, and the set $\mathcal{F}$ defines a class of estimators. We can now define the set of possible values of $\hat{\boldsymbol{x}}(\boldsymbol{m}, \boldsymbol{f})$ when $\boldsymbol{f}$ can take on any value in $\mathcal{F}$ as

$$\hat{\mathcal{X}}(\boldsymbol{m}) \triangleq \{\hat{\boldsymbol{x}}(\boldsymbol{m}, \boldsymbol{f}) : \boldsymbol{f} \in \mathcal{F}\} \subset \mathbb{R}^n.$$

We can define three upper bounds on the ℓ_2 norm of estimation error $e \triangleq \|\hat{\boldsymbol{x}}(\boldsymbol{m}, \boldsymbol{f}) - \boldsymbol{x}\|_2$ as

$$e \leq u_1(\hat{\boldsymbol{x}}(\boldsymbol{m}, \boldsymbol{f})) \triangleq \sup_{\boldsymbol{x} \in \mathcal{X}(\boldsymbol{m})} \|\hat{\boldsymbol{x}}(\boldsymbol{m}, \boldsymbol{f}) - \boldsymbol{x}\|_2, \tag{4}$$

$$e \leq u_2(\boldsymbol{x}) \triangleq \sup_{\hat{\boldsymbol{x}} \in \hat{\mathcal{X}}(\boldsymbol{m})} \|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2, \tag{5}$$

$$e \leq u_3 \triangleq \sup_{\boldsymbol{x} \in \mathcal{X}(\boldsymbol{m}),\ \hat{\boldsymbol{x}} \in \hat{\mathcal{X}}(\boldsymbol{m})} \|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2. \tag{6}$$

We note that all bounds depends on $\boldsymbol{m}$, which, for simplicity, is neglected in the notation. Moreover, it is easy to see that $u_1(\hat{\boldsymbol{x}}(\boldsymbol{m}, \boldsymbol{f})) \leq u_3$ and $u_2(\boldsymbol{x}) \leq u_3$. Fig. 2 graphically shows the different upper bounds.

Remark 1. *The bound $u_1(\hat{\boldsymbol{x}}(\boldsymbol{m}, \boldsymbol{f}))$ is an upper bound of the norm of the estimation error for a certain estimate ($\boldsymbol{f}$ and $\boldsymbol{m}$ are fixed). Hence, if $u_1(\hat{\boldsymbol{x}}(\boldsymbol{m}, \boldsymbol{f}))$ can be computed together with the estimate, this would greatly increase the value of the estimate, since we can now guarantee that the norm of the estimation error in $\hat{\boldsymbol{x}}(\boldsymbol{m}, \boldsymbol{f})$ does not exceed $u_1(\hat{\boldsymbol{x}}(\boldsymbol{m}, \boldsymbol{f}))$.*

Remark 2. *The bound u_3 could potentially be computed together with the estimate and is therefore of value in a practical situation. However, u_3 will only be interesting if it is easier to compute than $u_1(\hat{\boldsymbol{x}}(\boldsymbol{m}, \boldsymbol{f}))$, since $u_1(\hat{\boldsymbol{x}}(\boldsymbol{m}, \boldsymbol{f})) \leq u_3$.*

Remark 3. *the bound $u_2(\boldsymbol{x})$ can be interpreted as the error of the worst estimate that is computed from the observed data $\boldsymbol{m}$ by the class of estimators defined by $\mathcal{F}$. This is useful to judge the worst case performance of a class of estimators. However, since the bound is a function of $\boldsymbol{x}$ (the unknown parameter), it cannot be computed together with an estimate, and its practical value is therefore limited.*

We can also formulate lower bounds by replacing sup with inf in Eqs. (4)– (6),

$$e \geq \ell_1(\hat{\boldsymbol{x}}(\boldsymbol{m}, \boldsymbol{f})) \triangleq \inf_{\boldsymbol{x} \in \mathcal{X}(\boldsymbol{m})} \|\hat{\boldsymbol{x}}(\boldsymbol{m}, \boldsymbol{f}) - \boldsymbol{x}\|_2, \tag{7}$$

$$e \geq \ell_2(\boldsymbol{x}) \triangleq \inf_{\hat{\boldsymbol{x}} \in \hat{\mathcal{X}}(\boldsymbol{m})} \|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2, \tag{8}$$

$$e \geq \ell_3 \triangleq \inf_{\boldsymbol{x} \in \mathcal{X}(\boldsymbol{m}),\ \hat{\boldsymbol{x}} \in \hat{\mathcal{X}}(\boldsymbol{m})} \|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2. \tag{9}$$

In general, there are no guarantees that any of the bounds in Eqs. (4)–(9) are nontrivial, i.e., that the upper bounds are finite and the lower bounds are greater than zero. For example, if the set $\mathcal{X}(\boldsymbol{m})$ or $\hat{\mathcal{X}}(\boldsymbol{m})$ is unbounded, it is clear that the upper bound (4) or (6) is trivial. However, as we will see in the remainder of this paper, there are indeed practical situations when the bounds are nontrivial.

3. System Model

Let us consider an n-dimensional network, $n = 2$ or 3, with N reference nodes at known positions $\boldsymbol{a}_i = [a_{i,1} \ \cdots \ a_{i,n}]^T \in \mathbb{R}^n$, $i = 1, ..., N$. Suppose that a target node is placed at an unknown position $\boldsymbol{x} = [x_1 \ \cdots \ x_n]^T \in \mathbb{R}^n$. The range measurement between the target node and reference node i is given by

$$\hat{d}_i = d_i(\boldsymbol{x}, \boldsymbol{a}_i) + \epsilon_i, \qquad i = 1, \ldots, N, \tag{10}$$

where $d_i(\boldsymbol{x}, \boldsymbol{a}_i)$ is the actual Euclidian distance between the target node and reference node i, i.e., $d_i(\boldsymbol{x}, \boldsymbol{a}_i) = \|\boldsymbol{a}_i - \boldsymbol{x}\|_2$, and ϵ_i is the measurement error.

In the literature the measurement error is commonly modeled as a zero mean Gaussian random variable [1, 26**?**]. In some scenarios, however, other distributions seem to be more reasonable. For instance, in NLOS conditions the measured distances are larger than the actual distances with high probability. A number of distributions have been considered to model NLOS conditions, e.g., an exponential distribution or a uniform distribution [27]. The Gaussian distribution with large positive mean has also been considered to model the NLOS condition [27, 28]. We may also model the NLOS measurements using two different distributions, namely, a Gaussian distribution for thermal noise and a particular distribution, e.g., exponential, for NLOS part. In this paper for the purpose of deriving an upper bound, we assume that the distance measurements have positive errors, meaning that the measurement errors are nonnegative. The positive

measurement assumption can be fulfilled, e.g., in NLOS conditions (with high probability). In recent practical measurements using UWB, it has been observed that the measurement noise tends to be positive [22]. In fact, time-of-arrival-based ranging typically involves setting a threshold such that false alarms (negative errors due to noise peaks) are negligible. That way negative ranging errors can be considered to occur very rarely, if at all. For details of such behavior of measurement errors, please see [22].

The positioning problem, then, is to find the position of the target node based on the positions of N reference nodes and measurements made in (10).

4. Positioning algorithms

In this section, we consider three well-known positioning algorithms, namely, SDP, LLS, and POCS, formulated in the literature. Note that there are a huge number of algorithms and we have selected these three algorithms for the following reasons:
1-SDP gives a good estimate in many scenarios; 2-LLS has a closed form solution and also gives good estimates in some scenarios; 3-POCS is a simple algorithm that gives a good coarse estimate. POCS is also robust to large positive distance errors and is therefore suitable for NLOS scenarios. The idea behind the POCS is also employed to formulate upper bounds on the position error for estimates computed by POCS or other approaches. In this section, we briefly review the POCS and for details of SDP and LLS, we refer the reader to [9, 10]. Note that in this study, the main goal is to obtain a technique to find an upper bound on the position error. To evaluate the technique, we will consider a number of techniques. For example for NLOS, one may use approaches investigated in, e.g., [29, 30, 31, 32, 33] to evaluate the bound. In this study, we consider the same algorithms for both LOS and NLOS, perhaps not optimal, to assess the proposed upper bound using these algorithms. Similar results can be obtained for other bounds.

To formulate the POCS, let us consider a least squares minimization to find an estimate of the target position as follows:

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}\in\mathbb{R}^n} \sum_{i=1}^{N} \Big(\hat{d}_i - d_i(\boldsymbol{x}, \boldsymbol{a}_i)\Big)^2. \tag{11}$$

In the absence of measurement errors, i.e., $\hat{d}_i = d_i(\boldsymbol{x}, \boldsymbol{a}_i)$, it is clear that the target node, at unknown position $\boldsymbol{x}$, can be found in the intersection of a number of spheres with radii $d_i(\boldsymbol{x}, \boldsymbol{a}_i)$ and centers $\boldsymbol{a}_i$. For nonnegative measurement errors, we relax spheres to balls and deduce that the target node position definitely lies inside the intersection of a number of balls. Let us define the (closed bounded) ball $\mathcal{B}_i$ centered at $\boldsymbol{a}_i$ as

$$\mathcal{B}_i \triangleq \big\{\boldsymbol{x} \in \mathbb{R}^n \ : \ \|\boldsymbol{x} - \boldsymbol{a}_i\|_2 \le \hat{d}_i\big\}, \qquad i = 1, \ldots, N. \tag{12}$$

It is then reasonable to define an estimate of $\boldsymbol{x}$ as a point in the intersection $\mathcal{B}$ (a closed bounded set) of the balls $\mathcal{B}_i$ (a feasible point) as

$$\hat{\boldsymbol{x}} \in \mathcal{B} \triangleq \bigcap_{i=1}^{N} \mathcal{B}_i. \tag{13}$$

Therefore, the positioning problem can be rendered to the following convex feasibility problem (CFP):

$$\begin{aligned} &\underset{\boldsymbol{x}\in\mathbb{R}^n}{\text{minimize}}\, 0 \\ &\text{subject to } \|\boldsymbol{x}-\boldsymbol{a}_i\| \le \hat{d}_i, \quad i=1,\ldots,N. \end{aligned} \tag{14}$$

To solve (14), we note that CFP can be reformulated by minimizing the following convex function

$$f(\boldsymbol{x}) \triangleq \max\{\text{dist}(\boldsymbol{x},\mathcal{B}_1),\ldots,\text{dist}(\boldsymbol{x},\mathcal{B}_N)\}, \tag{15}$$

with $\text{dist}(\boldsymbol{x},\mathcal{B}_i)$ denoting the minimum distance between $\boldsymbol{x}$ and any point in set $\mathcal{B}_i$. We note that $f(\boldsymbol{x}) \ge 0$ with equality if and only if $\boldsymbol{x}$ is in all balls $\mathcal{B}_1, \mathcal{B}_2, \ldots, \mathcal{B}_N$.

