Research Article

Outlier-Resistant Localization for Smart Cities: Low-Rank Approximation-Based Approaches

Feng Wang ⑩,[1,2] **Peng Ma,**[2,3] **Rui Zhu** ⑩,[1] **and Jianxin Guo**[1]

[1]*Xijing University, Xi'an, Shaanxi 710123, China*
[2]*Shaanxi Key Laboratory of Integrated and Intelligent Navigation, Xi'an, Shaanxi 710071, China*
[3]*Xi'an Research Institute of Navigation Technology, Xi'an, Shaanxi 710071, China*

Correspondence should be addressed to Rui Zhu; zhu_r10@163.com

For smart cities, location-based services (LBSs) are indispensable; however, the urban environment is typically a multipath channel and achieving high accuracy localization is challenging, especially in GNSS-denied environments. It is already known that there are two key factors that constrain IoT localization performance, namely, the presence of outliers in the inter-node ranging data and the difficulty to guarantee that ranging is carried out between all nodes, which means that we are dealing with an incomplete Euclidean distance matrix (EDM) contaminated with outliers. In this paper, we propose a robust localization framework, termed low-rank approximation-based localization (LRAbL). LRAbL enables network localization in a stepwise manner using partially observed EDMs that contain coarse noise (outliers). Specifically, the working process of LRAbL can be divided into three stages: the preprocessing of the observed EDM, aiming to eliminate outliers with large values, the use of low-rank approximation means to obtain the complete EDM, and finally the application of non-classical MDS to calculate the coordinates of the nodes in the network. To confirm the applicability of the proposed framework, extensive numerical experiments were conducted, which indicated that LRAbL was still able to achieve satisfactory localization results when the network links were sparse (only about 1/4 of the entries can be observed) and contained a certain percentage of large value outliers. In summary, our work provides a solution worthy of consideration for location-based services in the future Internet of Things.

1. Introduction

The last decade has witnessed a boom in Internet of Things (IoT) technologies, and it can be predicted that location-based services (LBSs) will play an increasingly important role in smart cities [1, 2]. Existing localization techniques, such as Global Navigation Satellite System (GNSS), cellular networks, or Wi-Fi, have their limitations and are either infrastructure-dependent or not suitable for working in urban dense buildings. Therefore, in order to achieve high-accuracy localization of IoT in smart cities, we expect the network nodes to have relative localization capabilities, that is, they can obtain relative coordinates even without relying on additional equipment and using only ranging data.

By using received signal strength indication (RSSI), time of flight (TOF), or time difference of arrival (TDOA), the distance between nodes can be measured and, theoretically, the relative coordinates of nodes can be easily calculated using the Euclidean distance matrix (EDM) constructed from range data. However, in real-world scenarios, the performance of distance-based IoT localization is often affected by two negative factors, i.e., the loss of range data and outliers caused by non-line-of-sight (NLOS) propagation.

Although some localization algorithms, such as the classical MDS and its variants [3–5], have been proposed in the hope of accurately calculating the relative coordinates of nodes, to the best of our knowledge, robust localization methods that work well in complex environments, where range data loss is common and can contain outliers, are still lacking; therefore, the design of robust and less complex relative localization frameworks remains an area of interest.

Focusing on this issue, the existing work can be broadly classified into three categories, namely, outlier processing, matrix completion, and localization based on optimization methods.

Two main methods are used to detect outliers in EDM without considering missing entries, outlier detection using only ranging data, and NLOS link identification based on statistical characteristics of the received signal. Using only EDM, Blouvshtein and Cohen-Or [6] proposed an outlier detection method based on triangle inequalities, which is able to detect errors of large amplitude in the ranging data but requires an artificially determined threshold according to the number of broken triangles. By projecting the observed EDM onto a high dimensional space, Zhu et al. [7] proposed a novel outlier detection method; nevertheless, the performance of the method cannot be guaranteed when the number of outliers is relatively large.

Based on the statistical characteristics of the received signal strength (RSS) time series, Xiao et al. [8] achieved the separation of LOS/NLOS data; experiments show that this method can obtain a detection success rate of 95%. Similarly, by modelling the error data due to NLOS, Momtaz et al. [9] also proposed a fast detection method using feature vectors. Recently, Nguyen et al. [10] introduced machine learning tools into the localization system and used the relevance vector machine (RVM) to identify the NLOS links and suppress the gross errors. Using the data generated by the WLAN, Choi et al. [11] demonstrated that the data collected in a short period of time can be used to train a deep learning model to better learn the non-linear relationship between input and output, which in turn enables the identification of NLOS links. In addition, based on expectation maximization (EM) algorithm of Gaussian mixture model, Fan and Awan [12] introduced unsupervised machine learning method into the UWB system for LOS/NLOS link identification.

