(Article begins on next page)

ROBUST DISTRIBUTED POSITIONING ALGORITHMS FOR COOPERATIVE NETWORKS

Mohammad Reza Gholami, Henk Wymeersch, Erik G. Ström, and Mats Rydström

Chalmers University of Technology, Department of Signals and Systems, Gothenburg, Sweden

ABSTRACT

The problem of positioning targets based on distance estimates is studied for cooperative wireless sensor networks when there is limited *a priori* information about measurements noise. To solve this problem, two different methods of positioning are considered: statistical and geometrical. Based on a geometric interpretation, we show that the positioning problem can be rendered as finding the intersection of a number of convex sets. To find this intersection, we propose two different methods based on *projection onto convex sets* and *outer-approximation*. In the statistical approach, a partly novel *two-step* linear estimator is proposed which can be expressed in a closed-form solution. We also propose a new *constrained* non-linear least squares algorithm based on constraints derived in the outer-approximation approach. Simulation results show that the geometrical methods are more robust against non-line-of-sight measurements than the statistical approaches while in dense networks with line-of-sight measurements statistical approaches outperform geometrical methods.

Index Terms— Cooperative networks, positioning, statistical and geometrical estimators, robust estimation.

1. INTRODUCTION

Position information is one of the key requirement for a wireless sensor network (WSN). In conventional WSNs, positioning of target nodes is often carried out by the network itself using a number of reference nodes, also called anchor nodes, with known positions [1]. In the positioning literature, various methods of positioning have been proposed for different types of measurements. In general, positioning approaches can be divided into two different categories: statistical and geometrical [2, 3].

Recently, there has been a growing interest in *cooperative positioning* since it has excellent potential to localize targets in a scenario with few reference nodes [4]. Positioning of target nodes in cooperative networks is more complicated than in conventional networks, since the measurements among targets as well as measurements between targets and references need to be taken into account. Bayesian estimators can be employed to solve the cooperative positioning problem [4]. Since there are limitations in applying Bayesian estimators in practice, mainly due to complexity and robustness issues, there is a need to investigate more robust approaches with lower complexity.

In this paper, we employ a least squares approach and we formulate two versions of least squares, non-linear least squares (NLS) and a partly new linear least squares (LLS). In LLS, we obtain a two-step linear estimator which is algebraic and closed-form. Due to least squares' poor performance in non-line-of-sight (NLOS) condition, we consider a geometric technique to obtain a *robust* algorithm. We formulate the positioning problem as finding a point in the intersection of a number of convex sets, rendering the positioning problem to a feasibility problem. To find a feasible point, we use two geometric estimators; projection onto convex sets (POCS) and outer-approximation (OA). The proposed methods, which can be implemented in a distributed and cooperative manner, are robust in NLOS condition. We also derive a new distributed constrained NLS (CLNS) approach that shows great improvements in some situations. To evaluate and compare different methods, Monte Carlo simulations are performed for different scenarios. Simulation results show that the proposed algorithms are more robust than statistical methods for NLOS measurements.

The main contributions of this paper are four new cooperative positioning algorithms:

- a cooperative two-step LLS that is algebraic and closed-form in each step in section 3.1.2. For LOS measurements and dense networks, this estimator shows good performance;
- a cooperative POCS algorithm derived in section 3.2.1 that is robust against NLOS measurements;
- a cooperative geometric algorithm based on outer-approximation derived in section 3.2.2 that is a robust technique for positive measurement error and shows good performance for NLOS measurements;
- a constrained NLS derived in sections 3.1.1 and 3.2.2 that combines NLS with constraints derived from the results of the outer-approximation method. It shows good performance for both LOS and NLOS measurements.

The remainder of this paper is organized as follows. The signal model is briefly explained in section 2 and in section 3 different positioning algorithms are developed. Simulation results are discussed in section 4. Finally, section 5 concludes the paper.