Using negative subgradient updating method [34, 15], we can obtain a solution to (15) by

$$\boldsymbol{x}^{k+1} = \boldsymbol{x}^k - \alpha_k \boldsymbol{g}^k, \quad k=0,1,\ldots, \tag{16}$$

where $\boldsymbol{x}^k$ is the kth iterate, α_k is the kth step size, and $\boldsymbol{g}^k$ is a subgradient[2]. A subgradient $\boldsymbol{g}^k$ of f at $\boldsymbol{x}^k$ can be computed as

$$\boldsymbol{g}^k = \begin{cases} 0, & \text{if } f(\boldsymbol{x}^k)=0, \\ \frac{\boldsymbol{x}^k - \mathcal{P}_{\mathcal{B}_j}(\boldsymbol{x}^k)}{\|\boldsymbol{x}^k - \mathcal{P}_{\mathcal{B}_j}(\boldsymbol{x}^k)\|_2}, & \text{if } f(\boldsymbol{x}^k)\neq 0,\ j = \arg\max\limits_i\ \text{dist}(\boldsymbol{x}^k,\mathcal{B}_i), \end{cases} \tag{17}$$

where $\mathcal{P}_{\mathcal{B}_j}(\boldsymbol{x}^k)$ is the orthogonal projection of $\boldsymbol{x}^k$ onto the set $\mathcal{B}_j$. By choosing the step size as $\alpha_k = f(\boldsymbol{x}^k)/\|\boldsymbol{g}^k\|_2^2 = \|\boldsymbol{x}^k - \mathcal{P}_{\mathcal{B}_j}(\boldsymbol{x}^k)\|_2/\|\boldsymbol{g}^k\|_2^2 = \|\boldsymbol{x}^k - \mathcal{P}_{\mathcal{B}_j}(\boldsymbol{x}^k)\|_2$ in (16), according to Polyak approach [15], we derive the following approach, called alternating projections [35] or POCS, for updating

$$\begin{aligned} \boldsymbol{x}^{k+1} &= \boldsymbol{x}^k - \alpha_k \boldsymbol{g}^k \\ &= \boldsymbol{x}^k - \|\boldsymbol{x}^k - \mathcal{P}_{\mathcal{B}_j}(\boldsymbol{x}^k)\|_2 \frac{\boldsymbol{x}^k - \mathcal{P}_{\mathcal{B}_j}(\boldsymbol{x}^k)}{\|\boldsymbol{x}^k - \mathcal{P}_{\mathcal{B}_j}(\boldsymbol{x}^k)\|_2}, \\ &= \boldsymbol{x}^k - (\boldsymbol{x}^k - \mathcal{P}_{\mathcal{B}_j}(\boldsymbol{x}^k)) = \mathcal{P}_{\mathcal{B}_j}(\boldsymbol{x}^k). \end{aligned} \tag{18}$$

where index j is the one used in (17).

[2]Let $\mathcal{D}$ be a nonempty set in $\mathbb{R}^n$. A vector $\boldsymbol{g}\in\mathbb{R}^n$ is a subgradient of a function $f:\mathcal{D}\to\mathbb{R}$ at $\boldsymbol{x}\in\mathcal{D}$ if $f(\boldsymbol{y}) \ge f(\boldsymbol{x}) + \boldsymbol{g}^T(\boldsymbol{y}-\boldsymbol{x})$ for all $\boldsymbol{y}\in\mathcal{D}$ [15].

As mentioned before, POCS gives an estimate that is feasible (if the intersection $\mathcal{B}$ is nonempty). In each step, POCS projects the current point $\boldsymbol{x}^k$ onto the farthest convex set. For example, Fig. 3 shows a 2-dimensional network in which the measured distances in reference nodes have positive errors. The POCS' estimate in this figure converges to a point in the intersection of three discs after two iterations. For more details on variations of the POCS algorithm and the application of POCS for the positioning problem, we refer the reader to [15] and [36, 12, 14], respectively.

Remark 4. *As mentioned earlier, for negative measurement errors the intersection $\mathcal{B}$ defined in* (13) *does not contain the location of the target node, even for nonempty intersection. For such a scenario, we may need more information about the noise statistics. One simple approach can be considered as follows. Let the measurement error in* (10) *be bounded from below, i.e., $-\rho^i \leq \epsilon_i \;\; \rho^i, \in \mathbb{R}_+$. It means that the measured distances can be bounded as [37]*

$$d_i(\boldsymbol{x}, \boldsymbol{a}_i) - \rho^i \leq \hat{d}_i. \tag{19}$$

We now modify measurements as

$$\tilde{d}_i = \hat{d}_i + \rho^i \geq d_i(\boldsymbol{x}, \boldsymbol{a}_i), \tag{20}$$

where we assume that the values of ρ_1^i are known in advance. It is observed that the new measurements, i.e., $\tilde{d}_i$, have positive errors. Let us form a new intersection $\mathcal{B}'$ as

$$\mathcal{B}' = \bigcap_{i=1}^{N} \mathcal{B}'_i, \tag{21}$$

where $\mathcal{B}'_i \triangleq \{\boldsymbol{x} : \|\boldsymbol{x} - \boldsymbol{a}_i\|_2 \leq \tilde{d}_i\}$.

We can deduce that the non-empty feasible set $\mathcal{B}'$ definitely contains the target node position. It is clear that the intersection $\mathcal{B}'_i$ might be large, resulting in loose upper bounds. Note that if the measurement errors are not bounded from below, we are able to use a similar technique to end up an intersection that contains the target node location with some probability. For example, for the Gaussian noise we can consider a bounded interval in which noise samples reside with high probability. For instance for zero-mean Gaussian noise, if we define $\rho^i = 3\sigma$, noise samples belong to the set $\{\alpha \in \mathbb{R} : -3\sigma \leq \alpha \leq 3\sigma\}$ with probability $p = 0.9973$. Future investigations are needed to find approaches for efficiently dealing with the measurements when errors can be both positive and negative.

5. Geometric upper bounds

In this study, taking the assumption of positive measurement errors into account and considering the discussion in Section 2.3, we derive two different upper bounds. The first bound is derived based on the

availability of an estimate. If such an estimate is available (feasible or infeasible), we can bound it by finding the maximum distance between the estimate and any point in the feasible set. The second bound is derived without the need for an estimate, as the maximum length of the intersection set. Let us recall the definition of the position error

$$e \triangleq \|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2, \tag{22}$$

where $\hat{\boldsymbol{x}}$ is an estimate of the target node position given by a positioning algorithm. In a practical scenario it is not possible to compute the exact position error in (22) since the position of a target node is unknown. However, according to the discussion in Section 2.3, we can bound the position error, for a specific value of $\hat{\boldsymbol{x}}$, as

$$e \leq v_{\max,1} \triangleq \max_{\boldsymbol{x}\in\mathcal{B}} \ \|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2, \tag{23}$$

where $\mathcal{B}$ defines a set (closed bounded) in which the target node $\boldsymbol{x}$ belongs. In fact, definition (23) is a special case of the upper bound defined in (4) in Section 2.3 when $\mathcal{X}(\boldsymbol{m}) = \mathcal{B}$.

Alternatively, if an algorithm always produces one point in the feasible set $\mathcal{B}$ as an estimate, we are still able to define an upper bound on the position error, even without having access to an estimate, by setting $\mathcal{X}(\boldsymbol{m}) = \hat{\mathcal{X}}(\boldsymbol{m}) = \mathcal{B}$ in (6),

$$e \leq v_{\max,3} \triangleq \max_{\boldsymbol{x},\boldsymbol{y}\in\mathcal{B}} \ \|\boldsymbol{x} - \boldsymbol{y}\|_2. \tag{24}$$

5.1. A bound for the case an estimate exists

As mentioned in previous section, we can upper bound the position error due to an estimate $\hat{\boldsymbol{x}}$ (either feasible or infeasible), by solving the optimization problem (23). For example, let us consider Fig. 4 where an estimate $\hat{\boldsymbol{x}}$ of the target node position inside the intersection of three discs is available. The position error and an upper bound on the position error are shown in this figure. Instead of directly solving the problem in (23), we consider a QCQP problem $\mathrm{QP}\{\boldsymbol{A}_i, \boldsymbol{b}_i, c_i\}_{i=0}^N$, where

$$\boldsymbol{A}_i = \boldsymbol{I}_n, \qquad \boldsymbol{b}_i = \begin{cases} -\hat{\boldsymbol{x}}, & \text{if } i = 0, \\ -\boldsymbol{a}_i, & \text{otherwise,} \end{cases} \qquad c_i = \begin{cases} \|\hat{\boldsymbol{x}}\|^2, & \text{if } i = 0, \\ \|\boldsymbol{a}_i\|^2 - \hat{d}_i^2, & \text{otherwise.} \end{cases} \tag{25}$$

Obviously, $v_{\mathrm{qp}}\{\boldsymbol{A}_i, \boldsymbol{b}_i, c_i\}_{i=0}^N = v_{\max,1}^2$. The optimization problem in (23) is nonconvex which makes the problem complicated. To solve the problem, we employ a relaxation technique, which allows us to compute an upper bound with reasonable complexity. Following the procedures explained in Section 2.2, we can get a relaxed SDP problem as $\mathrm{SDP}\{\boldsymbol{A}_i, \boldsymbol{b}_i, c_i\}_{i=0}^N$ and the position error can be upper bounded as

$$e = \|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq v_{\max,1} \leq \sqrt{v_{\mathrm{sdp}}\{\boldsymbol{A}_i, \boldsymbol{b}_i, c_i\}_{i=0}^N}. \tag{26}$$

