

Article

Efficient Distributed Method for NLOS Cooperative Localization in WSNs

Shiwa Chen ^{1,*}, Jianyun Zhang ¹, Yunxiang Mao ¹, Chengcheng Xu ¹ and Yu Gu ²

¹ College of Electronic Engineering, National University of Defense Technology, Hefei 230037, China; zjy921@sina.com (J.Z.); myxeei316@sina.com (Y.M.); xuchengcheng17@nudt.edu.cn (C.X.)

² State Key Laboratory of Pulsed Power Laser Technology, National University of Defense Technology, Hefei 230037, China; terryguyu@163.com

* Correspondence: chenshiwa17@nudt.edu.cn

Received: 24 January 2019; Accepted: 25 February 2019; Published: 7 March 2019



Abstract: The accuracy of cooperative localization can be severely degraded in non-line-of-sight (NLOS) environments. Although most existing approaches modify models to alleviate NLOS impact, computational speed does not satisfy practical applications. In this paper, we propose a distributed cooperative localization method for wireless sensor networks (WSNs) in NLOS environments. The convex model in the proposed method is based on projection relaxation. This model was designed for situations where prior information on NLOS connections is unavailable. We developed an efficient decomposed formulation for the convex counterpart, and designed a parallel distributed algorithm based on the alternating direction method of multipliers (ADMM), which significantly improves computational speed. To accelerate the convergence rate of local updates, we approached the subproblems via the proximal algorithm and analyzed its computational complexity. Numerical simulation results demonstrate that our approach is superior in processing speed and accuracy to other methods in NLOS scenarios.

Keywords: wireless sensor networks (WSN); cooperative localization; non-line-of-sight (NLOS); alternating direction method of multipliers (ADMM); convex relaxation

1. Introduction

Wireless-sensor-network (WSN) technology has rapidly developed because of its convenience and prospects. It can significantly improve living quality in many important fields, such as environment monitoring [1] and surveillance [2], vehicle tracking [3,4], exploration [5,6], and other sensing tasks [7]. According to Reference [1], WSNs extend the human ability “to monitor and control physical world”. This means that WSNs provide a more intelligent way to link the physical world with humans. It is worth noting that the positions of sensor nodes are key information for WSNs to fulfil various tasks. Therefore, as a preliminary task, cooperative localization has aroused increasing interest, especially in indoor scenarios where satellite communications cannot be employed. In general, cooperative localization has two main categories: rangefree methods and range-based methods [8]. Rangefree methods are easier to implement, but their accuracy is lower than range-based methods [9]. The most common range-measurement techniques of range-based localization are based on received signal-strength indicator (RSSI) [10] and time of flight (TOF) [11]. The RSSI technique has a lower cost than the TOF technique, but the latter has higher accuracy. In this paper, we mainly study the localization problem based on range-based technology. Some related work in this field is listed as follows.

1.1. Related Work

The maximum-likelihood (ML) problem of sensor cooperative localization is a complicated nonconvex problem of high dimensionality. It is quite difficult to obtain an optimal solution. Most existing approaches try to find a feasible solution by applying relaxation methods to the original nonconvex problem. Among them, the most typical ones are listed. Semidefinite programming (SDP) relaxation for cooperative localization is proposed in References [12–14]. In Reference [13], the authors formulated the cooperative localization problem via graph realization theory, and derived the upper and lower bounds of the SDP objective function. In Reference [14], the authors developed three relaxations: node-based SDP (NSDP), edge-based SDP (ESDP), and sub-SDP (SSDP). NSDP and ESDP are weaker than the original SDP relaxation of Reference [12], but they both remain efficient and accurate. SSDP paves a faster way to efficiently solve a general SDP problem without sacrificing solution quality. In Reference [15], the authors proposed a novel model by combining angle information with range measurements, and transformed the model into an SDP problem. This method has excellent performance, but it also has considerably high computational complexity. The work in References [16,17] proposes second-order cone programming (SOCP) by relaxing distance constraints. The work in Reference [16] further applies SOCP to alleviate the computational burden of the standard SDP problem at the price of performance degradation. The authors in Reference [17] designed a method suited for both Gaussian and Laplacian noise environments. In addition to the aforementioned convex relaxations, some other methods also contribute to improving localization performance. In Reference [18], the authors formulated the problem as a regression problem over adaptive bases. They utilized the eigenvector of a distance affinity matrix as the initial point and implemented iterations by conjugate gradient descent. In Reference [19], the authors derived a majorization–minimization (MM) algorithm with quadratic objective function. However, all the methods or algorithms above are implemented in a centralized framework.

The centralized framework is a classical paradigm that transmits data to the central or fusion node to fulfil entire network tasks. A centralized framework is easy to implement, but the computational burden becomes extremely heavy for large-scale sensor networks. Centralized algorithms are prone to data-traffic bottlenecks among sensors around the central node [20]. As the number of sensor nodes grows, the problem size and the computational complexity in the centralized framework dramatically increase. Scale limits and time delay brought by the centralized paradigm may affect the engineering implementation of WSNs. Due to the advent of large-scale networks, it is urgent to find an effective framework to satisfy the requirements on both processing speed and localization accuracy. Therefore, the distributed paradigm has become a new tendency in this field.

Compared with the centralized paradigm, the distributed paradigm is an algorithm with all nodes performing the same type of computation. A number of distributed approaches have been proposed over the years, such as the distributed gradient descent method [21], multidimensional scaling method [22], and the sequential greedy optimization (SGO) algorithm [23]. In Reference [24], the authors developed a parallel distributed algorithm based on SOCP relaxation, but the convergence property of the algorithm was not established. The work in Reference [25] proposes a sequential method: the estimated neighbors are regarded as new anchors, which are then used to estimate other sensors. This process only stops when the whole network is covered. In Reference [23], the authors applied the SGO algorithm to ESDP and SOCP relaxation formulations. The work in Reference [26] developed an ESDP relaxation method and designed a distributed algorithm relying on the alternating direction method of multipliers (ADMM), of which the convergence property was guaranteed and proven in Reference [27]. Then, in Reference [28], the authors proposed a distributed method based on ADMM in the presence of harsh nonconvexities. In order to guarantee that the solution converges to the optimal point, the authors established the mathematic model to choose penalty parameters. The work in Reference [20] proposed a tighter convex problem via projection relaxation and put this problem into the distributed Nesterov framework. A hybrid solution was presented in Reference [29] by combining the convex relaxations of Reference [20] and the distributed method of Reference [28].

These distributed approaches are effective and bring a controlled computational burden. Convergence performance is generally guaranteed.

However, the aforementioned distributed algorithms work only in line-of-sight (LOS) scenarios. In some practical situations, such as in forests, cities, and indoor places, most connections between sensors are non-line-of-sight (NLOS) because of the obstacles in the direct paths of signal propagation. This phenomenon can severely degrade localization accuracy if the NLOS impact is not taken into consideration. In general, two approaches are applied to alleviate NLOS propagation in localization. The first approach distinguishes LOS and NLOS connections via prior information [30–32]. This approach needs other measurement techniques to provide NLOS information, such as Direction of Arrival (DOA). Hence, its estimation model is less suitable for cooperative localization methods based on range-based techniques. The second approach modifies the original model by weighting and adding constraints of range-measurement errors [33,34]. This kind of approach can be applied to more general scenarios. Some representative work is listed as follows. The work in Reference [35] is “the first one to address NLOS node localization in WSN”. According to three different scenarios, the authors provided three modified models of the ML estimator by adding the upper and lower bounds of range measurements. These models were solved in standard SDP problems with heuristic estimation. The work in Reference [34] proposed an ESDP method and added constraints to improve robustness. In Reference [36], the authors introduced NLOS bias parameters to the estimation model for the source-node locations and turned the model into a SDP problem. In Reference [37], the authors proposed a three-block ADMM algorithm based on the model in Reference [36]. However, this method does not decrease the size of the original problem, and the authors also mention that “it is hard to distribute the calculation” because it involves matrix projection. The work in Reference [38] proposed a distributed algorithm based on the Huber estimation of Reference [39], but it did not provide theoretical proof for the convergence property.

1.2. Contributions

Up to now, most NLOS mitigation techniques applied to WSN cooperative localization still utilize centralized optimum algorithms. In this paper, we propose a parallel distributed algorithm based on a tight relaxation technique to both decrease computational complexity and improve accuracy in NLOS environments.

First, we propose a modified convex model based on projection relaxation, which relaxes the original nonconvex problem into its convex envelope. This model can be applied to tough situations where NLOS connections are unidentifiable in all range measurements. According to the bounds of range measurements, we formulated the problem in the form of projection distances. Then, we relaxed this formulation into the projection on convex sets by using Cauchy–Schwartz inequality. Compared with SDP, this approach improves the decomposition property of the convex model and the accuracy of the estimation results.

Second, we developed a parallel distributed algorithm to solve the convex model. We designed a consensus form to decompose the large-scale problem into numerous local subproblems and provide the relevant theoretical proof. The proposed consensus form is more suitable for an ADMM framework. It enables each node to solve each subproblem in parallel. We derived the concrete procedure of how to handle the problem in a parallel way. The distributed algorithm had much lower computational complexity than that in existing papers about NLOS cooperative localization.