2. SYSTEM MODEL

Let us consider a two-dimensional cooperative network with $N + M$ sensor nodes. Suppose M target nodes with unknown positions are randomly placed at $\mathbf{z}_i = [x_i\ y_i]^T \in \mathbb{R}^2$, $i = 1, ..., M$, and the remaining N reference nodes are located at known positions, $\mathbf{z}_j = [x_j\ y_j]^T \in \mathbb{R}^2$, $j = M + 1, ..., N + M$. Every target can connect to nearby reference nodes and some other targets. Let us define $\mathcal{A}_i = \{j |$ target j can communicate with reference node $i\}$ and $\mathcal{B}_i = \{j | j \neq i,$ target j can communicate with target $i\}$ as the sets of all reference and target nodes which are connected to target i. Suppose that sensor nodes are able to estimate distances to nodes with which they are connected, giving rise to the following observation:

$$\hat{d}_{i,j} = d(\mathbf{z}_i, \mathbf{z}_j) + \epsilon_{i,j},\ i = 1, \ldots, M,\ j \in \mathcal{A}_i \cup \mathcal{B}_i, \qquad (1)$$

This work was supported by the Swedish Research Council (contract no. 2007-6363)

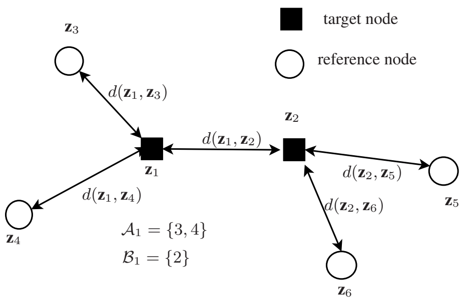

Fig. 1. A typical cooperative network

where $d(\mathbf{z}, \mathbf{z}_j) = \|\mathbf{z} - \mathbf{z}_j\|$ is the actual distance from sensor node j to the point $\mathbf{z}$ and $\epsilon_{i,j}$ is the measurement error. Throughout this paper we only assume that measurement errors are independent and identically distributed (i. i. d), and positive [5]. Fig. 1 shows a cooperative network consisting of two target nodes and four reference nodes.

3. POSITIONING ALGORITHMS

3.1. Statistical positioning algorithms

3.1.1. Non-linear least squares

In NLS, position estimates, based on the ranging measurement (1), can be obtained as the solution to the following non-linear non-convex minimization problem:

$$\hat{\mathbf{Z}} = \arg \min_{\substack{\mathbf{z}_i \in \mathbb{R}^2 \\ i=1,\ldots,M}} \sum_{i=1}^{M} \sum_{j \in \mathcal{A}_i \cup \mathcal{B}_i} \left(\hat{d}_{i,j} - d(\mathbf{z}_i, \mathbf{z}_j) \right)^2 , \tag{2}$$

where $\hat{\mathbf{Z}} = [\hat{\mathbf{z}}_1, \ldots, \hat{\mathbf{z}}_M]$. There is, in general, no analytical solution to (2) and we resort to numerical methods. For the rest of the paper let us consider

$$\tilde{\mathbf{z}}_j = \begin{cases} \hat{\mathbf{z}}_j, & j = 1, \ldots, M \\ \mathbf{z}_j, & j = M+1, \ldots, M+N \end{cases} . \tag{3}$$

A distributed solution of NLS can be obtained for cooperative networks as follows:

- $\mathcal{C}_i = \emptyset$, $i = 1, \ldots, M$
- For $k = 0$ until convergence or a predefined number K
- For $i = 1, \ldots, M$

$$\hat{\mathbf{z}}_i = \arg \min_{\mathbf{z}_i \in \mathbb{R}^2} \sum_{j \in \mathcal{A}_i \cup \mathcal{C}_i} \left(\hat{d}_{i,j} - d(\mathbf{z}_i, \tilde{\mathbf{z}}_j) \right)^2 ,$$
$$\text{if } i \notin \mathcal{C}_m \text{ then } \mathcal{C}_m = \mathcal{C}_m \cup \{i \cap \mathcal{B}_m\},\ m = 1, \ldots, M. \tag{4}$$

It is clear that the performance of the algorithm strongly depends on initial position estimates.

3.1.2. Linear Estimator

In this section we obtain a two-step linear estimator. In the first step a coarse estimate is obtained that is then refined in a second step. Let us define $\mathcal{E}_i \subseteq \mathcal{B}_i$ as the set of all target nodes which have already been localized connected to the target i. For the first step of updating for target i, squaring both sides of (1), after dropping small terms, yields

$$\tilde{d}_{i,j} = \hat{d}_{i,j}^2 - \|\tilde{\mathbf{z}}_j\|^2 \approx [-2\tilde{\mathbf{z}}_j^T\ 1]\boldsymbol{\psi}_i + 2d(\mathbf{z}_i, \tilde{\mathbf{z}}_j)\epsilon_{i,j},$$
$$j \in \mathcal{A}_i \cup \mathcal{E}_i, \tag{5}$$