We call the bound derived in (26) as Upp-MaxDis. In order to investigate the tightness of the second inequality in (26), i.e., the gap between the relaxed SDP and the original QCQP, let us write the QCQP problem QP$\{\boldsymbol{A}_i, \boldsymbol{b}_i, c_i\}_{i=0}^N$ parameterized in (25) as

$$\begin{aligned} &\underset{\boldsymbol{x}\in\mathbb{R}^n,\ \tau\in\mathbb{R}}{\text{maximize}}\ \operatorname{tr}\left(\boldsymbol{B}\left[\boldsymbol{x}^T\ \tau\right]^T\left[\boldsymbol{x}^T\ \tau\right]\right) \\ &\text{subject to } \operatorname{tr}\left(\boldsymbol{B}_i\left[\boldsymbol{x}^T\ \tau\right]^T\left[\boldsymbol{x}^T\ \tau\right]\right) \le t_i,\quad i=1,\ldots,N+1, \end{aligned} \tag{27}$$

where

$$\begin{aligned} &\boldsymbol{B}_{N+1} = \begin{bmatrix} \mathbf{0}_{n\times n} & \mathbf{0}_n \\ \mathbf{0}_n^T & 1 \end{bmatrix},\quad \boldsymbol{B} = \begin{bmatrix} \boldsymbol{I}_n & -\hat{\boldsymbol{x}} \\ -\hat{\boldsymbol{x}}^T & \|\hat{\boldsymbol{x}}\|^2 \end{bmatrix},\quad \boldsymbol{B}_i = \begin{bmatrix} \boldsymbol{I}_n & -\boldsymbol{a}_i \\ -\boldsymbol{a}_i{}^T & \|\boldsymbol{a}_i\|_2^2+\epsilon^2 \end{bmatrix}, \\ &t_i = \hat{d}_i^2+\epsilon^2,\ i\le N,\quad t_{N+1}=1, \end{aligned} \tag{28}$$

where $\epsilon \neq 0$ is any nonzero real value. It is seen that $\boldsymbol{B}_i \succ 0$ for $1 \le i \le N$. Then, $\sum_{i=1}^{N+1} \boldsymbol{B}_i \succ 0$, meaning the interior of the feasible set is nonempty.

Proposition 5.1. *Consider the nonconvex QCQP problem parameterized in* (25) *with optimal value* $v_{\rm qp}\{\boldsymbol{A}_i, \boldsymbol{b}_i, c_i\}_{i=0}^N$. *Denote the optimal value of the corresponding relaxed problem, i.e, the SDP, as* $v_{\rm sdp}\{\boldsymbol{A}_i, \boldsymbol{b}_i, c_i\}_{i=0}^N$. *The optimal value of* QP$\{\boldsymbol{A}_i, \boldsymbol{b}_i, c_i\}_{i=0}^N$ *can be bounded from below as*

$$\alpha\, v_{\rm sdp}\{\boldsymbol{A}_i, \boldsymbol{b}_i, c_i\}_{i=0}^N \le v_{\rm qp}\{\boldsymbol{A}_i, \boldsymbol{b}_i, c_i\}_{i=0}^N, \tag{29}$$

where

$$\alpha = \frac{1}{2\ln(2(N+1)\mu)},\qquad \mu = \min\{N+1, n+1\}. \tag{30}$$

Proof. Considering $\operatorname{rank}(\boldsymbol{B}_i) = n+1$ and $\mu = \min\{N+1, \max_i \operatorname{rank}(\boldsymbol{B}_i)\} = \min\{N+1, n+1\}$, and then recalling the results of [38], which determines a lower bound on the optimal value of a QCQP based on its relaxed SDP, the Proposition is proved. □

The bound (23) is then bounded as

$$\sqrt{\alpha v_{\rm sdp}\{\boldsymbol{A}_i, \boldsymbol{b}_i, c_i\}_{i=0}^N} \le v_{\max,1} \le \sqrt{v_{\rm sdp}\{\boldsymbol{A}_i, \boldsymbol{b}_i, c_i\}_{i=0}^N}. \tag{31}$$

For details of deriving lower bounds on a nonconvex QCQP, we refer the reader to [24, 39, 38] and references therein.

5.2. Bound regarding the feasible set

In this section, we investigate the upper bound defined in (24) and repeated here for convenience

$$v_{\max,3} = \max\left\{\|\boldsymbol{x}-\boldsymbol{v}\|_2 :\ \boldsymbol{x},\boldsymbol{v}\in\mathcal{B}\right\}. \tag{32}$$

If a feasible point $\hat{\boldsymbol{x}} \in \mathcal{B}$ is available, it is expected that the first upper bound $v_{\max,1}$ yields a tighter bound compared to the bound defined in (32) (the maximum length of the intersection). In fact, for a fixed $\hat{\boldsymbol{x}} \in \mathcal{B}$,

$$\max_{\boldsymbol{x},\boldsymbol{w}\in\mathcal{B}} \|\boldsymbol{x}-\boldsymbol{w}\|_2 \geq \max_{\boldsymbol{x}\in\mathcal{B}} \|\boldsymbol{x}-\hat{\boldsymbol{x}}\|_2. \tag{33}$$

5.2.1. Bound based on the minimum ball covering the intersection:

The optimization problem in (32) is nonconvex and in general difficult to solve. Instead of solving the problem formulated in (32), we find a minimum ball covering the intersection $\mathcal{B}$. Let us consider the center $\boldsymbol{x}_c$ and the radius R of such a ball and formulate the minimum ball enclosing the intersection $\mathcal{B}$ in decision variables $\boldsymbol{x}_c$ and $\gamma = R^2$ as

$$\begin{aligned} &\underset{\boldsymbol{x}_c\in\mathbb{R}^n,\ \gamma\in\mathbb{R}_+}{\text{minimize}} \quad \gamma \\ &\text{subject to } \|\boldsymbol{x}-\boldsymbol{x}_c\|^2 \leq \gamma,\ \boldsymbol{x}\in\mathcal{B}. \end{aligned} \tag{34}$$

Let the optimal solution of (34) be $v'_{\max,3}$. Then, $v_{\max,3} = 2\sqrt{v'_{\max,3}}$. A sufficient condition for the constraint (34) to hold is that there are a number of positive values $\boldsymbol{\lambda} \in \mathbb{R}^N_+$ such that [24]

$$\|\boldsymbol{x}-\boldsymbol{x}_c\|^2 - \gamma \leq \sum_{i=1}^{N} \lambda_i (\|\boldsymbol{x}-\boldsymbol{a}_i\|^2 - \hat{d}_i^2). \tag{35}$$

We first introduce a fact for a quadratic function. For a proof of the claim in Lemma 5.2, see, e.g., [40, 24].

Lemma 5.2. *A quadratic function $f(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x} + 2\boldsymbol{b}^T\boldsymbol{x} + c$, with symmetric n by n matrix $\boldsymbol{A}$ is always nonnegative for all $\boldsymbol{x} \in \mathbb{R}^n$ if and only if*

$$\begin{bmatrix} \boldsymbol{A} & \boldsymbol{b} \\ \boldsymbol{b}^T & c \end{bmatrix} \succeq 0. \tag{36}$$

Applying the result of Lemma 5.2 to (35) and fixing $\boldsymbol{x}_c$ in (34), we can get the following optimization problem to find the minimum ball enclosing the intersection $\mathcal{B}$:

$$\begin{aligned} &\underset{\gamma\in\mathbb{R}_+}{\text{minimize}} \quad \gamma \\ &\text{subject to } \begin{bmatrix} (\sum_{i=1}^N \lambda_i - 1)\boldsymbol{I}_n & \boldsymbol{x}_c - \sum_{i=1}^N \lambda_i \boldsymbol{a}_i \\ (\boldsymbol{x}_c - \sum_{i=1}^N \lambda_i \boldsymbol{a}_i)^T & \gamma - \|\boldsymbol{x}_c\|_2^2 + \sum_{i=1}^N \lambda_i (\|\boldsymbol{a}_i\|_2^2 - \hat{d}_i^2) \end{bmatrix} \succeq 0, \\ &\boldsymbol{x}_c \in \mathbb{R}^n, \quad \boldsymbol{\lambda} \in \mathbb{R}^N_+. \end{aligned} \tag{37}$$

From the properties of a positive semidefinite matrix (nonnegative diagonal entries), we conclude that

$$\sum_{i=1}^{N} \lambda_i \geq 1 \tag{38}$$

and

$$\gamma - \|\boldsymbol{x}_c\|_2^2 + \sum_{i=1}^{N} \lambda_i (\|\boldsymbol{a}_i\|_2^2 - \hat{d}_i^2) \geq 0. \tag{39}$$

The constraint in (37) using Schur complement [40] can also be expressed as

$$\gamma - \|\boldsymbol{x}_c\|_2^2 + \sum_{i=1}^{N} \lambda_i (\|\boldsymbol{a}_i\|_2^2 - \hat{d}_i^2) - \frac{\|\boldsymbol{x}_c - \sum_{i=1}^{N} \lambda_i \boldsymbol{a}_i\|_2^2}{\sum_{i=1}^{N} \lambda_i - 1} \geq 0. \tag{40}$$

Hence, the problem in (37) can be expressed as

$$\begin{aligned} \underset{\gamma \in \mathbb{R}_+,\ \boldsymbol{\lambda} \in \mathbb{R}_+^N}{\text{minimize}} \ & \|\boldsymbol{x}_c\|_2^2 - \sum_{i=1}^{N} \lambda_i (\|\boldsymbol{a}_i\|_2^2 - \hat{d}_i^2) + \frac{\|\boldsymbol{x}_c - \sum_{i=1}^{N} \lambda_i \boldsymbol{a}_i\|_2^2}{\sum_{i=1}^{N} \lambda_i - 1} \\ \text{subject to } & \boldsymbol{x}_c \in \mathbb{R}^n, \ \sum_{i=1}^{N} \lambda_i \geq 1. \end{aligned} \tag{41}$$