Third, we further improved the convergence rate of local updates. The local subproblems are convex and nonquadratic differentiable, which makes them less appropriate for the Newton method and interior-point algorithm. Hence, with the guarantee of Lipschitz continuity, we propose an iterative algorithm to solve the untypical convex subproblems based on the proximal method.

The paper is organized as follows. Section 2 formulates the cooperative localization problem and the corresponding convex relaxation. Section 3 presents the consensus form and solves it

in a distributed way. Section 4 derives the iterative method for local subproblems. In Section 5, the simulation results are reported. Section 6 concludes the paper.

2. Problem Formulation

2.1. Mathematic Model

The mathematic model of range-based cooperative localization is described as follows. As is shown in Figure 1, consider a sensor network consisting of N source sensors, of which locations $x_i \in \mathbb{R}^d$, $i = 1, 2, \dots, N$ are unknown, and M anchor sensors of which the location $x_k \in \mathbb{R}^d$, $k=N+1, \dots, N+M$ is known. d is the coordinate dimension. Source nodes are collected in the set $\mathcal{S} = \{1, 2, \dots, N\}$, and anchors are collected in the $\mathcal{A} = \{N+1, N+2, \dots, N+M\}$ set. We denote the Euclidean distance between node i and j as $d_{i,j}$, and the noisy range measurement as $r_{i,j}$. In general, not every pair of nodes can communicate because the communication distance has an upper limit. We denoted this distance limit as r_μ . The neighbor of node i is denoted as $j \in \mathcal{N}_i$ if $r_{i,j}$ is available, i.e., $\mathcal{N}_i = \{j | r_{i,j} \leq r_\mu, \forall j \in \mathcal{S}\}$, $\forall i \in \mathcal{S} \cup \mathcal{A}$. We denoted the pairwise of source sensors as $(i, j) \in \mathcal{Z}_S$, and the pairwise between a source sensor and an anchor as $(i, k) \in \mathcal{Z}_A$. The distance between two nodes is defined as

$$d_{i,j} = \|x_i - x_j\| \quad (1)$$

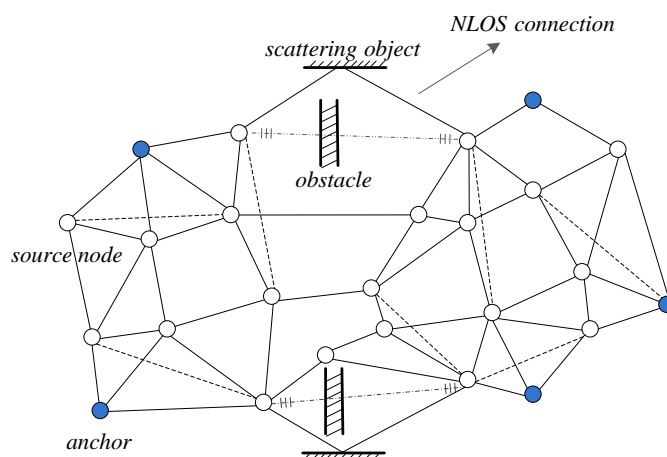


Figure 1. Wireless sensor network (WSN) figuration. Blue circles denote anchors; white circles denote source nodes. The physical model of non-line-of-sight (NLOS) connections is presented by *obstacle* and *scattering object*.

Considering the impact of NLOS propagation on distance estimation, we divided the range measurements into two sets. We used \mathcal{Z}_{LOS} (and, respectively, $\mathcal{Z}_{\text{NLOS}}$) to denote the set of pairwise nodes in which connections between nodes are LOS (and, respectively, NLOS). Hence, the range measurements are defined as:

$$\begin{cases} r_{i,j} = d_{i,j} + n_{i,j}, (i, j) \in \mathcal{Z}_{\text{LOS}} \\ r_{i,j} = d_{i,j} + \varepsilon_{i,j} + n_{i,j}, (i, j) \in \mathcal{Z}_{\text{NLOS}} \end{cases} \quad (2)$$

where $n_{i,j} \sim N(0, \sigma_{i,j}^2)$ is the measurement noise following a zero-mean Gaussian distribution with variance $\sigma_{i,j}^2$, and $\varepsilon_{i,j}$ is the error of the NLOS measurement that is exponentially distributed with a mean parameter $\alpha_{i,j} = \alpha_{\text{NLOS}}$. The value of α_{NLOS} depends on the NLOS propagation environment [40–42].

In most cases, prior information of NLOS connections, such as NLOS distribution, is unavailable. In this paper, our model was designed for such tough scenarios, i.e., NLOS connections being

unidentifiable among all connections. First, since we could not distinguish which connections were NLOS, we assumed that all range measurements were NLOS. Model $r_{i,j}$ was modified as

$$r_{i,j} = d_{i,j} + \varepsilon_{i,j} + n_{i,j}, (i, j) \in \mathcal{Z}_{\text{NLOS}} \cup \mathcal{Z}_{\text{LOS}} \quad (3)$$

2.2. Convex Relaxation

The method in Reference [35] provided two bounds of $d_{i,j}$ in NLOS environments. The upper bound is

$$u_{i,j} = r_{i,j} + 2\sigma_{i,j} \geq d_{i,j} \quad (4)$$

the lower bound is

$$l_{i,j} = r_{i,j} - 4\sigma_{i,j} \leq d_{i,j} \quad (5)$$

For the single-constraint case, the upper and lower bounds provide an annular feasible region. Heuristic localization estimation lies on the circle with a radius of $(u_{i,j} + l_{i,j})/2$. Hence, the optimization problem of cooperative localization in an NLOS environment is modified as:

$$\hat{\mathbf{X}}_s = \underset{\mathbf{X}_s}{\operatorname{argmin}} \sum_{(i,j) \in \mathcal{Z}_S \cup \mathcal{Z}_A} \frac{1}{2} \left(\|\mathbf{x}_i - \mathbf{x}_j\| - \frac{1}{2}(l_{i,j} + u_{i,j}) \right)^2 \quad (6)$$

This model takes NLOS impact into account and uses heuristic points $(u_{i,j} + l_{i,j})/2$ to replace range measurements $r_{i,j}$. In severe environments where most connections are NLOS, range measurements may contain much false information and, thus, cannot directly be used as distance metrics. In Model (6), the bounds can neutralize most NLOS errors in range measurements. However, Problem (6) is nonconvex. In this paper, we propose a tight relaxation method rather than SDP. We used projection relaxation to obtain the convex envelope of the original nonconvex problem. The theoretical derivation is given as follows.

First, we rewrote the fundamental part of Problem (6) in the form of a squared distance of the projection on a certain set by using Cauchy–Schwartz inequality.

The proof is given as follows:

If $\|\mathbf{b}\| = d_0$, we have

$$\begin{aligned} (\|\mathbf{a}\| - d_0)^2 &= \|\mathbf{a}\|^2 - 2\|\mathbf{a}\|\|\mathbf{b}\| + \|\mathbf{b}\|^2 \\ &\leq \|\mathbf{a}\|^2 - 2\mathbf{a}^T\mathbf{b} + \|\mathbf{b}\|^2 \\ &= \|\mathbf{a} - \mathbf{b}\|^2 \end{aligned} \quad (7)$$

Then, function

$$f(\mathbf{a}, d) = (\|\mathbf{a}\| - d_0)^2 \quad (8)$$

can be rewritten as

$$f(\mathbf{a}, d) = \inf_{\|\mathbf{b}\|=d_0} \|\mathbf{a} - \mathbf{b}\|^2 = \operatorname{dist}^2(\mathbf{a}, \mathbf{B}) \quad (9)$$

where $\mathbf{B} = \{\mathbf{b} \mid \|\mathbf{b}\| = d_0\}$ is a nonconvex set. Hence, the original problem can be written in a simpler form:

$$\min \sum_{(i,j) \in \mathcal{Z}_S \cup \mathcal{Z}_A} \frac{1}{2} \operatorname{dist}^2(\mathbf{x}_i - \mathbf{x}_j, \mathbf{B}_{i,j}) \quad (10)$$

where set $\mathbf{B}_{i,j}$ is a spherical surface depending on the bounds, i.e., $\mathbf{B}_{i,j} = \{\mathbf{b} \mid \|\mathbf{b}\| = \frac{1}{2}(l_{i,j} + u_{i,j})\}$.

A feasible approach is to find the convex envelope by relaxing constraint $\|\mathbf{b}\| = d_0$ to $\|\mathbf{b}\| \leq d_0$. Therefore, we obtained the convex form based on Function (8).

$$\tilde{f}(\mathbf{a}, d) = \inf_{\|\mathbf{b}\| \leq d_0} \|\mathbf{a} - \mathbf{b}\|^2 = \text{dist}^2(\mathbf{a}, C) \quad (11)$$

where $C = \{\mathbf{b} \mid \|\mathbf{b}\| \leq d\}$ is the convex hull of the non-convex set B.