where $\boldsymbol{\psi}_i = \left[\mathbf{z}_i^T\ \|\mathbf{z}_i\|^2\right]^T$. Now a set of linear equations can be written as

$$\mathbf{d}_i = \mathbf{A}_i \boldsymbol{\psi}_i + \boldsymbol{\nu}_i , \tag{6}$$

where $\mathbf{d}_i = \left[\tilde{d}_{i,j_1} \tilde{d}_{i,j_2} \ldots \tilde{d}_{i,j_L}\right]^T$, $\mathbf{A}_i = \left[\mathbf{a}_{j_1}^T \ldots \mathbf{a}_{j_L}^T\right]^T$, $\boldsymbol{\nu}_i = 2\left[d(\mathbf{z}_i, \tilde{\mathbf{z}}_{j_1})\epsilon_{i,j_1} \ldots d(\mathbf{z}_i, \tilde{\mathbf{z}}_{j_L})\epsilon_{i,j_L}\right]^T$, $\{j_1, \ldots, j_L\} = \mathcal{A}_i \cup \mathcal{E}_i$, $L = |\mathcal{A}_i \cup \mathcal{E}_i|$ is the cardinality of set $\mathcal{A}_i \cup \mathcal{E}_i$, and $\mathbf{a}_{j_p} = [-2\tilde{\mathbf{z}}_{j_p}^T\ 1]$.

Suppose that the matrix $\mathbf{A}_i$ is full rank, so the unknown parameter $\boldsymbol{\psi}_i$ can be estimated by [6]

$$\hat{\boldsymbol{\psi}}_i = (\mathbf{A}_i^T \mathbf{C}_{\boldsymbol{\nu}_i}^{-1} \mathbf{A}_i)^{-1} \mathbf{A}_i^T \mathbf{C}_{\boldsymbol{\nu}_i}^{-1} \mathbf{d}_i , \tag{7}$$

where matrix $\mathbf{C}_{\boldsymbol{\nu}_i}$, for i. i. d measurement noise, is computed as $\mathbf{C}_{\boldsymbol{\nu}_i} = 4\text{diag}(d^2(\mathbf{z}_i, \tilde{\mathbf{z}}_{j_1}), \ldots, d^2(\mathbf{z}_i, \tilde{\mathbf{z}}_{j_L}))$. The covariance matrix of $\hat{\boldsymbol{\psi}}_i$ is $\text{cov}(\hat{\boldsymbol{\psi}}_i) = (\mathbf{A}_i^T \mathbf{C}_{\boldsymbol{\nu}_i}^{-1} \mathbf{A}_i)^{-1}$ [6].

To compute the weighting matrix $\mathbf{C}_{\boldsymbol{\nu}_i}$, since in practice the real distances are not available, we instead use the measured distances in (1).

Now we update the set $\mathcal{E}_m$ and the ith target's position

$$\mathcal{E}_m = \mathcal{E}_m \cup \{i \cap \mathcal{B}_m\} ,\ m = 1, \ldots, M,\ \hat{\mathbf{z}}_i = [\hat{\boldsymbol{\psi}}_i]_{2\times1}, \tag{8}$$

where in general, $\mathbf{A}_{n\times m}$ denotes the upper left $n \times m$ part of matrix $\mathbf{A}$. The first step of updating for target i, which gives a coarse estimate, is just run once. In the next steps, the previous estimate is refined using a new estimator. Let us apply a first order Taylor-series expansion around $\hat{\mathbf{z}}_i$ for the measurement in (1) to get

$$\hat{d}_{i,j} = d(\hat{\mathbf{z}}_i, \tilde{\mathbf{z}}_j) + \frac{\hat{\mathbf{z}}_i - \tilde{\mathbf{z}}_j}{d(\hat{\mathbf{z}}_i, \tilde{\mathbf{z}}_j)}(\mathbf{z}_i - \hat{\mathbf{z}}_i) + \epsilon_{i,j} ,\ j \in \mathcal{A}_i \cup \mathcal{E}_i,$$