Taking similar steps as done in [41], it can be shown that for the optimal solution $\sum_{i=1}^{N} \lambda_i = 1$ and

$$\boldsymbol{x}_c = \sum_{i=1}^{N} \lambda_i \boldsymbol{a}_i. \tag{42}$$

Hence, from (39) we can obtain a convex optimization problem to find an upper bound on the squared radius of the smallest ball enclosing the set $\mathcal{B}$ in the Euclidian norm sense as

$$\begin{aligned} \underset{\boldsymbol{\lambda} \in \mathbb{R}_+^N}{\text{minimize}} \ & \|\sum_{i=1}^{N} \lambda_i \boldsymbol{a}_i\|_2^2 - \sum_{i=1}^{N} \lambda_i (\|\boldsymbol{a}_i\|_2^2 - \hat{d}_i^2) \\ \text{subject to } & \sum_{i=1}^{N} \lambda_i = 1. \end{aligned} \tag{43}$$

Finally, an upper bound on the maximum length of $\mathcal{B}$ is given by

$$v_{\max,3} \leq 2R, \tag{44}$$

where $R = \sqrt{\|\sum_{i=1}^{N} \lambda_i \boldsymbol{a}_i\|_2^2 - \sum_{i=1}^{N} \lambda_i (\|\boldsymbol{a}_i\|_2^2 - \hat{d}_i^2)}$.

It has been proved in [41] that when the number of constraints N (here the number of reference nodes) is equal to or less than n (the dimensionality), (43) gives the optimal solution to (34). Otherwise, when $N > n$, which is the practical case for positioning, the optimal solution in (43) is an upper bound to the optimal solution in (34). The upper bound obtained by solving (43) then gives the maximum Euclidian length of the intersection. We call this bound as Upp-MinBall.

Another approach to compute an upper bound on $v_{\max,3}$ is to replace $\mathcal{B}$ with an enclosing set in (24). We will in the following consider two such sets. The first enclosing set is the bounding box[3] for $\mathcal{B}$, and, given

[3]By the bounding box of the set $\mathcal{A}$, we mean the smallest cuboid [42] that is enclosing $\mathcal{A}$.

the bounding box, it is very easy to compute an upper bound on $v_{\max,3}$, see Fig. 6. The second enclosing set is found by replacing $\mathcal{B}_i$ with their bounding boxes.

In the next sections, we investigate upper bounds on position error based on bounding box approaches.

5.2.2. Bound based on bounding box covering the intersection

To compute the bounding box for $\mathcal{B}$, we study the following optimization problem:

$$\begin{aligned} &\text{maximize } \|\boldsymbol{x}-\boldsymbol{y}\|_\infty \\ &\text{subject to } \boldsymbol{x},\boldsymbol{y}\in\mathcal{B}. \end{aligned} \tag{45}$$

The optimization problem in (45) again is nonconvex. Using the definition of the ℓ_∞ norm, we can write

$$\begin{aligned} &\underset{\boldsymbol{x},\boldsymbol{y}}{\text{maximize}} \;\; \max(|x_1-y_1|,\ldots,|x_n-y_n|) \\ &\text{subject to} \quad \boldsymbol{x},\boldsymbol{y}\in\mathcal{B}. \end{aligned} \tag{46}$$

Using a dummy variable β, we have

$$\max\{\alpha_1,\ldots,\alpha_n\}\geq\beta \Longleftrightarrow \alpha_1\geq\beta \text{ or } \alpha_2\geq\beta \;\ldots\; \text{or } \alpha_n\geq\beta. \tag{47}$$

Thus, using a simple technique, we need to solve two optimization problems for every dimension $\ell=1,\ldots,n$ as follows:

$$\begin{aligned} &\underset{\beta\in\mathbb{R},\ \boldsymbol{x}\in\mathbb{R}^n}{\text{maximize}} \;\; \beta \\ &\text{subject to} \;\; \|\boldsymbol{x}-\boldsymbol{a}_i\|\leq\hat{d}_i, \qquad i=1,\ldots,N, \\ &\qquad\qquad x_\ell\geq\beta, \end{aligned} \tag{48a}$$

$$\begin{aligned} &\underset{\beta\in\mathbb{R},\ \boldsymbol{x}\in\mathbb{R}^n}{\text{minimize}} \;\; \beta \\ &\text{subject to} \;\; \|\boldsymbol{x}-\boldsymbol{a}_i\|\leq\hat{d}_i, \qquad i=1,\ldots,N \\ &\qquad\qquad x_\ell\leq\beta. \end{aligned} \tag{48b}$$

The optimization problems in (48a) or (48a) is a second order cone program. Suppose that the optimal solution to problems (48a) and (48b) along a dimension ℓ are $x^*_{\ell_1}$ and $x^*_{\ell_2}$, respectively. Let the maximum length for the ℓth dimension be $v_{\text{socp},\ell}=|x^*_{\ell_1}-x^*_{\ell_2}|$. Then, the maximum length of the intersection can be upper bounded as

$$v_{\text{bbox}}=\sqrt{\sum_{i=1}^{n}(v_{\text{socp},\ell})^2}. \tag{49}$$

Thus

$$v_{\max,3} \leq v_{\text{bbox}}. \tag{50}$$

We call the bound obtained in this section as Upp-BBox

5.2.3. Bound based on the maximum length of the intersection of a number of bounding boxes

In the second approach, the ℓ_2 balls in (12) are replaced by the corresponding ℓ_∞ balls,

$$\mathcal{B}_i{}' \triangleq \{\boldsymbol{x} \in \mathbb{R}^n : \|\boldsymbol{x} - \boldsymbol{a}_i\|_\infty \leq \hat{d}_i\}, \quad i = 1, 2, \ldots, N,$$

and noting that

$$\mathcal{B} \subseteq \mathcal{B}' \triangleq \bigcap_{i=1}^{N} \mathcal{B}_i{}'.$$

Hence, an upper bound to $v_{\max,3}$ is found by considering the length of $\mathcal{B}'$, see Fig. 7.

To compute an upper bound on $v_{\max,3}$ based on $\mathcal{B}'$, we consider the following optimization problem:

$$\begin{aligned} &\underset{\boldsymbol{x},\boldsymbol{y}}{\text{maximize}} \, \|\boldsymbol{x} - \boldsymbol{y}\|_\infty \\ &\text{subject to } \boldsymbol{x}, \boldsymbol{y} \in \mathcal{B}', \end{aligned} \tag{51}$$

For example Fig. 7 shows the concept of relaxing the constraint for a 2-dimensional network. Following the same procedure to obtain (48), we obtain two optimization problems, called linear programs (LPs), for every dimension. For instance, the two LPs for the ℓth dimension can be written as (where $a_{i,\ell}$ is the ℓ-th element of $\boldsymbol{a}_i$)

$$\begin{aligned} &\underset{t_\ell \in \mathbb{R}}{\text{maximize}} \; t_\ell \\ &\text{subject to} \; \; t_\ell - a_{i,\ell} - \hat{d}_i \leq 0, \\ &\qquad\qquad t_\ell - a_{i,\ell} + \hat{d}_i \leq 0, \quad i = 1, \ldots, N, \end{aligned} \tag{52a}$$

$$\begin{aligned} &\underset{t_\ell \in \mathbb{R}}{\text{minimize}} \; t_\ell \\ &\text{subject to} \; \; t_\ell - a_{i,\ell} - \hat{d}_i \leq 0, \\ &\qquad\qquad a_{i,\ell} - t_\ell + \hat{d}_i \leq 0, \quad i = 1, \ldots, N. \end{aligned} \tag{52b}$$

The optimal solution to the optimization problems (52a) and (52b), i.e., $t^*_{\ell_1}$ and $t^*_{\ell_2}$, are simply computed as

$$t^*_{\ell_1} = \min\{a_{1,\ell} + \hat{d}_1, \ldots, a_{N,\ell} + \hat{d}_N\}, \quad t^*_{\ell_2} = \max\{a_{1,\ell} - \hat{d}_1, \ldots, a_{N,\ell} - \hat{d}_N\}. \tag{53}$$

Let $v_{\mathrm{lp},\ell} = |t^*_{\ell_1} - t^*_{\ell_2}|,\ \ell = 1, \ldots, n$, be the maximum length along the ℓth dimension. The maximum length of the intersection $\mathcal{B}$ is then upper bounded by

$$v_{\mathrm{rbbox}} = \sqrt{\sum_{i=1}^{n} (v_{\mathrm{lp},\ell})^2}. \tag{54}$$

Therefore, an upper bound on position error based on a bounding box approach is given by

$$v_{\max,3} \leq v_{\mathrm{rbbox}}. \tag{55}$$

We call this relaxed bounding box approach as Upp-RBBox.

It is clear that $v_{\mathrm{bbox}} \leq v_{\mathrm{rbbox}}$. The Upp-RBBox is the loosest bound investigated in this study, but it is easy to compute.

Table 1 summarizes the various types of bounds derived in this study.

6. Simulation results

In this section we evaluate the validity of different upper bounds through computer simulations for different scenarios. We consider a 1000 m^3 cubic space for simulation. N reference nodes, i.e., $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_N$, are placed at fixed positions according to Table 2. One target node is randomly placed inside the volume. We also consider a 2D network and obtain confidence regions (based on an estimate and an upper bound) for positions of a moving target. We generate the noisy distance as

$$\hat{d}_i = d_i(\boldsymbol{x}, \boldsymbol{a}_i) + \varepsilon_i,$$

where

$$\varepsilon_i = \begin{cases} \varepsilon_{E,i}, & \text{LOS conditions} \\ \varepsilon_{E,i} + b_i \varepsilon_{U,i}, & \text{NLOS conditions} \end{cases}$$

where $\varepsilon_{E,i}$ are i.i.d exponential random variables with mean 1, $b_i \in \{0, 1\}$ are iid Bernoulli random variables with parameter $p_{\mathrm{NLOS}} = \Pr\{b_i = 1\}$, and $\varepsilon_{U,i} \sim \mathcal{U}(0, r)$ are iid uniform random variables. For NLOS, we set $r = 10$ m and assume 20% of measurements are NLOS. In every scenario, we generate 1000 networks. The validity of exponential distribution for LOS measurement for UWB practical data has been justified in recent work [22]. The validity of the uniform distribution for NLOS has been clarified by some authors, e.g., [43, 27].