Focusing on the outlier detection problem, although some research achievements have been made, it must be noted that NLOS link identification based on statistical methods requires the collection of sufficient training data, and thus the real-time performance of localization is hardly guaranteed; moreover, such methods require more computing resources and storage space.

In practice, as mentioned already, most of the time only some entries of the EDM can be observed; for this reason, researchers have developed various methods to solve this problem. MDS-MAP [13] uses shortest path methods to calculate the approximation of missing entries, while SVD-MDS [14] uses singular value decomposition (SVD) to reconstruct the EDM, both of which are able to recover the missing entries, but the accuracy is not satisfactory. To obtain more accurate EDM, an intuitive idea is to use matrix completion (MC) algorithms, for which a preliminary summary is given in [15], and in addition, we suggest the readers to follow work such as [16]. Indeed, several researchers have noticed the problem presented in this paper, namely, how to obtain the exact coordinates of the nodes when the incomplete EDM contains outliers. Nguyen et al. [17] formulated matrix completion as an unconstrained optimization problem on Riemannian manifolds, solved it

using a modified conjugate gradient algorithm, and also considered the case of the presence of outliers. Similarly, using optimization algorithms such as block coordinate descent (BCD) and alternating direction multiplication (ADMM), Xiao et al. and Guo and Lin [18, 19] hoped to deal with two things simultaneously, namely, outlier filtering and matrix completion, by solving an optimization problem.

Using an optimization algorithm to deal with both outliers and missing entries is mathematically well interpreted; however, it imposes a large computational burden and, to achieve good results, requires repeated attempts to adjust the optimization parameters for different scenarios. To address these issues, we propose a low-rank approximation-based localization (LRAbL) scheme to achieve robust localization using partially observed EDM that contains outliers. Compared to existing schemes, our work is different in the following ways:

(i) We propose a general framework LRAbL, in which the three main modules, outlier detection, low-rank approximation, and coordinate calculation, can be flexibly replaced as required; for example, outlier processing can be done using only EDM or using statistical detection methods.

(ii) Considering the characteristics of NLOS link ranging, we propose an outlier detection method using only observed EDM; unlike existing methods, our approach identifies and filters outliers with large values without artificially setting a decision threshold.

(iii) Several mainstream low-rank approximation algorithms are applied to LRAbL, their performance is compared in depth, and the corresponding theoretical analysis is carried out. In addition, we discuss the potential of RPCA for robust localization, which has rarely been addressed in the literature.

2. Problem Formulation

Considering a sensor network with N nodes deployed in d-dimensional space, the coordinates of all nodes are represented as $\mathbf{X} \in R^{N \times d}$. Define $d_{ij} = \| X_i - X_j \|_2$ as the Euclidean distance between node i and node j; obviously there is $d_{ij} = d_{ji}$; a symmetric square matrix, called Euclidean distance matrix (EDM), can be constructed, and subsequently, the coordinates of the nodes can be easily calculated using methods such as MDS.

Figure 1 illustrates the mechanism of generating the Euclidean distance matrix (EDM) in a complex environment, and it can be seen that because there is no direct ranging link between nodes n_1 and n_3 (because of too far distance or severe signal fading), the entry in the EDM at the corresponding position is missing (denoted as ?); moreover, due to the presence of obstacle between nodes n_3 and n_4, the measurement of the distance between them is actually implemented on the NLOS link; therefore, the measured distance may be much larger than the true distance (denoted as Δ); in other words, an outlier is introduced in the EDM.

$$D = \begin{bmatrix} 0 & d_{12}^2 & ? & d_{14}^2 & d_{15}^2 \\ d_{21}^2 & 0 & d_{23}^2 & d_{24}^2 & ? \\ ? & d_{32}^2 & 0 & \Delta & d_{35}^2 \\ d_{41}^2 & d_{42}^2 & \Delta & 0 & d_{45}^2 \\ d_{51}^2 & ? & d_{53}^2 & d_{54}^2 & 0 \end{bmatrix}$$
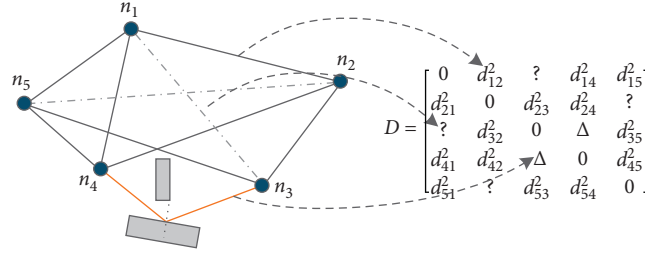
FIGURE 1: Generation of the observed EDM, where ? denotes missing entries, while Δ denotes entries that contain outliers.