The new linear set of measurements can be written in matrix form as

$$\tilde{\mathbf{d}}_i = \mathbf{G}_i \triangle \mathbf{z}_i + \boldsymbol{\epsilon}_i, \tag{9}$$

where $\triangle \mathbf{z}_i = \mathbf{z}_i - \hat{\mathbf{z}}_i$ and vectors $\tilde{\mathbf{d}}_i$, $\boldsymbol{\zeta}_i$, and matrix $\mathbf{G}_i$ are obtained as follows

$$\tilde{\mathbf{d}}_i = \left[\hat{d}_{i,j_1} - d(\hat{\mathbf{z}}_i\ \tilde{\mathbf{z}}_{j_1})\ \ldots\ \hat{d}_{i,j_P} - d(\hat{\mathbf{z}}_i, \tilde{\mathbf{z}}_{j_P})\right]^T,$$

$$\mathbf{G}_i = \begin{bmatrix} \frac{(\hat{\mathbf{z}}_i - \tilde{\mathbf{z}}_{j_1})^T}{d(\hat{\mathbf{z}}_i, \tilde{\mathbf{z}}_{j_1})} \\ \vdots \\ \frac{(\hat{\mathbf{z}}_i - \tilde{\mathbf{z}}_{j_P})^T}{d(\hat{\mathbf{z}}_i, \tilde{\mathbf{z}}_{j_P})} \end{bmatrix},\ \boldsymbol{\epsilon}_i = [\epsilon_{i,j_1}\ \ldots\ \epsilon_{i,j_P}]^T ,$$

where $\{j_1, \ldots, j_P\} = \mathcal{A}_i \cup \mathcal{E}_i$ and $P = |\mathcal{A}_i \cup \mathcal{E}_i|$.

To solve the linear equation in (9), we add a constraint on the unknown parameter $\triangle \mathbf{z}_i$ to be small (if possible). Therefore we can consider the following optimization problem

$$\underset{\triangle \mathbf{z}_i}{\text{minimize}}\ \|\tilde{\mathbf{d}}_i - \mathbf{G}_i \triangle \mathbf{z}_i\|^2 + \gamma\ \|\triangle \mathbf{z}_i\|^2, \tag{10}$$

where the regularized parameter $\gamma > 0$ determines the tradeoff between $\|\tilde{\mathbf{d}}_i - \mathbf{G}_i \triangle \mathbf{z}_i\|^2$ and $\|\triangle \mathbf{z}_i\|^2$. The solution to (10) can be obtained as [7]

$$\hat{\triangle}\mathbf{z}_i = \left(\mathbf{G}_i^T\mathbf{G}_i + \gamma\mathbf{I}_2\right)^{-1}\mathbf{G}_i^T\tilde{\mathbf{d}}_i, \quad (11)$$

where $\mathbf{I}_n$ denotes an $n \times n$ identity matrix. The updated estimate is $\hat{\hat{\mathbf{z}}}_i = \hat{\mathbf{z}}_i + \hat{\triangle}\mathbf{z}_i$.

3.2. Geometrical positioning algorithms

The estimators introduced in previous sections fail to work in NLOS, or when reference node density is low. In this section, we propose robust techniques based on a geometric interpretation of the positioning problem that avoid both drawbacks.

For target i, from (2) it is clear that the minimum of each term is obtained when $d(\mathbf{z}_i, \mathbf{z}_j) = \hat{d}_{i,j}$. Now, suppose that we define the disc $\mathcal{D}(\mathbf{z}_j, \hat{d}_{i,j})$ centered at $\mathbf{z}_j$ as

$$\mathcal{D}(\mathbf{z}_j, \hat{d}_{i,j}) = \left\{\mathbf{z} \in \mathbb{R}^2 : \|\mathbf{z} - \mathbf{z}_j\| \le \hat{d}_{i,j}\right\}, \ j \in \mathcal{A}_i \cup \mathcal{B}_i, \quad (12)$$

it then is reasonable to define an estimate of $\mathbf{z}_i$ as a point in the intersection, i.e., $\mathcal{D}_i$, of the discs $\mathcal{D}(\mathbf{z}_j, \hat{d}_{i,j})$

$$\hat{\mathbf{z}}_i \in \mathcal{D}_i = \bigcap_{j \in \mathcal{A}_i \cup \mathcal{B}_i} \mathcal{D}(\mathbf{z}_j, \hat{d}_{i,j}) \ . \quad (13)$$

Geometric positioning algorithms solve the following feasibility problem:

$$\text{find } \mathbf{Z} = [\mathbf{z}_1 \ \ldots \ \mathbf{z}_M] \text{ such that } \mathbf{z}_i \in \mathcal{D}_i, \ i = 1, \ldots, M.$$

In the sequel we study this class of estimators.