We consider the POCS, LLS (a closed-form linear least squares followed by a correction technique [10, 11]), and SDP algorithms to estimate the target node positions. POCS always gives an estimate inside the

intersection $\mathcal{B}$ in (13) . To solve the optimization problems formulated in this study (and also SDP for localization), we use the *CVX* toolbox [44].

To evaluate the tightness of the bounds in Table 1, we define the tightness of the bound as $t_v \triangleq (v - e)$ for a bound v and the true position error e. We also define the relative tightness as $\tau_v \triangleq (v - e)/e$. To illustrate how the tightness varies with, e.g., network deployment, measurement noise, estimator parameters, we study the cumulative distribution function (CDF) of t_v and τ_v, i.e., $\Pr\{t_v \leq x\}$ and $\Pr\{\tau_v \leq x\}$, where the randomness comes from selecting, e.g., the deployment in a random fashion. In the following, we will generate e from POCS, CLS, or SDP.

6.1. Line-of-sight

In this section, we evaluate the performance of different bounds for LOS scenarios. To evaluate the four bounds in Table 1, we employ the POCS algorithm since it always gives a feasible estimate (which is required for Upp-MinBall, Upp-BBox, and Upp-RBBox). We also assess Upp-MaxDis for other estimators later in this section.

Fig. 8 shows the CDF of the relative tightness of the upper bounds versus POCS position error for different number of reference nodes, N. As expected, Upp-MaxDis shows better performance compared to the other bounds. For instance, Fig. 8(a) shows that in 80% of the cases, Upp-MaxDis in a network consisting of eight reference nodes is less than 2 times the actual position error (considering the relative tightness). This figure also shows, as expected, that Upp-RBBox is the loosest bound. When the number of reference nodes increases, Upp-BBox gets closer to Upp-MinBall. Roughly speaking except for Upp-BBox, we can say that the behavior of other upper bounds (based on the relative tightness $\tau_v = (v-e)/e$) does not change considerably with increasing the number of reference nodes. For large number of reference nodes the bounding box would be close to the minimum ball enclosing the intersection and we solve the bounding box approach exact while for the minimum ball approach, we approximately solve it. That is the reason why Upp-BBox shows better performance than Upp-MaxDis for large number of reference nodes.

In the sequel, we evaluate the tightness of Upp-MaxDis for the LLS and SDP estimators. Fig. 9 shows the CDF of the relative tightness for different algorithms. From this figure we see that Upp-MaxDis for the POCS estimate is tighter than the corresponding bounds for SDP and LLS estimates. In fact, for the same intersection (same data), we expect the relative tightness to degrade when the position error decreases, which explains why SDP and LLS have worse relative tightness compared to POCS. Although the bounds on SDP and LLS estimates are not very tight in the relative sense, they can still provide valuable information. For instance, Upp-MaxDis computed from LLS estimate in 80% of the cases is less than 3.7 times the actual

LLS position errors. It is also seen that the increasing the number of reference nodes improves the tightness of bounds for LLS and SDP estimates.

To evaluate the bound on average, we study the average tightness defined as $\bar{t}_v \triangleq \mathbb{E}\, t_v$. In Fig. 10, we plot the average tightness $\bar{t}_v$ versus the number of reference nodes for different algorithms. As seen, $\bar{t}_v$ decreases with increasing number of reference nodes.

In the next simulation, we evaluate a confidence region (based on an estimate of the target position and an upper bound on the position error) definitely containing the position of a moving target. We consider a 2D network in which a number of reference nodes are placed a long with lines $y = 0$ and $y = 100$ (Fig. 11). We assume the same distribution of the measurement noise as considered in the previous simulations. We run the POCS algorithm and obtain an estimate of the moving target position at different locations. The target moves on a trajectory according to a quadratic curve $y = -0.0004x^2 + 0.3x + 10$ in the xy plane (see the red curve in Fig. 11). We then obtain an upper bound using Eqn.(26), i.e., Upp-Maxdis. The discs formed by the estimate and the corresponding upper bound definitely contains the location of the target node as shown in Fig. 11. It is observed that in some positions, the corresponding discs are small due to small intersections containing the location of the target node positions. In some positions the bound may not be tight, resulting a large confidence disc. In general, we can conclude that the bound in this scenario provides useful information about the location of the target node.

6.2. Non-line-of-sight

In this section, we evaluate Upp-Maxdis in NLOS scenarios for different algorithms. In Fig. 12, we plot the CDF of the relative tightness fo Upp-Maxdis when estimates from POCS, LLS, and SDP are available. Comparing Fig. 9 and Fig. 12, we see that the relative tightness of Upp-Maxdis for SDP and LLS is better in the NLOS case compared to the LOS case, which can be expected since the LLS and SDP errors are larger for in NLOS conditions compared to the LOS case. It is also seen that the relative tightness of the bound for LLS and SDP improves as the number of reference nodes increases.

To have a graphical view about the tightness of Upp-Maxdis for different algorithms, we plot 1000 samples of the position error (for 1000 realizations of the network) and the corresponding Bound 1, when 14 reference nodes are considered, in Fig. 13. It is seen that the tightness of the bounds vary with the accuracy of the estimate. The average tightness $\bar{t}_v$ for every approach is also mentioned in the figures. From these figures we again see that the bound for POCS is tighter that the ones for SDP and LLS.

6.3. Gaussian error

As mentioned in Remark 4, when the measurement errors are negative, the intersection does not contain the target node location. We may increase the distance estimates to obtain a new intersection containing the location of the target node. In this simulation, we consider the Gaussian distribution for the measurement errors. We considered 10 reference nodes and the measurement errors are modeled by $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$ with $\sigma = 1$ [m]. We randomly distribute a target node inside the network area. The results are shown in figure 14. Note that CVX returns +Inf for empty intersection. It is observed that when the distance estimates are increased by a small value, the bound may not be valid or the intersection does not contain the target location. For example in this figure, when distance estimates are enlarged by σ, in 10% of the time the intersection is empty, so the bound is not known, and in 20% of the time the bound is not valid. In addition, when a large value, e.g., 4σ, are added to the measurements, the intersection becomes nonempty with hight probability and also the bound is valid with high probability, but the tightness might be questionable. From these observation, we see a tradeoff between the tightness and the validity should be considered to design an approach for Gaussian distribution based on manipulating the distance estimate.

Based on numerical results and theoretical evidences, we can summarize the effect of geometry and the accuracy of the estimate on the tightness and relative tightness as follows:

- for a relatively fixed intersection, both tightness and relative tightness degrade with the accuracy of the estimation;
- for a relatively fixed estimation error, both tightness and relative tightness degrade with increasing (the volume of) the intersection;
- in general, the accuracy of the estimation also depends on the shape (volume) of the intersection. Therefore, the relation between tightness and relative tightness with the volume of the intersection and the accuracy of the estimation can be complicated.

7. Conclusions and future studies

In this paper we have formulated a number of upper bounds on the realization of the positioning error, i.e., the error which is produced by an estimator, or a class of estimators, given a certain realization of the measurements. The idea is that the target node position is first limited to closed (possibly convex) set and then an upper bound on the position error is defined with respect to the set. For instance, in a range-based positioning approach with positive distance measurement errors, the target node position is confined to the intersection of a number of balls derived from measurements. We note that non-negative distance errors

are likely to occur in non-line-of-sight environments and also in some line-of-sight scenarios [22]. We have then studied two classes of (geometric) upper bounds with respect to the feasible set confining the target node position. If an estimate of the target position is available, we have defined an upper bound on the position error as the maximum distance between the estimate and every point in the intersection area (the first bound). The resulting bound is formulated as a (difficult) nonconvex problem and we have employed a relaxation technique to solve the problem.

Assuming a closed bounded feasible set, a positioning algorithm such as POCS can be designed to give a point inside the intersection area. Assuming an algorithm always gives a feasible point, we have formulated an upper bound independent of the estimate as the largest distance between two points in the intersection area, i.e., maximum length of the intersection area (the second bound). The corresponding bound is formulated as a nonconvex problem and can be approximately solved. We have also investigated variants of the second bound by replacing either the intersection area or each ball by a corresponding bounding box.

Simulation results based on the POCS estimate for different situations show that the proposed upper bounds (summarized in Table 1) provide reasonably tight bounds. As expected from the theoretical part and confirmed by the simulation results, Upp-Maxdis is the tightest bound among different upper bounds formulated in this paper. The numerical results also show that the relative tightness of the different bounds, except Upp-BBox does not considerably change with node density. It is also concluded from both theoretical aspects and simulation results that Upp-MinBall and Upp-BBox are tighter than Upp-RBBox. There might still be motivated to used Upp-RBBox, e.g., if we care more about the computational complexity than tightness of the bound. Numerical results for other positioning algorithms based on LLS and SDP also show reasonable tightness in some scenarios.

Finally, it is clear that it is very valuable if we, in a practical situation, can append an estimated position with an upper bound of the position error. This is much stronger than saying something about the statistics of the position error (e.g., the mean squared error) if the bound has reasonable tightness. The methods developed in this paper provides tools for bounding the position error, albeit in somewhat limited situations, i.e., when the feasible set is nonempty and has finite length. There are practical situations where this is a valid assumption, but also cases when it is not.

Future work includes formulating bounds for a general case when distance errors can be both positive and negative and extending the bounding techniques for cooperative positioning problems. Of course, it would also be of value to improve the tightness of the bounds. For example by involving the prior information about the location of the target node, if available, we may be able to improve the tightness of the bound. One interesting and challenging problem is to investigate upper bounds for cooperative positioning scenarios.