According to Equation (11), we could establish the convex model corresponding to Problem (10):

$$\min \sum_{(i,j) \in \mathcal{Z}_S \cup \mathcal{Z}_A} \frac{1}{2} \text{dist}^2(\mathbf{x}_i - \mathbf{x}_j, C_{i,j}) \quad (12)$$

where $C_{i,j} = \{\mathbf{b} \mid \|\mathbf{b}\| \leq \frac{1}{2}(l_{i,j} + u_{i,j})\}$ is a convex set.

3. Distributed Framework

In order to approach the solution of convex optimization Problem (12) in a distributed way, we built the consensus form to formulate Problem (12) and design a parallel distributed algorithm based on ADMM. Problem (12) cannot be directly applied to parallel ADMM because it involves matrix calculation. We built the consensus form to decompose large-scale Problem (12) into $N + M$ subproblems with independent variables. Then, we could design the ADMM framework to solve these subproblems in a parallel way. In this section, we prove the feasibility of decomposing Problem (12) and provide the concrete procedure of the parallel distributed algorithm.

The consensus form could independently represent the connections between each sensor and its neighbors in the whole network by duplicating a new vector of each node j at neighbor node i .

Let $\mathbf{z}_i = \left[(z_i)_j^T \right]^T \in \mathbb{R}^{(\hat{\mathcal{N}}_i+1)d}$, $i \in \mathcal{S} \cup \mathcal{A}$ be the local variable, where $\hat{\mathcal{N}}_i$ represents the length of \mathcal{N}_i . Component $(z_i)_j$ has the same dimension as \mathbf{x}_j . Each of these local variables consists of selected elements of global variable $\mathbf{x} = [\mathbf{x}_1^T, \mathbf{x}_2^T, \dots, \mathbf{x}_{N+M}^T]^T$. Mapping from the local variable to the global variable is

$$\begin{cases} (z_i)_1 = \mathbf{x}_i & i \in \mathcal{S} \cup \mathcal{A}, l \in \mathcal{N}_i, j = 2, \dots, \hat{\mathcal{N}}_i + 1 \\ (z_i)_j = \mathbf{x}_l \end{cases} \quad (13)$$

when we put the elements of neighbor set \mathcal{N}_i in corresponding vector \mathbf{N}_i in an ascending order, the relation between $(z_i)_j$ and \mathbf{x}_l is $l = \mathbf{N}_i(j - 1)$. $\mathbf{N}_i(j)$ means the j th element of \mathbf{N}_i . In other words, $(z_i)_j$ is the duplication of $(j - 1)$ th neighbor node at node i .

Therefore, Problem (12) has an equivalent form:

$$\begin{aligned} \min & \sum_{i=1}^{N+M} \sum_{j=2}^{\hat{\mathcal{N}}_i+1} \frac{1}{2} \text{dist}^2((z_i)_1 - (z_i)_j, C_{i,l}) \\ \text{s.t.} & \mathbf{z}_i - \tilde{\mathbf{x}}_i = 0, \forall i \in \mathcal{S} \cup \mathcal{A} \\ & (z_k)_1 = \mathbf{x}_k, \forall k \in \mathcal{A} \end{aligned} \quad (14)$$

where $\tilde{\mathbf{x}}_i$ is the linear function of \mathbf{x} .

We denoted local objective function as $F_i(z_i) = \sum_{j=2}^{\hat{N}_i+1} \frac{1}{2} \text{dist}^2((z_i)_1 - (z_i)_j, C_{i,l})$. Obviously, total objective function $F(z) = \sum_{i=1}^{N+M} F_i(z_i)$ is separable. Then, we could rewrite Problem (14) in another, simpler form:

$$\begin{aligned} \min \quad & \sum_{i=1}^{N+M} F_i(z_i) \\ \text{s.t.} \quad & z_i - A_i x = 0, \forall i \in \mathcal{S} \cup \mathcal{A} \\ & z \in \mathcal{R}_z \end{aligned} \tag{15}$$

where \mathcal{R}_z is a linear space satisfying

$$\mathcal{R}_z = \{z | (z_k)_1 = x_k, \forall k \in \mathcal{A}\} \tag{16}$$

Matrix A_i indicates the linear relation between local variable z_i and global variable x , of which the form is

$$A_i = \begin{bmatrix} e_1 \\ (e_l)_{\forall l \in \mathcal{N}_i} \end{bmatrix} \otimes I_d \tag{17}$$

where e_l is the l th unit-row vector in \mathbb{R}^{N+M} , I_d is an identity matrix of order d , and \otimes is a Kronecker product.

By introducing an indicator function $G_z(z)$ of \mathcal{R}_z , Problem (14) can be rewritten in a general consensus form as

$$\begin{aligned} \min \quad & F(z) + G_z(z) \\ \text{s.t.} \quad & z - Ax = 0 \end{aligned} \tag{18}$$

where z is the shorthand notation for $[z_1^T, z_2^T, \dots, z_{N+M}^T]^T$, and $A = [A_1; A_2; \dots; A_{N+M}]$.

Before deriving the ADMM framework, we needed to build the augmented Lagrangian function of Problem (18)

$$\begin{aligned} L(z, x, \lambda; c) &= [F(z) + G_z(z)] + \lambda^T(z - Ax) + \sum_i \frac{1}{2} c \|z_i - A_i x\|^2 \\ &= \sum_i [F_i(z_i) + G_{z_i}(z_i) + \lambda_i^T(z_i - A_i x) + \frac{1}{2} c \|z_i - A_i x\|^2] \end{aligned} \tag{19}$$

where λ is a vector consisting of Lagrangian multipliers, and c is the given penalty parameter. Indicator function $G_z(z)$ can be written as the sum of numerous subfunctions, namely, $G_z(z) = \sum_i G_{z_i}(z_i)$, by decomposing linear space \mathcal{R}_z as the Cartesian product of $(N + M)$ closed convex sets $\mathcal{R}_{z_1}, \mathcal{R}_{z_2}, \dots, \mathcal{R}_{z_{N+M}}$:

$$\mathcal{R}_z = \mathcal{R}_{z_1} \times \mathcal{R}_{z_2} \times \dots \times \mathcal{R}_{z_N} \times \mathcal{R}_{z_{N+1}} \times \dots \times \mathcal{R}_{z_{N+M}}$$

Indicator functions of sets $\mathcal{R}_{z_1}, \dots, \mathcal{R}_{z_N}$ equal to identity matrices. Convex sets $\mathcal{R}_{z_{N+1}}, \dots, \mathcal{R}_{z_{N+M}}$ have the affine form of

$$\mathcal{R}_{z_k} = \{z_k | T_k z_k = x_k\}, \forall k \in \mathcal{A} \tag{20}$$

where T_k is a matrix satisfying $T_k = \left(1, \underbrace{0, \dots, 0}_{\mathcal{N}_k} \right)^T \otimes I_d, k \in \mathcal{A}$.

Thus, we can conclude that the objective function, constraints, and penalty term are separable for each sensor. Distributed optimization could be obtained by applying the ADMM method to

Problem (18). The local updates are handled in an alternate way that can be separately optimized for variables z, x , and λ , as follows, for $\forall i \in \mathcal{S} \cup \mathcal{A}$

$$\begin{aligned} z_i^{t+1} &\in \operatorname{argmin}_{z_i} \left[F_i(z_i) + G_{z_i}(z_i) + \frac{1}{2}c\|z_i - A_i x^t + \tilde{\lambda}_i^t\|^2 \right] \\ x^{t+1} &\in \operatorname{argmin}_x \left(\sum_i \frac{1}{2}c\|z_i^{t+1} - A_i x + \tilde{\lambda}_i^t\|^2 \right) \\ \tilde{\lambda}_i^{t+1} &\in \tilde{\lambda}_i^t + c(z_i^{t+1} - A_i x^{t+1}) \end{aligned} \tag{21}$$

where $\tilde{\lambda}_i = \lambda_i/c$ is the scaled dual variable.

z_i -updates and $\tilde{\lambda}_i$ -updates can be independently carried out in parallel for each node i . The concrete implementation steps are shown as follows.

- (1) Initialize variables $z_i^0, x^0, \tilde{\lambda}_i^0$ for all nodes.
- (2) Each node updates its local variables z_i^{t+1} according to the first step of Problem (21), corresponding to the action

$$z_i^{t+1} \in \operatorname{argmin}_{z_i} \left[F_i(z_i) + G_{z_i}(z_i) + \frac{1}{2}c\|z_i - A_i x^t + \tilde{\lambda}_i^t\|^2 \right] \tag{22}$$

If we put indicate function $G_{z_i}(z_i)$ in the constraints, Problem (22) can be written in an equivalent form:

$$\begin{aligned} \min & \left[\sum_{j=1}^{N_i} \frac{1}{2} \operatorname{dist}^2((z_i)_1 - (z_i)_j, C_{i,j}) + \frac{1}{2}c\gamma_i \right] \\ \text{s.t.} & \|z_i - A_i x^t + \tilde{\lambda}_i^t\|^2 \leq \gamma_i \\ & \gamma_i \geq 0 \\ & z_i \in \mathcal{R}_{z_i} \end{aligned} \tag{23}$$

where γ_i are the slack variables working as the quadratic penalty in the local cost function.