3.2.1. *Projection onto convex sets (POCS)*

For non-cooperative networks, POCS is a robust positioning method that can be implemented in a distributed manner [8]. To our knowledge, POCS has not previously applied for the cooperative positioning. To apply POCS, we must unambiguously define all the discs, $\mathcal{D}(\mathbf{z}_j, \hat{d}_{i,j})$, for every target i. From (12), it is clear that some discs, i.e., discs centered around a reference node, can be defined without any ambiguity, i.e., $\mathcal{D}(\mathbf{z}_j, \hat{d}_{i,j}), \ j \in \mathcal{A}_i$. Other discs derived from measurements between targets, i.e., discs centered around an unknown target, have an unknown center, i.e., $\mathcal{D}(\mathbf{z}_j, \hat{d}_{i,j}), \ j \in \mathcal{B}_i$. Let us consider Fig. 2 where for target node one, we want to involve the measurement between target two and target one. Since there is no prior knowledge about the position of target two, the disc centered around target two cannot be involved in the localization process for target one. Suppose, based on applying POCS to the discs defined by reference nodes $\mathbf{z}_5$ and $\mathbf{z}_6$, we obtain an initial estimate $\hat{\mathbf{z}}_2$ for target two. Now, based on distance estimate $\hat{d}_{1,2}$, we can define a new disc centered around $\hat{\mathbf{z}}_2$. This new disc can be combined with the two other discs defined by reference nodes $\mathbf{z}_3$ and $\mathbf{z}_4$. Fig. 2 shows the process for localizing target one. For target two, the same procedure is followed.Now we can implement cooperative POCS (Coop-POCS) as follows:

- $\mathcal{C}_i = \emptyset$, $i = 1, \ldots, M$
- For $k = 0$ until convergence or a predefined number K

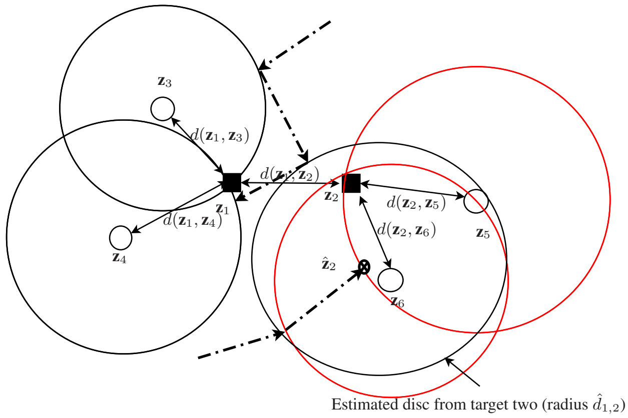

Fig. 2. Applying POCS for target one considering initial estimate of target two, $\hat{\mathbf{z}}_2$.

- For $i = 1, \ldots, M$, find $\hat{\mathbf{z}}_i$ with POCS such that

$$\hat{\mathbf{z}}_i \in \mathcal{D}_i = \bigcap_{j \in \mathcal{A}_i \cup \mathcal{C}_i} \mathcal{D}(\tilde{\mathbf{z}}_j, \hat{d}_{i,j}),$$

if $i \notin \mathcal{C}_m$ then $\mathcal{C}_m = \mathcal{C}_m \cup \{i \cap \mathcal{B}_m\}, m = 1 \ldots M$.

For inconsistent problem for the target i, i.e., empty intersection, POCS minimizes the sum of squared distances to the sets $\mathcal{D}(\tilde{\mathbf{z}}_j, \hat{d}_{i,j}), \ j \in \mathcal{A}_i \cup \mathcal{C}_i$ [8].

3.2.2. *Outer-approximation (OA)*

As we saw before, the position of an unknown target can be found in the intersection of a number of discs. The intersection in general may have any convex shape. Based on the assumption in section 2 of positive measurement noise, the target i definitely lies inside of the intersection $\mathcal{D}_i$ in (13).

In contrast to POCS, which tries to find a point in the intersection as an estimate, OA determines an outer-approximation of the intersection and then one point inside of it is taken as an estimate. The main problem is how to accurately approximate the intersection. There is some work in the positioning literature to approximate the intersection by convex regions such as polytopes or ellipsoids [9]. To apply OA for cooperative networks, we consider a disc approximation of the intersection since it can be easily obtained and exchanged between targets.