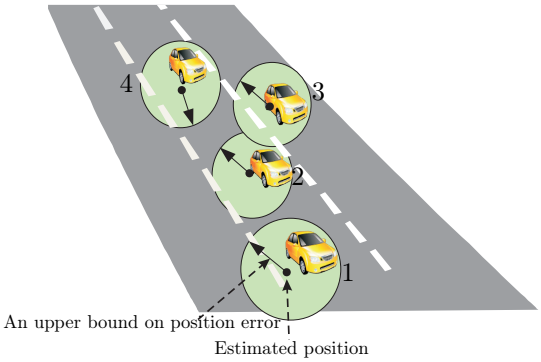


Figure 1: An example of the application of an upper bound on the position error for traffic safety. A solid circle defines the area in which a vehicle definitely lies. In this figure based on an upper bound on the position error, car 2 and 3 might collide.

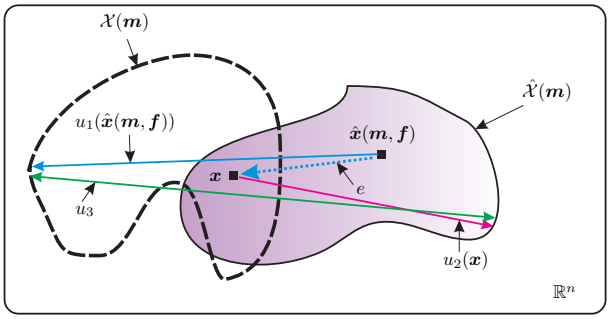


Figure 2: Different upper bounds.

8. acknowledgment

Authors would like to thank Prof. Stephen P. Boyd and Prof. Arkadi Nemirovski for comments on the optimization problems considered in this paper. They also would like to thanks Dr. Sinan Gezici for comments on the paper. The simulations were performed on resources provided by the Swedish National Infrastructure for Computing (SNIC) at C3SE.

References

[1] G. Mao, B. Fidan, Localization Algorithms and Strategies for Wireless Sensor Networks, Information Science reference, Hershey. New York, 2009.

[2] L. Doherty, K. S. J. Pister, L. E. Ghaoui, Convex position estimation in wireless sensor networks, in: INFOCOM, Vol. 3, 2001, pp. 1655–1663.

[3] A. H. Sayed, A. Tarighat, N. Khajehnouri, Network-based wireless location: challenges faced in developing techniques for accurate wireless location information, IEEE Signal Process. Mag. 22 (4) (2005) 24–40.

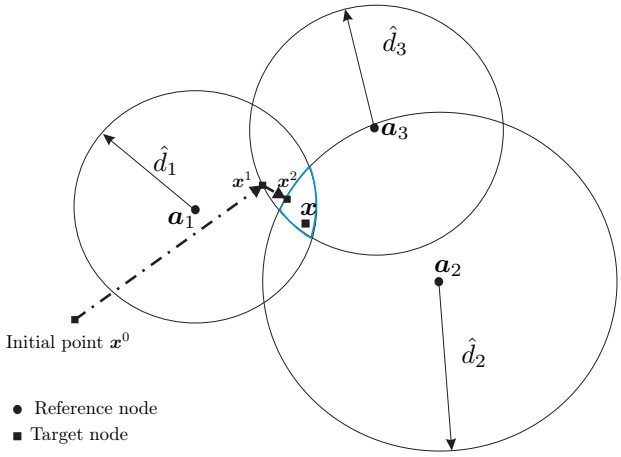


Figure 3: A 2-dimensional network consisting of three reference nodes and one target node. For nonnegative measurement errors, the target node at position $\boldsymbol{x}$ is found in the intersection of three discs. The POCS estimate converges to a point $\hat{\boldsymbol{x}}$ inside the intersection area (in this case on the boundary).

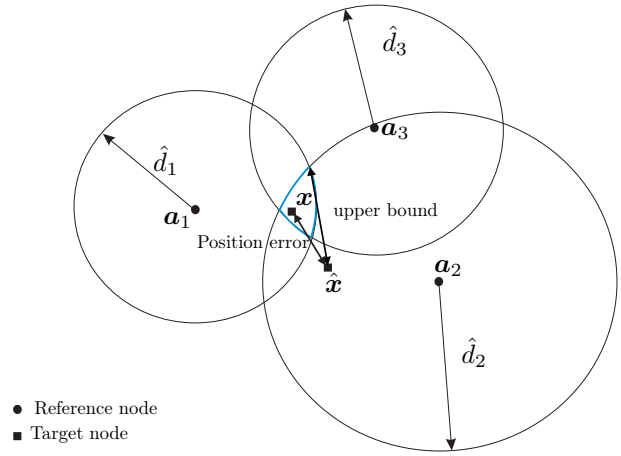


Figure 4: The position error and an upper bound on the position error for an estimate $\hat{\boldsymbol{x}}$ of the target for the network considered in Fig. 3.

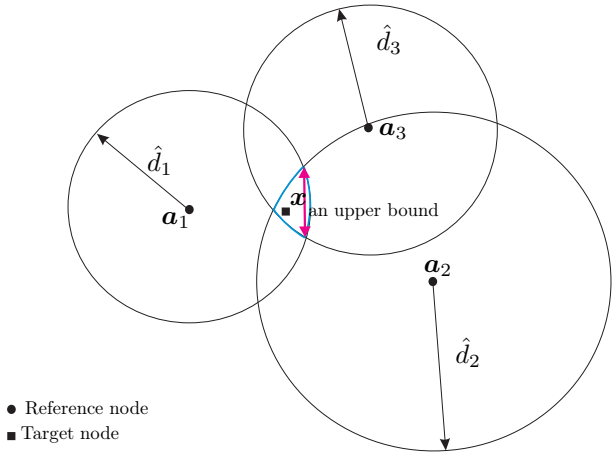


Figure 5: Maximum Euclidian distance of the intersection as an upper bound on the position error for an estimate inside the intersection area of Fig. 3.

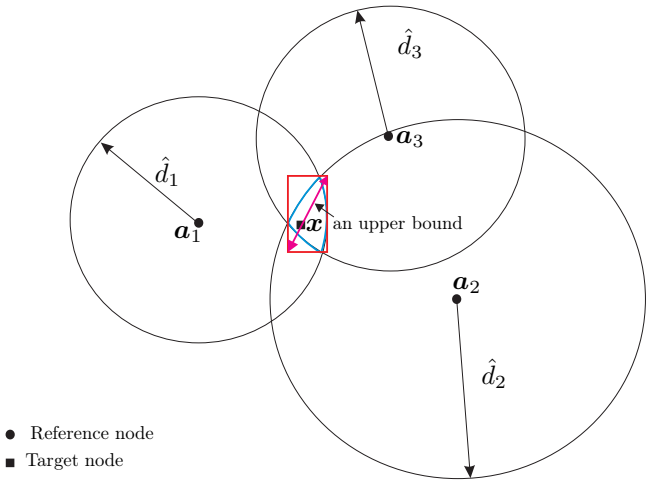


Figure 6: The maximum length of the bounding box of the intersection as an upper bound on the position error for the estimate considered in Fig. 3.

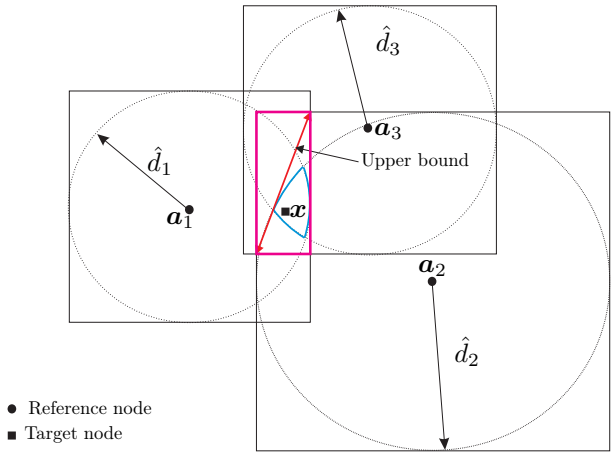


Figure 7: Every constraint is replaced with a bounding box and then a bounding box enclosing the intersection of relaxed constraints is computed. The maximum length of the bounding box enclosing the intersection gives an upper bound for the position error for the estimate considered in Fig. 3.

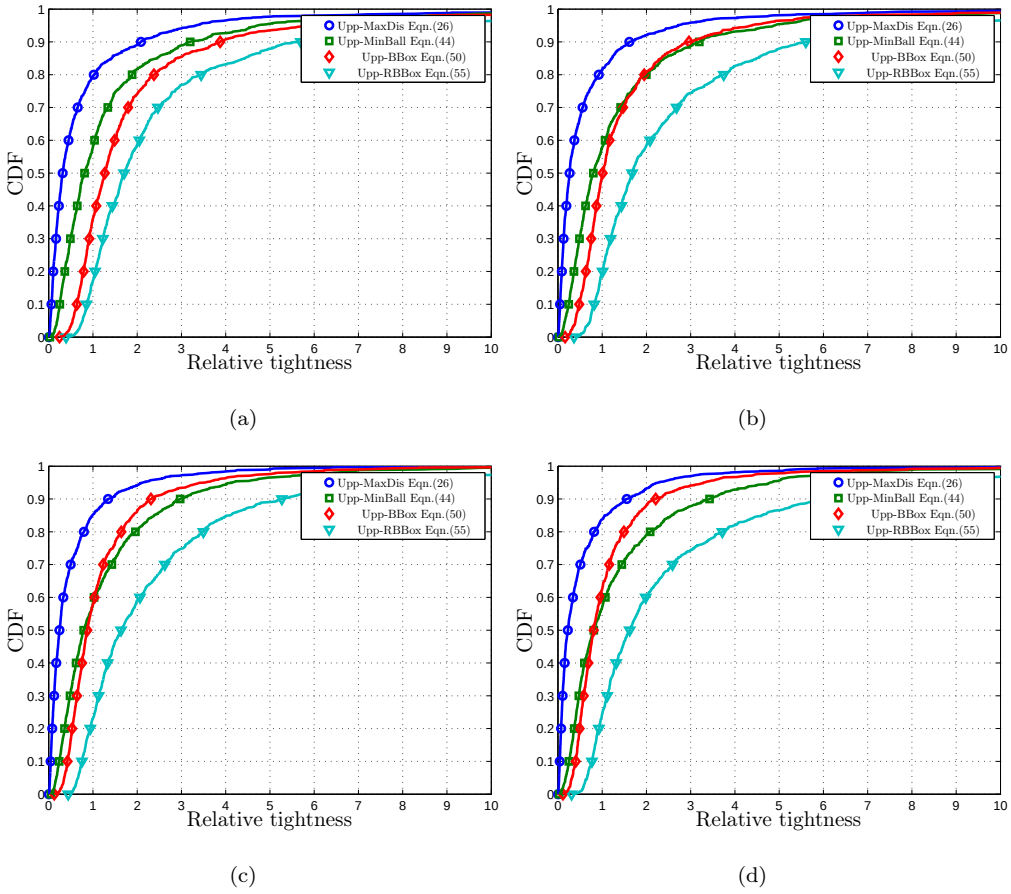


Figure 8: Comparison between the CDF of relative tightness of upper bounds versus the POCS position error for, (a) 8 reference nodes, (b) 14 reference nodes, (c) 20 reference nodes, and (d) 26 reference nodes.