Obviously, Problem (23) is untypically convex. The gradient of $F_i(z_i)$ is discontinuous. Hence, classical methods such as the interior point method and the Newton method are not appropriate to solve these problems. In Section 4, we further derive an iterative algorithm to approach the solution of Problem (23). This iterative algorithm has the same convergence rate with the interior point method.

- (3) Broadcast local variables z_i^{t+1} to all nodes of neighbor sets.
- (4) Each node computes variables \tilde{x}_i^{t+1} according to

$$x^{t+1} \in \operatorname{argmin}_x \left(\sum_i \frac{1}{2}c\|z_i^{t+1} - A_i x + \tilde{\lambda}_i^t\|^2 \right) \tag{24}$$

Problem (24) is actually easy to solve by averaging all components of z_i^{k+1} of which the local indices correspond to global index l . The x -update step has a simpler solution:

$$x_l^{t+1} = \frac{1}{k_l} \left((z_l)_1^{t+1} + \sum_{(i,j) \in \mathcal{Z}_{x_l}} (z_i)_j^{t+1} \right) \tag{25}$$

where k_l is the number of local variable components corresponding to global components z_l and $\mathcal{Z}_{x_l} = \{(i, j) | N_i(j-1) = l\}$.

- (5) Each node updates dual variables $\tilde{\lambda}_i^{t+1}$ according to

$$\tilde{\lambda}_i^{t+1} \in \tilde{\lambda}_i^t + c(z_i^{t+1} - A_i x^{t+1}) \tag{26}$$

A summary of the proposed parallel algorithm is illustrated as Algorithm 1 and Figure 2:

Algorithm 1 ADMM_Projection (ADMM_P) algorithm for Problem (18).

Input: number of anchors M , location of anchors $x_k \in \mathbb{R}^2, k \in \mathcal{A}$, number of source sensors N , range measurements $\{r_{i,j}, (i, j) \in \mathcal{Z}_{\text{NLOS}} \cup \mathcal{Z}_{\text{LOS}}\}$;

Output: $(z_i)_1, i \in \mathcal{S}$

1. **Initialization:** $t = 0$;

- (1) Initialize local location of sensors x^0 and set dual variables λ_i^0 to zero;
- (2) According to x_i^0 and λ_i^0 , compute the initial values:

$$z_i^0 = A_i x^0$$

2. **Updating iteration:** $t++$

- (1) Update local location variables z_i^t at each node: $\forall i \in \mathcal{A} \cup \mathcal{S}$

$$z_i^{t+1} \in \underset{z_i}{\operatorname{argmin}} \left[F_i(z_i) + G_{z_i}(z_i) + \frac{1}{2}c \|z_i - A_i x^t + \tilde{\lambda}_i^t\|^2 \right]$$

- (2) Broadcast local variable z_i^{t+1} to its neighbors \mathcal{N}_i ;
 - (3) Update variable x according to Problem (25);
 - (4) Update the local dual variable $\tilde{\lambda}_i^{t+1}$ according to Problem (26).
-

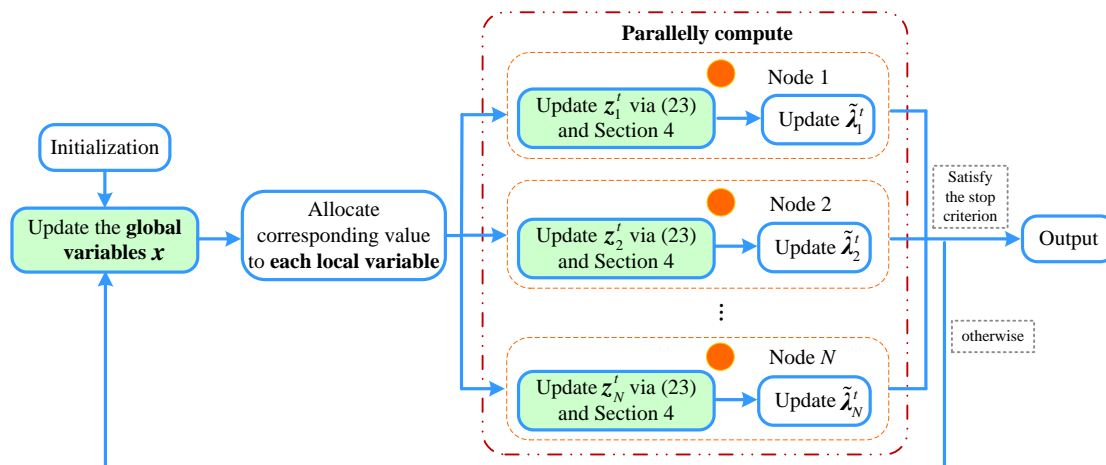


Figure 2. Flow chart of the ADMM_P algorithm.

4. Solution to the Subproblem

Section 3 implies that the ADMM_P algorithm is an iterative method. Reference [27] guarantees that our algorithm converges to the same minimum as that in the centralized framework because its original counterpart (12) is convex. The algorithm summary indicates that the total computational complexity of ADMM_P almost relies on the local z -update (22). To satisfy the requirements on processing speed, we needed to design an effective algorithm that had both a fast convergence rate and low complexity. The Newton method and interior point method have these properties, but they were designed for smooth optimization problems [43]. Unfortunately, Problem (22) is a nonquadratic differential. Hence, we designed an iterative method based on the accelerated proximal algorithm, which is an analogous tool for nonsmooth optimization problems [21,44]. Given the structure of WSNs, we derived the proximal algorithm for source nodes and anchors, respectively.

4.1. Lipschitz Constant

The proximal algorithm needs the guarantee of Lipschitz continuity. First, we prove the premise in this subsection. To simplify notations, we defined the functions:

$$\begin{aligned}
 h(z_i) &= \frac{1}{2} \text{dist}^2(\mathbf{H}_i z_i, C_i) = \sum_{j=2}^{\mathcal{N}_i+1} \frac{1}{2} \text{dist}^2((z_i)_1 - (z_i)_j, C_{i,l}) \\
 g(z_i) &= \frac{1}{2} c \|z_i - \mathbf{A}_i x^t + \tilde{\lambda}_i^t\|^2
 \end{aligned}
 \tag{27}$$

where C_i is Cartesian product of balls $C_{i,l}$, $l \in \mathcal{N}_i$, and \mathbf{H}_i has the form of

$$[\mathbf{1}_{\mathcal{N}_i}, -\mathbf{I}_{\mathcal{N}_i}] \otimes \mathbf{I}_d
 \tag{28}$$

Function $g(z_i)$ in Functions (27) is convex and differentiable, and its gradient is

$$\nabla g(z_i) = c [z_i - (\mathbf{A}_i x^t - \tilde{\lambda}_i^t)]
 \tag{29}$$

Obviously, this gradient is Lipschitz-continuous, and we could obtain its Lipschitz constant as follows

$$\begin{aligned}
 \|\nabla g(x) - \nabla g(y)\| &= \|cx - cy\| \\
 &\leq k_g |c| \|x - y\| \\
 \Rightarrow \frac{\|\nabla g(x) - \nabla g(y)\|}{\|x - y\|} &\leq k_g |c| = L_g, \forall x, y \in \mathbb{R}^{d(\mathcal{N}_i+1)}
 \end{aligned}
 \tag{30}$$

where $k_g \geq 1$ is a constant.

4.2. Source Nodes

For source nodes, local Subproblem (22) has the equivalent simpler form

$$\min h(z_i) + g(z_i)
 \tag{31}$$

the proximal mapping of function $h(z_i) = \frac{1}{2} \text{dist}^2(\mathbf{H}_i z_i, C_i)$ is

$$\text{prox}_{\mu h}(v_i) = \mathbf{H}_i^\dagger \left[\mathbf{H}_i v_i + \frac{\mu}{1 + \mu} (\mathbf{P}_{C_i}(\mathbf{H}_i v_i) - \mathbf{H}_i v_i) \right]
 \tag{32}$$

where $\mu = 1/L_g$ is the step size, \mathbf{H}_i^\dagger is the pseudoinverse of \mathbf{H}_i , and $\mathbf{P}_{C_i}(\mathbf{H}_i z_i)$ is the orthogonal projection of point $\mathbf{H}_i z_i$ onto the convex set C_i , i.e.,

$$\mathbf{P}_{C_i}(\mathbf{H}_i z_i) \in \arg \min \{ \|x - \mathbf{H}_i z_i\| \mid x \in C_i \}
 \tag{33}$$

Hence, we can obtain the accelerated proximal algorithm for local Subproblems (31), as follows, for $k \geq 1$:

$$\begin{aligned}
 \omega_i^{(k-1)} &= \zeta_i^{(k-1)} - \mu_k \nabla g(\zeta_i^{(k-1)}) \\
 &= \zeta_i^{(k-1)} - \mu_k c \left[\zeta_i^{(k-1)} - (\mathbf{A}_i z^t - \tilde{\lambda}_i^t) \right] \\
 \eta_i^{(k)} &= \frac{1}{1 + \mu_k} \omega_i^{(k-1)} + \frac{\mu_k}{1 + \mu_k} \mathbf{H}_i^\dagger \mathbf{P}_{C_i}(\mathbf{H}_i \omega_i^{(k-1)}) \\
 \zeta_i^{(k)} &= \eta_i^{(k)} + \frac{k-1}{k+2} (\eta_i^{(k)} - \eta_i^{(k-1)})
 \end{aligned}
 \tag{34}$$

This iterative procedure stops when the stopping criterion is satisfied. μ_k is the step size. Here, we used the fixed step size $\mu_k = \mu = 1/L_g$. We used the estimation results of the last loop as the initial point of this subupdate, i.e., $\xi_i^{(0)} = z_i^t$. The subupdate outputs $\xi_i^{(k)} = z_i^{t+1}$ as the local estimation result.