Using simple geometry, we are able to find all intersection points between different discs. Let $\mathbf{z}_k^I$, $k = 1, \ldots, L$ be the set of intersection points. Among all intersection points, some of them are redundant and will be discarded. Therefore the common points that belong to the intersection are selected as $\mathcal{S}_{\text{int}} = \{\mathbf{z}_k^I | \mathbf{z}_k^I \in \mathcal{D}_i\}$. The problem therefore renders to finding a disc which contains $\mathcal{S}_{\text{int}}$ and cover the intersection, which is a well-known convex optimization problem [7].

Now we implement cooperative OA (Coop-OA) as:

- $\mathcal{C}_i = \emptyset$, $i = 1, \ldots, M$
- For $k = 0$ until convergence or a predefined number K

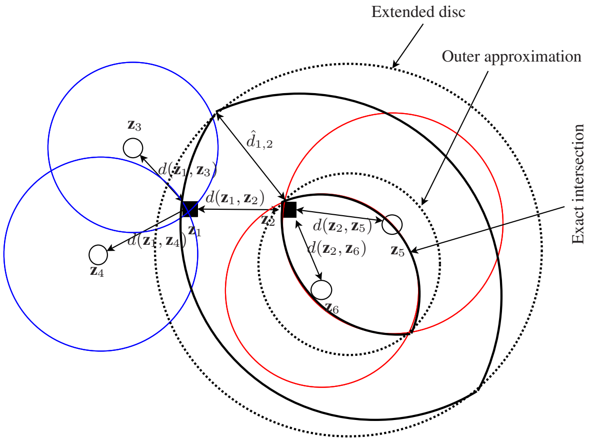

Fig. 3. Extending the convex region involving target two to help target one to find smaller intersection.

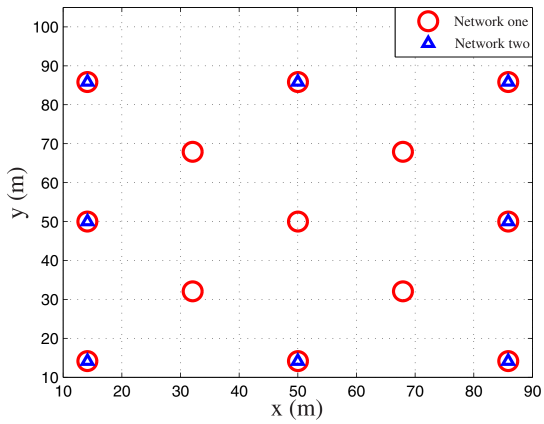

Fig. 4. Simulation environment for network one and two.

- For $i = 1, \ldots, M$

$$\mathcal{D}(\hat{\mathbf{z}}_i, \hat{R}_i) = \text{OA}\Big\{ \bigcap_{j \in \mathcal{A}_i \cup \mathcal{C}_i} \mathcal{D}(\tilde{\mathbf{z}}_j, \tilde{R}_j) \Big\},$$

$$(\tilde{\mathbf{z}}_j, \tilde{R}_j) = \begin{cases} (\mathbf{z}_j, \hat{d}_{i,j}), & j \in \mathcal{A}_i \\ (\hat{\mathbf{z}}_j, \hat{d}_{i,j} + \hat{R}_j) & j \in \mathcal{C}_i \end{cases},$$

if $i \notin \mathcal{C}_m$ then $\mathcal{C}_m = \mathcal{C}_m \cup \{i \cap \mathcal{B}_m\}$, $m = 1, \ldots, M$.

To see how this method works, consider Fig. 3 where target two helps target one to improve its positioning. Target two can be found in the intersection derived from two discs centered around $\mathbf{z}_5$ and $\mathbf{z}_6$ in non-cooperative mode (semi oval shape). Suppose that we approximate it by a disc (small dashed circle). To help target one to approximate its intersection in cooperative mode, this region should be involved in finding the intersection for target one. We can easily extend every point of this disc by $\hat{d}_{1,2}$ to come up with a bigger disc (big dashed circle). It is seen that the approximated intersection for target one is smaller than that for the non-cooperative case. Note if we use extended the exact intersection, we end up with a better approximation for the intersection of target one. We now consider the intersection obtained in Coop-OA as a constraint for NLS methods (CNLS) to improve the performance of the algorithm mentioned in (4). Suppose that for target i we obtain a final disc as $\mathcal{D}_i(\hat{\mathbf{z}}_i, \hat{R}_i)$, where $\hat{\mathbf{z}}_i$ and $\hat{R}_i$ are the center and the radius of the disc respectively. It is clear that we can define $\|\mathbf{z}_i - \hat{\mathbf{z}}_i\| \leq \hat{R}_i$ as a constraint for the ith target in the optimization problem (4). This problem can be solved iteratively similar to Section 3.1.1.