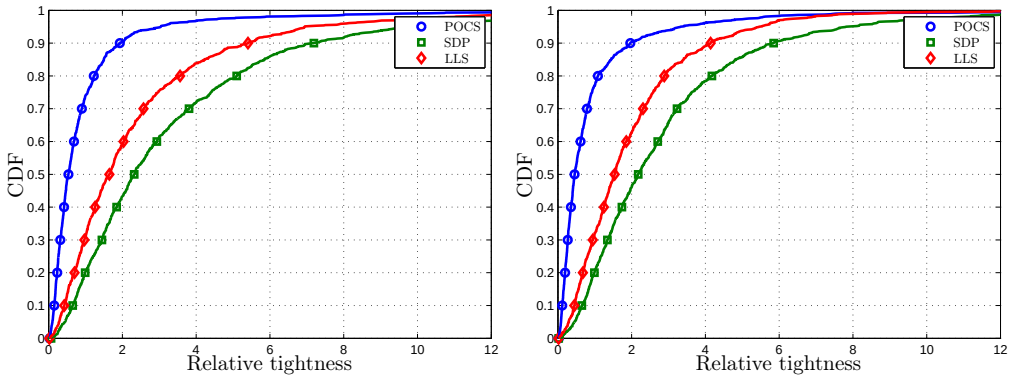


Figure 9: CDF of the relative tightness for different algorithms in LOS scenario, (a) 14 reference nodes, (b) 26 reference nodes.

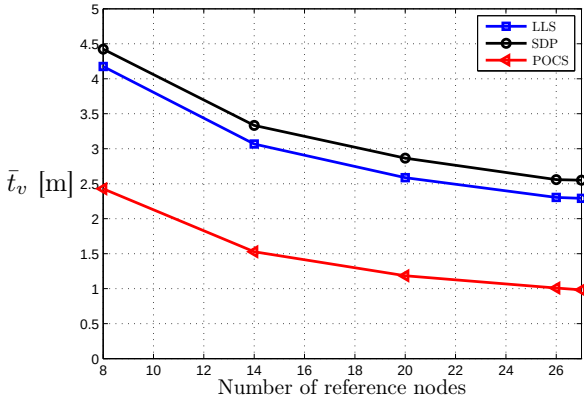


Figure 10: The average tightness $\bar{t}_v$ versus the number of reference nodes for different algorithms.

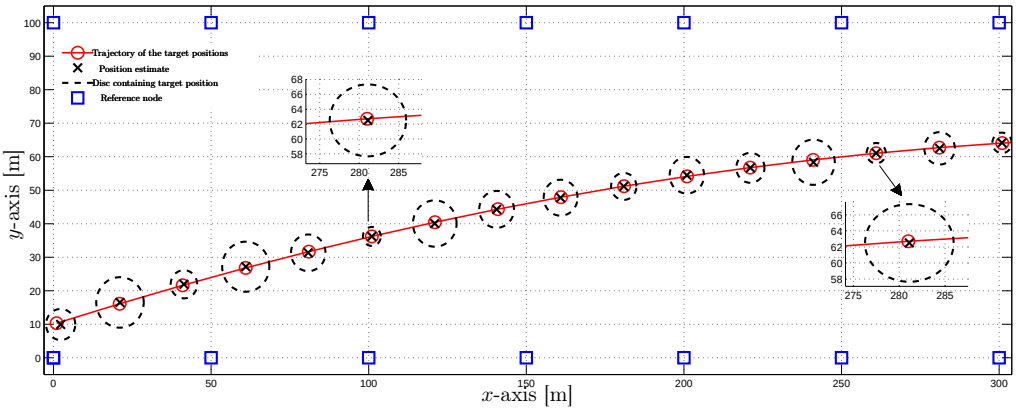
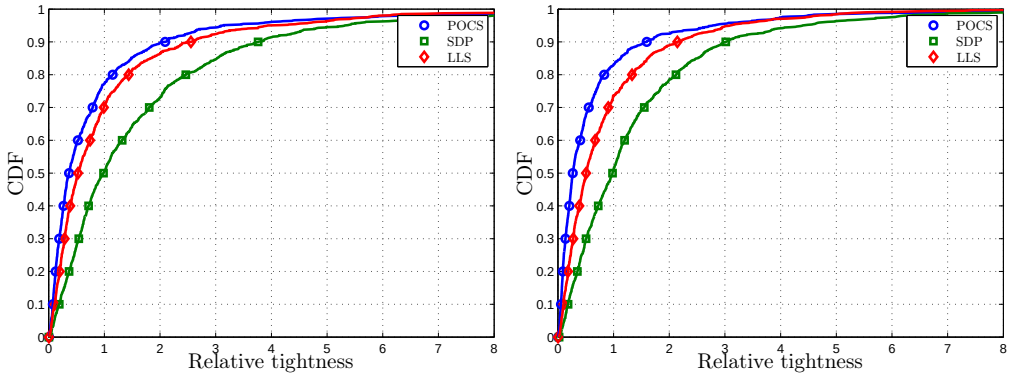
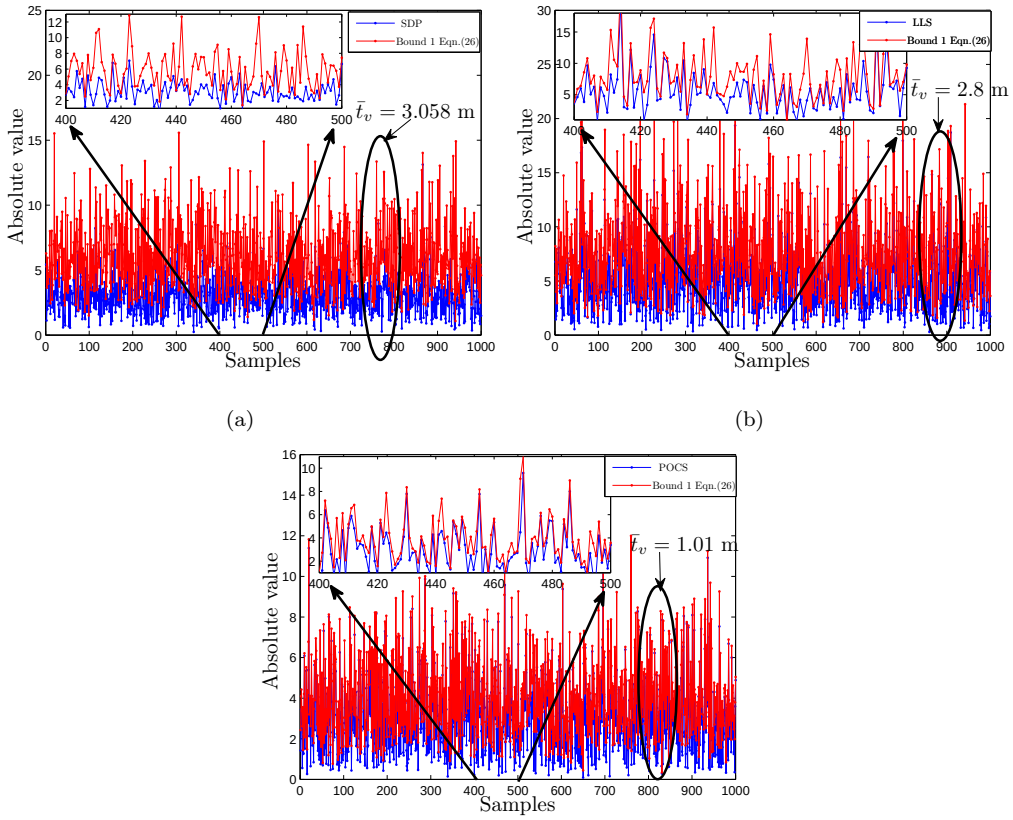


Figure 11: Discs, formed by an estimate of the target position and the corresponding upper bound (Bound 1 Eqn.(26)), contains the location of a moving target according to a trajectory.



(a) (b)

Figure 12: CDF of the normalized relative tightness of Bound1 for different algorithms in NLOS scenario, (a) 8 reference nodes and (b) 14 reference nodes.



(a) (b)

(c)

Figure 13: Absolute values of Bound 1 and position errors for three algorithms in NLOS scenario for 14 reference nodes for (a) SDP, (b) LLS, and (c) POCS.