4.3. Anchor Nodes

For anchor nodes $(z_i)_1 = x_i, i \in \mathcal{A}$, Problem (22) becomes an equality-constrained optimization problem for $i \geq N + 1$,

$$\begin{aligned} & \min h(z_i) + g(z_i) \\ & \text{s.t. } T_i z_i = x_i \end{aligned} \quad (35)$$

Our approach solving this problem was to eliminate the equality constraint via the equation

$$\{z_i | T_i z_i = x_i\} = \{Q_i \beta_i + \hat{z}_i | \beta_i \in \mathbb{R}^{\mathcal{N}_i d}\} \quad (36)$$

where $\hat{z}_i = \left(x_i^T, \underbrace{0, \dots, 0}_{\mathcal{N}_i d} \right)^T$, Q_i has the form of

$$Q_i = \begin{bmatrix} e_1^d - e_2^d & \mathbf{0}_1 \\ \mathbf{0}_2 & I_{\mathcal{N}_i d} \end{bmatrix} \quad (37)$$

where $e_i^d, i = 1$ or 2 is the i th unit vector in \mathbb{R}^d , $I_{\mathcal{N}_i d}$ is an identity matrix with order $\mathcal{N}_i d$, and $\mathbf{0}_i, i = 1$ or 2 is a zero matrix.

With Equation (36), we form eliminated optimization problem

$$\begin{aligned} & \min \hat{h}(\beta_i) + \hat{g}(\beta_i) \\ & \Rightarrow \min h(Q_i \beta_i + \hat{z}_i) + g(Q_i \beta_i + \hat{z}_i) \end{aligned} \quad (38)$$

which is an unconstrained problem with variable β_i . Then, we could derive the solution of equality-constrained Problem (35) according to Equations (34).

Introducing the affine structure does not change the convexity and continuity of the functions. The gradient of the $\hat{g}(\beta_i)$ function is still Lipschitz-continuous. The gradient of $\hat{g}(\beta_i)$ is

$$\nabla \hat{g}(\beta_i) = c \left[Q_i^T Q_i \beta_i + Q_i^T (\hat{z}_i - A_i x^t + \tilde{\lambda}_i^t) \right] \quad (39)$$

Hence, the new Lipschitz constant could be obtained by the following inequality:

$$\begin{aligned} \|\nabla \hat{g}(x) - \nabla \hat{g}(y)\| &= \|c Q_i^T Q_i x - c Q_i^T Q_i y\| \\ &= |c| \|Q_i^T Q_i (x - y)\| \\ &\leq |c| \nu_{\max}(Q_i^T Q_i) \|x - y\| \\ \Rightarrow \frac{\|\nabla \hat{g}(x) - \nabla \hat{g}(y)\|}{\|x - y\|} &\leq |c| \nu_{\max}(Q_i^T Q_i) = L_{\hat{g}}, \forall x, y \in \mathbb{R}^{d \mathcal{N}_i} \end{aligned} \quad (40)$$

where $\nu_{\max}(Q_i^T Q_i)$ is the maximal eigenvalue of $Q_i^T Q_i$.

By using the conclusion of Equation (32) to Equation (34), the accelerated proximal algorithm for the local subproblem of the anchor nodes is presented as follows:

$$\begin{aligned}\omega_i^{(k-1)} &= \zeta_i^{(k-1)} - \mu_k c [\mathbf{Q}_i^T \mathbf{Q}_i \zeta_i^{(k-1)} + \mathbf{Q}_i^T (\hat{\mathbf{z}}_i - \mathbf{A}_i \mathbf{x}^t + \tilde{\lambda}_i^t)] \\ \eta_i^{(k)} &= \frac{\hat{\mu}_k}{1 + \hat{\mu}_k} (\omega_i^{(k-1)} + \mathbf{Q}_i^+ \hat{\mathbf{z}}_i) + \frac{\hat{\mu}_k}{1 + \hat{\mu}_k} (\mathbf{H}_i \mathbf{Q}_i)^+ \mathbf{P}_{C_i} (\mathbf{H}_i (\mathbf{Q}_i \omega_i^{(k-1)} + \hat{\mathbf{z}}_i)) \\ \zeta_i^{(k)} &= \eta_i^{(k)} + \frac{k-1}{k+2} (\eta_i^{(k)} - \eta_i^{(k-1)})\end{aligned}\quad (41)$$

5. Numerical Simulations

In this section, we present the simulation results to demonstrate the strengths and weaknesses of the ADMM_P algorithm. We considered the worst case, where NLOS connections are unidentifiable and the probability of these NLOS connections is up to 95%. We preliminarily considered a network that had 10 anchor nodes and 40 source nodes randomly distributed in a $[-50 \text{ m}, 50 \text{ m}] \times [-50 \text{ m}, 50 \text{ m}]$ square. Noisy range measurements were available when distances $r_{i,j} \leq 30 \text{ m}$. The coordinate dimension is $d = 2$. Range measurements were generated according to Equations (1) and (2), where NLOS error $\varepsilon_{i,j}$ is exponentially distributed with mean parameter α_{NLOS} . For a noisy environment, we assumed that the measurement noise was independent and identically distributed, i.e., $\sigma_{i,j} = \sigma_n$.

We compared the accuracy of ADMM_P with four state-of-art algorithms, which are named:

- (1) SDP: SDP method of Reference [36] designed for NLOS environments;
- (2) SDPH: SDP model based on a heuristic solution [35];
- (3) EDM: EDM model based on three-block ADMM [37];
- (4) ADMM_SF: the parallel algorithm of Reference [20], which was only designed for LOS environments.

It is worth noting that ADMM_P is an iterative algorithm working in the parallel distributed framework, and that there are few authoritative publications about distributed algorithms for NLOS localization. Therefore, we only analyzed the convergence property for our algorithm. In the convergence analysis, we provide the estimation results of SDPH of Reference [35] as references.

All simulations were carried out in MATLAB. Local convex Problems (22) were handled via the MATLAB Optimization Toolbox with *fmincon* solver. Algorithm performance was evaluated via Cumulative Distribution Function (CDF), Root Mean Square Error (RMSE), and objective Function Value (FVAL):

$$\begin{aligned}\text{RMSE} &= \sqrt{\frac{\sum_{i \in \mathcal{S}} \|(z_i)_1 - p_i\|^2}{N}} \\ \text{FVAL} &= \sum_{i \in \mathcal{S}} \left[F_i(z_i) + \frac{1}{2} c_i \|z_i - \mathbf{A}_i \mathbf{x} + \tilde{\lambda}_i\|^2 \right]\end{aligned}\quad (42)$$

where p_i is the true position, and $(z_i)_1$ is the estimated location of node i .

5.1. Performance Comparison

In this part, we verify the superiority of our method in the aspects of localization accuracy and computational complexity, respectively. We fixed the number of source nodes and anchors as $N = 40$, $M = 10$. The mean parameter of NLOS propagation is $\alpha_{\text{NLOS}} = 3$. Sensor nodes (including anchors) were randomly distributed in the square of $[-50 \text{ m}, 50 \text{ m}] \times [-50 \text{ m}, 50 \text{ m}]$. To show the simulation results more precisely, we compared the ADMM_P algorithm with three other state-of-art methods in the same simulation environment.

5.1.1. Accuracy Comparison

Figures 3 and 4 show the simulation results of the four available methods. As is shown, ADMM_P had higher accuracy than other three methods. In Figure 3, the CDF curve of ADMM_P is in the leftmost side of the three other methods. This means that the estimation results of ADMM_P were more concentrated around lower errors. Especially at point Error = 2.5 m, the CDF of our method improved by up to 50% compared with the ADMM_SF, which did not utilize any NLOS mitigation techniques. In Figure 4, we further analyzed the performance of the four methods in varying environments with different σ_n and α_{NLOS} , respectively. The simulation results show that ADMM_P has higher accuracy, and performance is more stable in different environments.