4. SIMULATION

In this section, computer simulations are performed to evaluate the performance of the different algorithms. To compare different methods, we consider the cumulative distribution function (CDF) of the positioning error. The simulation area, a 100 m $\times$ 100 m square area, consists of a number of reference nodes placed at fixed positions and also 100 targets randomly distributed over the area. We assume a pair of nodes can connect and estimate the distance between each other if that distance is less than 20 m. The measurement noise is assumed to have an exponential distribution with mean and standard deviation equal to 50 cm and 50 cm respectively. We consider two different networks based on the number of reference nodes shown in Fig. 4. To implement the CNLS estimator, we use the MATLAB routine `fmincon` [10] initialized and constrained with Coop-OA described in Section 3.2. Without any attempt to optimize the regularization parameter, we simply set $\gamma = 0.5$. To make a fair comparison between different methods, we consider two scenarios for these network deployments, LOS and NLOS situations. In cooperative mode, the estimated parameters, points or discs, are exchanged among different targets 10 times. All CDFs were obtained for 30 networks. Note that for the non-cooperative case we used POCS to find an estimate.

The CDFs of errors for the different methods in a LOS scenario are shown in Fig. 5 and Fig. 6. There are some interesting results to be pointed out. As can be seen, CNLS outperforms the other methods in most of the cases. The linear estimator for network one works well for most of the targets since there are enough known reference nodes around them. In network two, this method fails since there are less than three known reference nodes for a target. It is seen that the geometrical approach can improve the performance of cooperative positioning compared to the non-cooperative case. The geometric methods also works well when the number of anchors is decreased. It is also seen that Coop-POCS shows good performance for small errors compared to the Coop-OA especially when the number of reference nodes are decreased.

To see the performance of different methods in an NLOS situation, we run another simulation for network one and network two where 20% of measurements suffer from a NLOS situation. To simulate a NLOS measurement, we add a uniform random number between 0 to 20 m to the LOS measurement. Fig. 7 and Fig. 8 show the error CDFs of the different methods. As can be seen in comparison to the LOS case (Fig. 5 and Fig. 6) the geometric approaches are more robust compared to the statistical approaches. Fig. 7 also shows that the linear approach performs poorly and it can not locate targets well even there are enough reference nodes in network one. Since the outer-approximation method, as well as POCS, is highly robust and can easily handle NLOS measurements, CNLS still performs well.

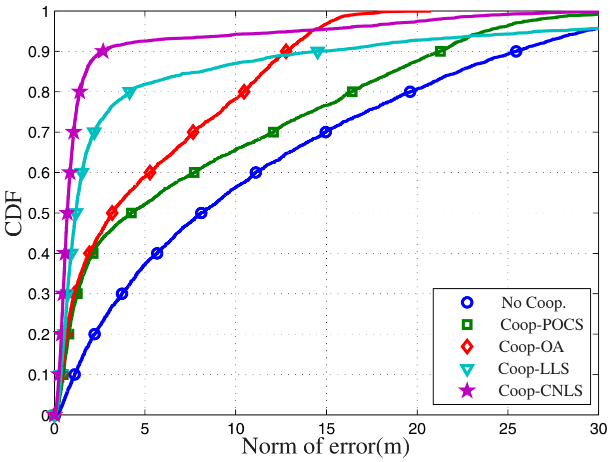

Fig. 5. CDFs of different algorithms for network one, LOS situation.

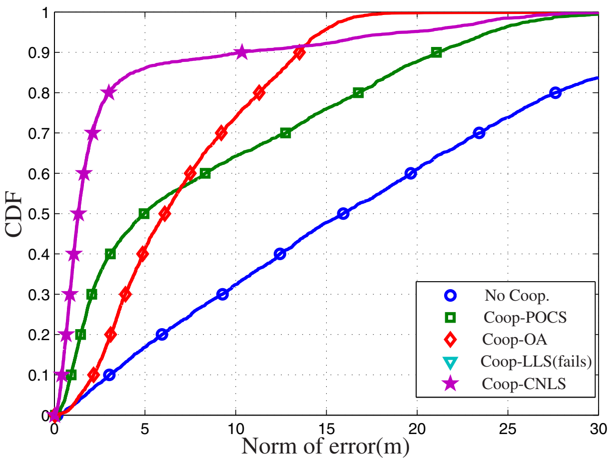

Fig. 6. CDFs of different algorithms for network two, LOS situation.