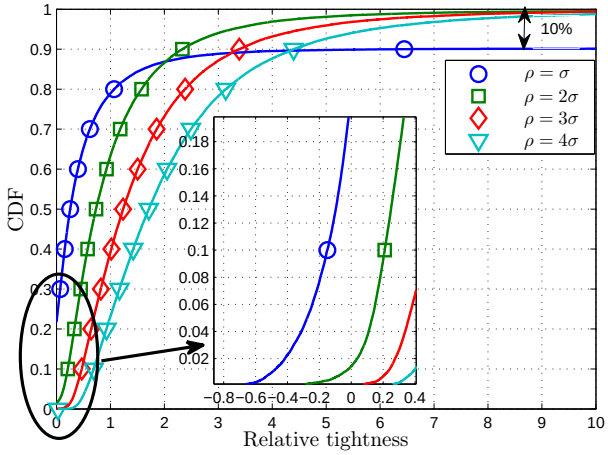


Figure 14: The CDF of the relative tightness of Upp-MaxDis versus the POCS position error for 10 reference nodes and different lower bounds ρ of the Gaussian measurement errors ($\sigma = 1$ [m]).

Table 1: Summary of bounds.

Definition:	Eqn.
$e \triangleq \|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2$	(22)
$v_{\max,1} \triangleq \max\limits_{\boldsymbol{x}\in\mathcal{B}} \|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_2$	(23)
$v_{\max,3} \triangleq \max\limits_{\boldsymbol{x},\boldsymbol{y}\in\mathcal{B}} \|\boldsymbol{x} - \boldsymbol{y}\|_2$	(24)
Upper Bounds:	**Eqn.**
Upp-MaxDis:	
$e \leqslant v_{\max,1} \leq \sqrt{v_{\text{sdp}}\{\boldsymbol{A}_i, \boldsymbol{b}_i, c_i\}_{i=0}^{N}}$	(26)
Upp-MinBall:	
$v_{\max,3} \leq 2R$	(44)
Upp-BBox:	
$v_{\max,3} \leq v_{\text{bbox}}$	(50)
Upp-RBBox:	
$v_{\max,3} \leq v_{\text{rbbox}}$	(55)

Table 2: Positions of reference nodes in 3D network.

$\boldsymbol{a}_i$ \ i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
x [m]	0	10	0	0	10	10	0	10	5	5	0	10	5	5	5	5	5	0	10	0	0	10	0	10	10	5	5
y [m]	0	0	10	0	10	0	10	10	5	5	5	5	0	10	0	10	0	0	0	10	5	5	5	10	5	10	5
z [m]	0	0	0	10	0	10	10	10	0	10	5	5	5	5	0	0	10	5	5	5	0	0	10	5	10	10	5

[4] S. Gezici, A survey on wireless position estimation, Wireless Personal Communications 44 (3) (2008) 263–282.

[5] S. Gezici, Z. Tian, G. B. Giannakis, H. Kobayashi, A. F. Molisch, H. V. Poor, Z. Sahinoglu, Localization via ultra-wideband radios: A look at positioning aspects for future sensor networks, IEEE Signal Process. Mag. 22 (4) (2005) 70–84.

[6] R. Huang, G. V. Zaruba, Beacon deployment for sensor network localization, in: IEEE Wireless Communications and Networking Conference, 2007, pp. 3188–3193.

[7] N. Bulusu, J. Heidemann, D. Estrin, GPS-less low-cost outdoor localization for very small devices, IEEE Personal Commun. 7 (5) (2000) 28–34.

[8] M. R. Gholami, Positioning algorithms for wireless sensor networks, Licentiate Thesis, Chalmers University of Technology (Mar. 2011).
URL http://publications.lib.chalmers.se/records/fulltext/138669.pdf

[9] P. Biswas, T.-C. Lian, T.-C. Wang, Y. Ye, Semidefinite programming based algorithms for sensor network localization, ACM Trans. Sens. Netw. 2 (2) (2006) 188–220.

[10] M. Sun, K. C. Ho, Successive and asymptotically efficient localization of sensor nodes in closed-form, IEEE Trans. Signal Process. 57 (11) (2009) 4522–4537.

[11] M. R. Gholami, S. Gezici, E. G. Ström, Improved position estimation using hybrid TW-TOA and TDOA in cooperative networks, IEEE Trans. Signal Process. 60 (7) (2012) 3770–3785.

[12] D. Blatt, A. O. Hero, Energy-based sensor network source localization via projection onto convex sets, IEEE Trans. Signal Process. 54 (9) (2006) 3614–3619.

[13] A. O. Hero, D. Blatt, Sensor network source localization via projection onto convex sets (POCS), in: Proc. IEEE International Conference on Acoustics, Speech and Signal Processing, Vol. 3, Philadelphia, USA, 2005, pp. 689–692.

[14] M. R. Gholami, H. Wymeersch, E. G. Ström, M. Rydström, Wireless network positioning as a convex feasibility problem, EURASIP Journal on Wireless Communications and Networking 2011 2011:161.

[15] Y. Censor, S. A. Zenios, Parallel Optimization: Theory, Algorithms, and Applications, Oxford Unversity Press, New York, 1997.

[16] Y. Censor, A. Segal, Iterative projection methods in biomedical inverse problems, in: Procceding of the interdisciplinary workshop on Mathematical Methods in Biomedical Imaging and Intensity-Modulated Radiation Therapy (IMRT), 2008, pp. 65–96.

[17] Y. Shen, M. Win, Fundamental limits of wideband localization–part i: A general framework, IEEE Trans. Inf. Theory 56 (10) (2010) 4956–4980.

[18] Z. Ma, W. Chen, K. Letaief, Z. Cao, A semi range-based iterative localization algorithm for cognitive radio networks, IEEE Trans. Veh. Technol. 59 (2) (2010) 704–717.

[19] S. M. Kay, Fundamentals of Statistical Signal Processing: Estimation theory, Englewood Cliffs, NJ: Prentice-Hall, 1993.

[20] S. S. Slijepcevic, S. Megerian, M. Potkonjak, Location errors in wireless embedded sensor networks: sources, models, and effects on applications, Sigmobile Mobile Computing and Communications Review 6 (2002) 67–78.

[21] M. R. Gholami, H. Wymeersch, E. G. Ström, M. Rydström, Robust distributed positioning algorithms for cooperative networks, in: SPAWC, 2011, pp. 156–160.

[22] H. Wymeersch, S. Maranó, W. M. Gifford, M. Z. Win, A machine learning approach to ranging error mitigation for UWB localization, IEEE Trans. Commun. 60 (6) (2012) 1719–1728.

[23] D. P. Palomar, Y. C. Eldar, Convex Optimization in Signal Processing and Communications, Cambridge University Press, 2010.

[24] A. Ben-Tal, A. Nemirovski, Lectures on modern convex optimization (2012).
URL `http://www2.isye.gatech.edu/~nemirovs/Lect_ModConvOpt.pdf`

[25] M. S. Bazaraa, H. D. Sherali, C. M. Shetty, Nonlinear Programming Theory and Algorithms, 3rd Edition, John Wiley & Sons, New Jersey, 2006.

[26] N. Patwari, J. Ash, S. Kyperountas, A. O. Hero, N. C. Correal, Locating the nodes: Cooperative localization in wireless sensor network, IEEE Signal Process. Mag. 22 (4) (2005) 54–69.

[27] P.-C. Chen, A non-line-of-sight error mitigation algorithm in location estimation, in: Proc. IEEE Wireless Communications and Networking Conference., 1999, pp. 316–320 vol.1.

[28] C. Liang, R. Piche, Mobile tracking and parameter learning in unknown non-line-of-sight conditions, in: Proc. 13th International Conference on Information Fusion, 2010.

[29] H. Chen, G. Wang, Z. Wang, H.-C. So, H. V. Poor, Non-line-of-sight node localization based on semi-definite programming in wireless sensor networks, IEEE Transactions on Wireless Communications 11 (2012) 108–116.

[30] R. M. Vaghefi, J. Schloemann, R. M. Buehrer, Nlos mitigation in TOA-based localization using semidefinite programming, in: 10th Workshop on Positioning Navigation and Communication (WPNC), 2013, pp. 1–6.

[31] V. Ekambaram, G. Fanti, K. Ramchandran, Semi-definite programming relaxation for non-line-of-sight localization, arXiv preprint arXiv:1210.5031.

[32] T. Jia, R. M. Buehrer, A set-theoretic approach to collaborative position location for wireless networks, IEEE Transactions on Mobile Computing 10 (9) (2011) 1264–1275.

[33] R. M. Vaghefi, R. M. Buehrer, Target tracking in nlos environments using semidefinite programming, in: IEEE Military Communications Conference, 2013, pp. 169–174.

[34] D. P. Bertsekas, Nonlinear Programming, 2nd Edition, Athena Scientific, 1999.

[35] S. Boyd, J. Dattorro, Alternating projections (2003).
URL `http://www.stanford.edu/class/ee392o/alt_proj.pdf`

[36] D. Blatt, A. O. Hero, APOCS: a rapidly convergent source localization algorithm for sensor networks, in: IEEE/SP Workshop on Statistical Signal Processing, 2005, pp. 1214–1219.

[37] M. R. Gholami, E. G. Ström, H. Wymeersch, S. Gezici, Upper bounds on position error of a single location estimate in wireless sensor networks, EURASIP Journal on Advances in Signal Processing 2014:4 (1).

[38] A. Nemirovski, C. Roos, T. Terlaky, On maximization of quadratic form over intersection of ellipsoids with common center, Mathematical Programming 86 (1999) 463–473.

[39] P. Tseng, Further results on approximating nonconvex quadratic optimization by semidefinite programming relaxation, SIAM J. Optim. 14 (1) (2003) 268–283.

[40] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, 2004.

[41] A. Beck, On the convexity of a class of quadratic mappings and its application to the problem of finding the smallest ball enclosing a given intersection of balls, Journal of Global Optimization 39 (2007) 113–126.

[42] E. W. Weisstein, Cuboid, A Wolfram Web Resource.
URL http://mathworld.wolfram.com/Cuboid.html

[43] A. Urruela, Signal processing techniques for wireless locationing, Ph.D. thesis, Technical University of Catalonia (2006).
URL http://spcom.upc.edu/documents/T_2006_urruela.pdf

[44] M. Grant, S. Boyd, CVX: Matlab software for disciplined convex programming, version 1.21 (Feb. 2011).
URL http://cvxr.com/cvx