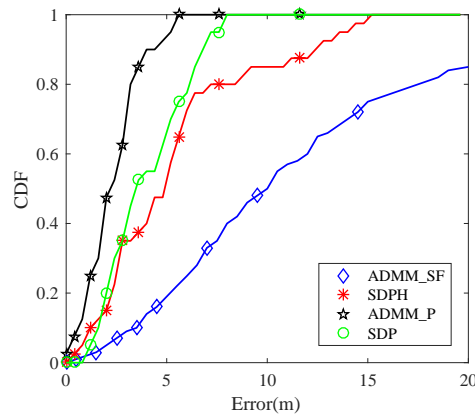


Figure 3. Cumulative Distribution Function (CDF) comparison of the four aforementioned methods for a $M = 10, N = 40$ sensor network. Anchors were randomly distributed. Probability of NLOS connections was set to $P_{NLOS} = 95\%$.

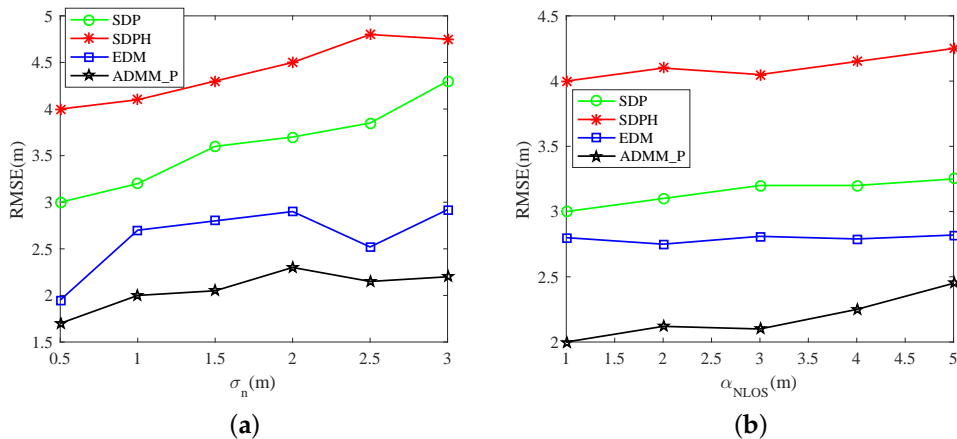


Figure 4. Performance comparison in varying environments with a $M = 10, N = 40$ sensor network. Probability of NLOS connections was set to $P_{NLOS} = 95\%$. (a) Root Mean Square Error (RMSE) comparison in varying noise environments; (b) RMSE comparison with different NLOS parameters.

5.1.2. Complexity Analysis

We compared computational complexity in Table 1, in which we used N_c to represent the connective numbers of pair nodes $(i, j) \in \mathcal{Z}_S \cup \mathcal{Z}_A$. We could find that ADMM_P is more suitable for practical situations because it has much lower computational complexity and smaller problem size. The parallel distributed framework plays a key role in decreasing complexity.

The SDPH method in Reference [35] and the SDP method in Reference [36] were implemented in the centralized framework. The problem sizes of SDPH and SDP were, respectively, $dN + 1/2 * N(N + 1)$

and $dN + 1/2N(N + 1) + N^2$, both depending on the whole scale of the WSN N . The EDM method in Reference [37] applied a three-block ADMM, but problem size also depends on N . According to Table 1, we can see that the problem size of the ADMM_P algorithm was determined by the decomposed local minimization and could be reduced to $d(\hat{N}_i + 1) + \hat{N}_i$, where \hat{N}_i represents the number of neighbors. Because of the limitation of r_μ , the size of the neighbor set was much smaller than the whole scale of the WSN, i.e., $\hat{N}_i \ll N$. This conclusion can theoretically prove that our method can drastically reduce the preprocessing burden, no matter the problem size or computational complexity.

Our method decomposes the original large-scale problem into $N + M$ small-scale subproblems via consensus form, and handles these subproblems in a parallel way. Hence, when compared with the centralized algorithm, our method can dramatically decrease computational complexity. In Sections 3 and 4, we could find that the total complexity of the proposed algorithm almost depends on solving the subproblems. Section 4 provides an iterative method to solve the subproblems, which are nonquadratic differentials. In this iterative method, with the guarantee of objective-function convexity, the optimal value is finite and attained at z_i^* . The derived method is known to converge at a rate of $O(1/k^2)$ [44]:

$$F_i(z_i^{(k)}) - F_i^* \leq \frac{2}{(k+1)^2\mu} \|z_i^{(0)} - z_i^*\|^2 \quad (43)$$

Subupdates require very simple operations, no matter if they are at anchor nodes or at source nodes. The inversion of matrix H_i and $H_i Q_i$ is easy to precompute because H_i and Q_i only consist of identity matrices and unit vectors. Projection on set C_i can be decomposed into projections on sets $C_{i,j}$, $j \in \mathcal{N}_i$, which involves very limited dimension $d \leq 3$ and is easy to compute. Therefore, the local update given by Equations (34) and (41) has much lower complexity than the three other algorithms. However, because every node shares the processing burden and plays the same role in the parallel framework, the communication cost of each node is higher than that of centralized frameworks.

Table 1. Complexity comparison of available algorithms.

	SDPH [35]	SDP [36]	EDM [37]	ADMM_P
Convex problem size	$dN + 1/2N(N + 1)$	$dN + 1/2N(N + 1) + N^2$	$N + M(M - 1)/2 + N_c$	$d(\hat{N}_i + 1)$
Computational complexity	$O(N^3)$	$O((2N)^3)$	$O(N^3 + N^2)$	$O(d^3 + (d\hat{N}_i)^2)$
Type of framework	Centralized SDP	Centralized SDP	Three-block ADMM	Parallel ADMM
Communication cost	N	N	N	$d(\hat{N}_i + 1)$

5.2. ADMM_P Properties in Different Noisy Environments

In this scenario, we fixed the number of anchors and the distribution parameter of NLOS errors to show localization accuracy and convergence property in different noisy environments. The mean parameter of NLOS error was fixed as $\alpha_{\text{NLOS}} = 3$. Ten anchors were randomly deployed in the region of 100×100 m. As is shown in Figure 5a,b, we present the simulation results with $\sigma_n^2 = 0.01 \times 100$ and $\sigma_n^2 = 0.1 \times 100$. We found that ADMM_P mitigates the effect of NLOS range measurements and provides better estimation accuracy. SDPH is solved with the interior-point method under a centralized framework, so its simulation results are only provided as accuracy references in Figure 5. From the figure, we can see that ADMM_P has good convergence performance in different noisy environments and higher accuracy than SDPH. In Figure 5b, we further analyzed the convergence performance of ADMM_P. Given that the local objective functions are based on heuristic solutions, the function value cannot converge to an extremely low level. Hence, the curve tendencies in Figure 5b still demonstrate a stable convergence of ADMM_P.

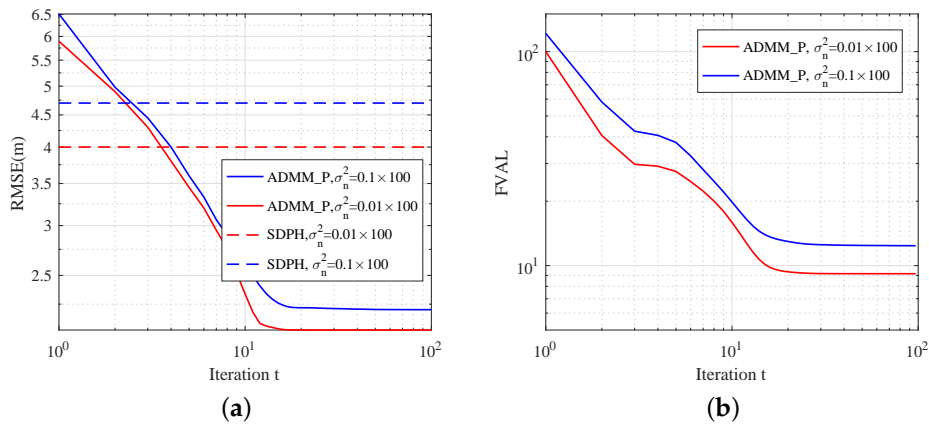


Figure 5. Estimation results with different σ_n^2 for $N = 40, M = 10$ sensor network versus iteration number t . Probability of NLOS connections was set to $P_{NLOS} = 95\%$. (a) RMSE; (b) convergence properties.

5.3. ADMM_P Properties in Different NLOS Environments

In this scenario, we fix the number of anchors and the noise level to study ADMM_P performance with different NLOS error distributions. The measurement noise variance was fixed as $\sigma_n^2 = 0.01 \times 100$. Other simulation settings were the same as those in Scenario B. In Figure 6a,b, we compared the performance of ADMM_P with $\alpha_{NLOS} = 3, 4, 5$. As we can see, although RMSE slightly increased as mean parameter α_{NLOS} grew, the accuracy of ADMM_P is still better than SDPH. This also indicates that the proposed algorithm effectively mitigates NLOS error. Simulation results in Figure 6b verify the stable convergence performance of ADMM_P as varying NLOS error distribution.