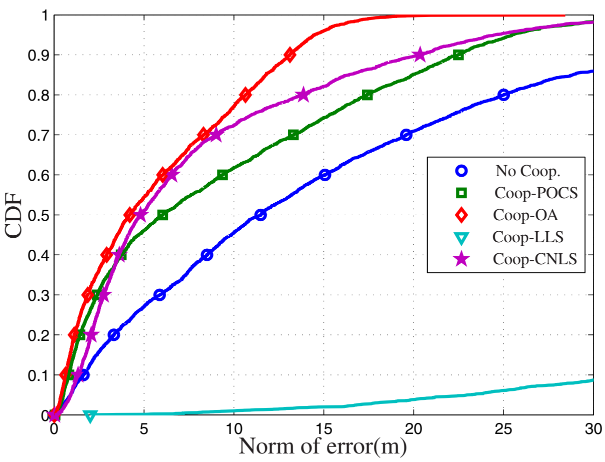

Fig. 7. CDFs of different algorithms for network one, NLOS situation

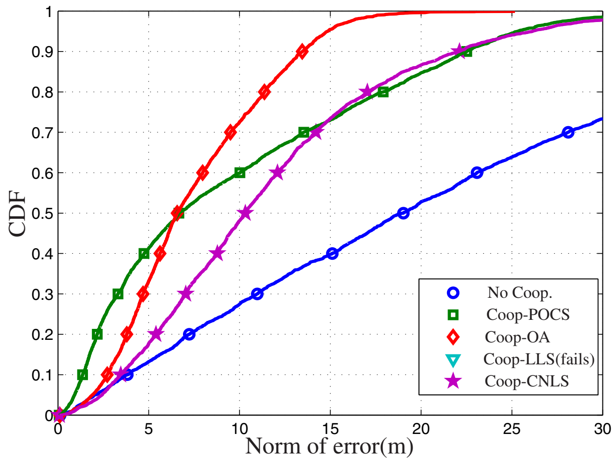

Fig. 8. CDF of different algorithms for network two, NLOS situation

5. CONCLUSION

In this paper, we have studied the problem of positioning based on distance measurements in a cooperative network using two classes of estimators; statistical and geometrical. The distributed non-linear least squares (NLS) and partly new distributed linear least squares were obtained in the statistical category. To have more robust algorithms based on a geometric interpretation, we have formulated the problem of positioning as finding the intersection of a number of convex sets and in the sequel, we have employed two different methods based on projection onto convex sets and outer-approximation (OA). We also proposed a version of NLS based on constraints derived using the OA method which can be considered as a hybrid estimator. Simulation results show that the considered geometric techniques are more robust to non-line of sight situations and low density of anchor nodes than the considered statistical methods.

6. REFERENCES

[1] G. Mao and B. Fidan, *Localization algorithms and strategies for wireless sensor networks*. Information Science reference, Hershey. New York, 2009.

[2] M. R. Gholami, "Positioning algorithms for wireless sensor networks," Licentiate thesis, Chalmers University of Technology, Mar. 2011.

[3] M. R. Gholami, E. G. Ström, and M. Rydström, "Indoor sensor node positioning using UWB range measurments," in *Proc. 17th European Signal Processing Conference(Eusipco)*, 2009, pp. 1943–1947.

[4] H. Wymeersch, J. Lien, and M. Z. Win, "Cooperative localization in wireless networks," *Proc. IEEE*, vol. 97, no. 2, pp. 427–450, Feb. 2009.

[5] S. Marano, W. M. Gifford, H. Wymeersch, and M. Z. Win, "Non-parametric obstruction detection for UWB localization," in *Proc. IEEE Global Communication Conference*, Dec. 2009, pp. 1–6.

[6] S. M. Kay, *Fundamentals of statistical signal processing: estimation theory*. Upper Saddle River, NJ: Prentice-Hall, 1993.

[7] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[8] D. Blatt and A. O. Hero, "Energy-based sensor network source localization via projection onto convex sets," *IEEE Trans. Signal Process.*, vol. 54, no. 9, pp. 3614–3619, Sep. 2006.

[9] L. Doherty, K. S. J. Pister, and L. E. Ghaoui, "Convex position estimation in wireless sensor networks," in *INFOCOM 2001*, vol. 3, 2001, pp. 1655–1663.

[10] The Mathworks Inc., 2010, http://www.mathworks.com.