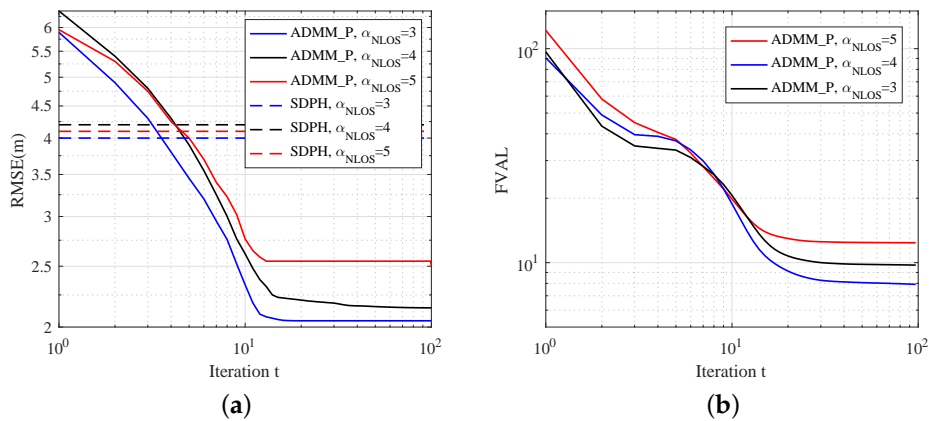


Figure 6. Estimation results with different α_{NLOS} for $N = 40, M = 10$ sensor network versus iteration number t . Probability of NLOS connections was set to $P_{NLOS} = 95\%$. (a) RMSE; (b) convergence properties.

5.4. ADMM_P Algorithm Properties with Different Anchor Placements

5.4.1. Anchor Placement

In this scenario, we deployed the anchors at the perimeter of the region. We set $\sigma_n^2 = 0.01 \times 100$ and $\alpha_{NLOS} = 3$. We compared localization accuracy when the anchor placement was random or fixed. In Reference [13], the author proposed that an appropriate anchor-placement design could further improve localization accuracy. Hence, we placed the anchors around the perimeter of the region to avoid getting crowded points, so that the higher dimensional projection of anchors could contain more

sensor nodes. The corresponding simulation results are shown in Figure 7b, from which we can see that the fixed anchor placement improves the localization performance of the ADMM_P algorithm in the same NLOS environment. Furthermore, in the scenario of fixed anchors, the accuracy of the ADMM_P algorithm is higher than that of the SDPH algorithm, even if the former employs the less anchors.

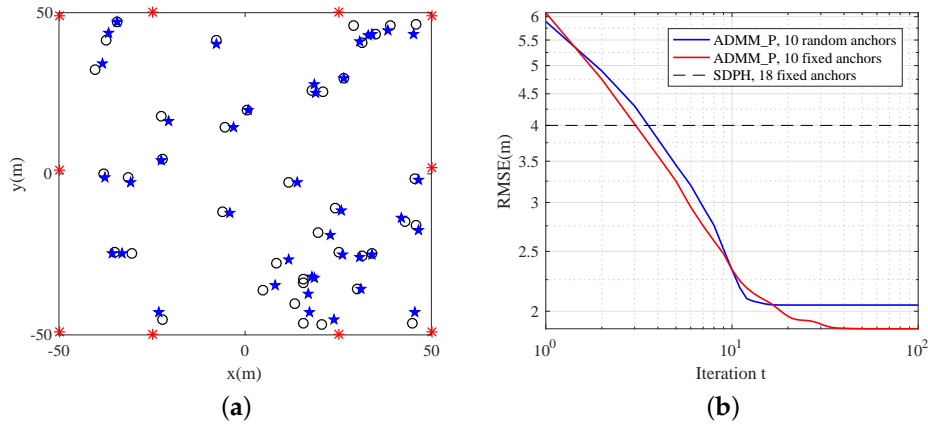


Figure 7. Estimation results of a sensor network with 10 fixed anchors and 40 randomly distributed sensors. (a) Estimation locations. Red asterisks *: anchor locations; black circles o: true sensor locations; blue stars *: estimation results; (b) RMSE comparison.

5.4.2. Varying Numbers of Anchors

Simulations in the last subsection show that a predesigned placement of anchors has a positive influence on localization performance. In this scenario, we evaluated the performance of the ADMM_P algorithm by varying the number of anchors. Taking the simulation results of Figure 7 into consideration, we fixed 10 anchors on the region boundary and randomly deployed the remaining anchors inside the region. Other simulation settings were the same as those in the last subsection. Node distribution is shown in Figure 8a, and the corresponding simulations are shown in Figure 8b, which shows that proper anchor placement and an increase in anchor number both lead to an improvement of localization performance.

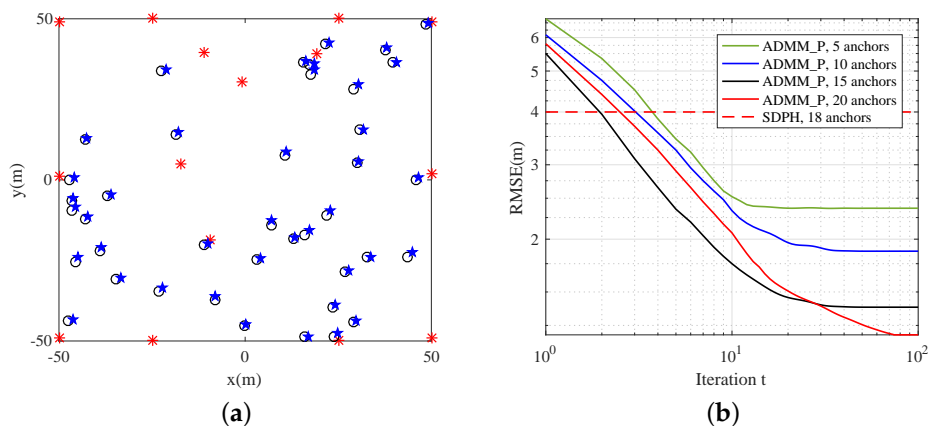


Figure 8. Estimation results of sensor network with 10 fixed anchors, five randomly distributed anchors, and 40 randomly distributed sensors. (a) Estimation locations. Red asterisks *: anchor locations; black circles o: true sensor locations; blue stars *: estimation results; (b) RMSE comparison.

6. Conclusions

In this paper, we proposed an efficient distributed algorithm for NLOS cooperative localization. We employed tight convex relaxation for the heuristic model, and approached it in a parallel ADMM framework to offer powerful computation ability. The distributed framework can drastically reduce the computational burden since it decomposes a large-scale problem into numerous small-scale problems, which similarly retains computational time when the number of sensors grows. Furthermore, we derived the accelerated proximal gradient method to improve the convergence rate of local subproblems. Simulation results demonstrate that our method efficiently alleviates the influence of NLOS propagation and has fairly good convergence performance.

In future works, we plan to study the level of influence that step size and penalty parameters could have on the performance of local subproblems and an ADMM framework. How to establish the mathematic model to find the optimal parameters needs further study as well. Moreover, the lower and upper bounds of range measurements provide a good feasible region, which is nevertheless nonconvex. We are interested in finding better convex relaxation rather than utilizing a heuristic solution as the optimal solution.

Author Contributions: Conceptualization, S.C.; data curation, C.X.; formal analysis, S.C.; investigation, Y.M.; methodology, S.C.; resources, Y.M. and Y.G.; supervision, J.Z.; validation, C.X. and Y.G.; writing—original draft, S.C.

Funding: This research received no external funding.

Conflicts of Interest: No conflict of interest exists in the submission of this manuscript, and the manuscript was approved by all authors for publication. I would like to declare on behalf of my coauthors that the described work was not previously published, and is not under consideration for publication elsewhere, in whole or in part.

References

1. Biswas, P.K.; Phoha, S. Self-organizing sensor networks for integrated target surveillance. *IEEE Trans. Comput.* **2006**, *55*, 1033–1047. [[CrossRef](#)]
2. Rätty, T.D. Survey on contemporary remote surveillance systems for public safety. *IEEE Trans. Syst. Cybern. Part C* **2010**, *40*, 493–515. [[CrossRef](#)]
3. Oh, S.; Schenato, L.; Chen, P.; Sastry, S. Tracking and coordination of multiple agents using sensor networks: System design, algorithms and experiments: Sensor networks that can rapidly locate, pursue and capture numerous moving targets can be used for surveillance of large areas. *Proc. IEEE* **2007**, *95*, 234–254. [[CrossRef](#)]
4. Liu, J.; Chu, M.; Reich, J.E. Multitarget Tracking in Distributed Sensor Networks. *IEEE Signal Process. Mag.* **2007**, *24*, 36–46. [[CrossRef](#)]
5. Leonard, N.E.; Paley, D.A.; Lekien, F.; Sepulchre, R.; Fratantoni, D.M.; Davis, R.E. Collective Motion, Sensor Networks, and Ocean Sampling. *Proc. IEEE* **2007**, *95*, 48–74. [[CrossRef](#)]
6. Sun, T.; Chen, L.J.; Han, C.C.; Gerla, M. Reliable sensor networks for planet exploration. In Proceedings of the IEEE Networking, Sensing & Control, Tucson, AZ, USA, 19–22 March 2005.
7. Arampatzis, T.; Lygeros, J.; Manesis, S. A Survey of Applications of Wireless Sensors and Wireless Sensor Networks. In Proceedings of the IEEE International Symposium on Mediterranean Conference on Intelligent Control, Limassol, Cyprus, 27–29 June 2005.
8. Maddumabandara, A.; Leung, H.; Liu, M. Experimental Evaluation of Indoor Localization Using Wireless Sensor Networks. *IEEE Sens. J.* **2015**, *15*, 5228–5237. [[CrossRef](#)]
9. Potdar, V.; Sharif, A.; Chang, E. Wireless Sensor Networks: A Survey. *Comput. Netw.* **2002**, *38*, 393–422.
10. Bardella, A.; Bui, N.; Zanella, A.; Zorzi, M. An Experimental Study on IEEE 802.15.4 Multichannel Transmission to Improve RSSI-Based Service Performance. In Proceedings of the Real-World Wireless Sensor Networks—International Workshop Realwsn 2010, Colombo, Sri Lanka, 16–17 December 2010; pp. 154–161.
11. Guvenc, I.; Chong, C.C. A Survey on TOA Based Wireless Localization and NLOS Mitigation Techniques. *IEEE Commun. Surv. Tutor.* **2009**, *11*, 107–124. [[CrossRef](#)]
12. Biswas, P.; Lian, T.C.; Wang, T.C.; Ye, Y. Semidefinite programming based algorithms for sensor network localization. *ACM Trans. Sens. Netw.* **2006**, *2*, 188–220. [[CrossRef](#)]

13. Biswas, P.; Liang, T.C.; Toh, K.C.; Ye, Y.; Wang, T.C. Semidefinite Programming Approaches for Sensor Network Localization With Noisy Distance Measurements. *IEEE Trans. Autom. Sci. Eng.* **2006**, *3*, 360–371. [[CrossRef](#)]
14. Wang, Z.; Zheng, S.; Ye, Y.; Boyd, S. Further Relaxations of the Semidefinite Programming Approach to Sensor Network Localization. *SIAM J. Optim.* **2008**, *19*, 655–673. [[CrossRef](#)]
15. Tomic, S.; Beko, M.; Rui, D. 3-D Target Localization in Wireless Sensor Network Using RSS and AoA Measurements. *IEEE Trans. Veh. Technol.* **2017**, *66*, 3197–3210. [[CrossRef](#)]
16. Tseng, P. *Second-Order Cone Programming Relaxation of Sensor Network Localization*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2007; pp. 156–185.
17. Oguz-Ekim, P.; Gomes, J.P.; Xavier, J.; Oliveira, P. Robust Localization of Nodes and Time-Recursive Tracking in Sensor Networks Using Noisy Range Measurements. *IEEE Trans. Signal Process.* **2011**, *59*, 3930–3942. [[CrossRef](#)]
18. Keller, Y.; Gur, Y. A Diffusion Approach to Network Localization. *IEEE Trans. Signal Process.* **2011**, *59*, 2642–2654. [[CrossRef](#)]
19. Korkmaz, S.; Veen, A.J.V.D. Robust localization in sensor networks with iterative majorization techniques. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Taipei, Taiwan, 19–24 April 2009; pp. 2049–2052.
20. Soares, C.; Xavier, J.; Gomes, J. Simple and Fast Convex Relaxation Method for Cooperative Localization in Sensor Networks Using Range Measurements. *IEEE Trans. Signal Process.* **2015**, *63*, 4532–4543. [[CrossRef](#)]
21. Chen, A.I.; Ozdaglar, A. A fast distributed proximal-gradient method. In Proceedings of the 2012 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton), Monticello, IL, USA, 1–5 October 2012; pp. 601–608.
22. Cheung, K.W.; So, H.C. A multidimensional scaling framework for mobile location using time-of-arrival measurements. *IEEE Trans. Signal Process.* **2005**, *53*, 460–470. [[CrossRef](#)]
23. Shi, Q.; He, C.; Chen, H.; Jiang, L. Distributed Wireless Sensor Network Localization Via Sequential Greedy Optimization Algorithm. *IEEE Trans. Signal Process.* **2010**, *58*, 3328–3340. [[CrossRef](#)]
24. Srirangarajan, S.; Tewfik, A.H.; Luo, Z.Q. Distributed sensor network localization using SOCP relaxation. *IEEE Trans. Wirel. Commun.* **2008**, *7*, 4886–4895. [[CrossRef](#)]
25. Chan, F.K.W.; So, H.C. Accurate Distributed Range-Based Positioning Algorithm for Wireless Sensor Networks. *IEEE Trans. Signal Process.* **2009**, *57*, 4100–4105. [[CrossRef](#)]
26. Simonetto, A.; Leus, G. Distributed Maximum Likelihood Sensor Network Localization. *IEEE Trans. Signal Process.* **2014**, *62*, 1424–1437. [[CrossRef](#)]
27. Boyd, S.; Parikh, N.; Chu, E.; Peleato, B.; Eckstein, J. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Found. Trends Mach. Learn.* **2010**, *3*, 1–122. [[CrossRef](#)]
28. Erseghe, T. A Distributed and Maximum-Likelihood Sensor Network Localization Algorithm Based Upon a Nonconvex Problem Formulation. *IEEE Trans. Signal Inf. Process. Netw.* **2015**, *1*, 247–258. [[CrossRef](#)]
29. Piovesan, N.; Erseghe, T. Cooperative Localization in WSNs: A Hybrid Convex/non-Convex Solution. *IEEE Trans. Signal Inf. Process. Netw.* **2016**, *4*, 162–172. [[CrossRef](#)]
30. Setlur, P.; Smith, G.E.; Ahmad, F.; Amin, M.G. Target Localization with a Single Sensor via Multipath Exploitation. *IEEE Trans. Aerosp. Electron. Syst.* **2012**, *48*, 1996–2014. [[CrossRef](#)]
31. Leitinger, E.; Meissner, P.; Rudisser, C.; Dumphart, G.; Witrissal, K. Evaluation of Position-Related Information in Multipath Components for Indoor Positioning. *IEEE J. Sel. Areas Commun.* **2015**, *33*, 2313–2328. [[CrossRef](#)]
32. Ma, Y.; Zhou, L.; Liu, K.; Wang, J. Iterative Phase Reconstruction and Weighted Localization Algorithm for Indoor RFID-Based Localization in NLOS Environment. *IEEE Sens. J.* **2014**, *14*, 597–611. [[CrossRef](#)]
33. Gao, Z.; Ma, Y.; Liu, K.; Miao, X.; Zhao, Y. An Indoor Multi-Tag Cooperative Localization Algorithm Based on NMDS for RFID. *IEEE Sens. J.* **2017**, *17*, 2120–2128. [[CrossRef](#)]
34. Ghari, P.M.; Shahbazian, R.; Ghorashi, S.A. Wireless Sensor Network Localization in Harsh Environments Using SDP Relaxation. *IEEE Commun. Lett.* **2016**, *20*, 137–140. [[CrossRef](#)]
35. Chen, H.; Wang, G.; Wang, Z.; So, H.C.; Poor, H.V. Non-Line-of-Sight Node Localization Based on Semi-Definite Programming in Wireless Sensor Networks. *IEEE Trans. Wirel. Commun.* **2012**, *11*, 108–116. [[CrossRef](#)]
36. Vaghefi, R.M.; Buehrer, R.M. Cooperative Localization in NLOS Environments Using Semidefinite Programming. *IEEE Commun. Lett.* **2015**, *19*, 1382–1385. [[CrossRef](#)]

37. Ding, C.; Qi, H.D. Convex Euclidean distance embedding for collaborative position localization with NLOS mitigation. *Comput. Optim. Appl.* **2017**, *66*, 187–218. [[CrossRef](#)]
38. Yousefi, S.; Chang, X.W.; Champagne, B. Distributed cooperative localization in wireless sensor networks without NLOS identification. In Proceedings of the 2014 11th Workshop on Positioning, Navigation and Communication (WPNC), Dresden, Germany, 12–13 March 2014; pp. 1–6.
39. Huber, P.J. Robust Estimation of a Location Parameter. *Ann. Math. Stat.* **1964**, *35*, 73–101. [[CrossRef](#)]
40. Turin, G.L.; Clapp, F.D.; Johnston, T.L.; Fine, S.B.; Dan, L. A Statistical Model of Urban Multipath Propagation. *IEEE Trans. Veh. Technol.* **1972**, *21*, 1–9. [[CrossRef](#)]
41. Aulin, T. A Modified Model for the Fading Signal at a Mobile Radio Channel. *IEEE Trans. Veh. Technol.* **1979**, *28*, 182–203. [[CrossRef](#)]
42. Yu, K.; Guo, Y.J. Improved Positioning Algorithms for Nonline-of-Sight Environments. *IEEE Trans. Veh. Technol.* **2008**, *57*, 2342–2353.
43. Boyd, S.; Vandenberghe, L.; Foybusovich, L. Convex Optimization. *IEEE Trans. Autom. Control* **2006**, *51*, 1859–1859.
44. Parikh, N.; Boyd, S. Proximal Algorithms. *Found. Trends Optim.* **2014**, *1*, 127–239. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).