In order to obtain the coordinates of the nodes, we need to have a complete and accurate EDM; however, even if the ranging noise is weak, reconstructing the ideal EDM L from D is still challenging and requires simultaneously recovering the missing entries and filtering out the outliers. Mathematically, the problem to be solved can be expressed as

$$\min_{L,S} \frac{1}{2} \| P_\Omega (L + S - D) \|_F^2 + \lambda \operatorname{Rank}(L) + \tau \|S\|_1, \qquad (1)$$

where $P_\Omega(D)$ denotes the sampling operator, defined as

$$\left(P_\Omega(D)\right)_{ij} = \begin{cases} D_{ij}, & (i,j) \in \Omega, \\ 0, & \text{other wise}, \end{cases} \qquad (2)$$

where L and S with the same dimensions as $D \in R^{N \times N}$, respectively, representing the ideal EDM and sparse outlier matrix caused by NLOS propagation, respectively. λ and τ are regularization parameters, and $\|\cdot\|_F^2$ is the Frobenius norm of the matrix; furthermore, $\operatorname{Rank}(L)$ denotes the rank of L, and S_1 denotes the l_1 norm of S. We can give an intuitive interpretation of (1) by first introducing Theorem 1.

Theorem 1 (see Theorem 1 in [15]). *The rank of an EDM* $L \in R^{N \times N}$ *corresponding to points* $X \in R^{N \times d}$ *is at most* $d + 2$, *i.e.,* $\operatorname{Rank}(L) \le d + 2$.

In order to suppress ranging noise, the first term of (1) requires that the reconstructed matrix is as close as possible to the entries of the EDM in the observable subset Ω. The second term comes from Theorem 1, and the third term is based on the reasonable assumption that the matrix of outliers due to NLOS propagation is sparse. The solutions to our problem of concern are summarized in Figure 2.

① in Figure 2 is, in fact, the MDS-MAP, and ③ is an improvement of ①, exploiting the low-rank nature of the EDM, while ② tries to go directly to solve problem (1). The shaded areas ④ and ⑤ represent our work, and it is important to highlight that the outlier detection we use differs from the approach of ⑥, where (P) and (C) denote the partial and complete EDM, respectively. Table 1 summarizes the characteristics of the six schemes, and it can be seen that our work contains three robust processing modules that, in theory, have the potential to perform better.

3. Proposed Framework

This section provides a detailed description of LRAbL, as shown in Algorithm 1. LRAbL first performs outlier detection

and filtering, followed by low-rank matrix reconstruction and, finally, calculates the relative coordinates of the nodes.

The operator $\odot$ in Algorithm 1 denotes the Hadamard product. It is important to emphasize the second step of Algorithm 1, the low-rank approximation (LRA) of the matrix, and both MC and RPCA use the mask matrix $\mathbf{W}_{OD} \in \{0, 1\}^{N \times N}$ as input parameters. In Figure 2, by intentionally introducing artificial outliers, ④ transforms the objective function into a typical RPCA problem in order to decompose it to obtain a low-rank EDM, whereas ⑤ replaces all outliers with unknown entries, thus transforming the objective function into a typical matrix completion problem. In addition, NMDS () in step 3 denotes the nonclassical MDS algorithm.

3.1. Outlier Detection.
Ideally, the Euclidean distances between any three nodes in a sensor network satisfy the triangle inequality, but if there are several NLOS links in the network, outliers with larger amplitudes will be introduced into the ranging data, and then the triangle inequality will be broken. As shown in Figure 3, the measured distance between nodes A and C, i.e., $\widehat{d_{AC}} = d_{AD} + d_{DC}$, is greater than the true distance d_{AC}, and may even be likely greater than $d_{AB} + d_{BC}$; therefore, the triangle inequality no longer holds.

Recently, Blouvshtein and Cohen-Or [6] proposed an outlier detection method based on the histogram of broken triangles, and the problem we face is slightly different from [6] in that, first, the network is not fully connected, and second, we do not consider inlier edges because NLOS propagation can only lead to outlier edges. In Algorithm 2, we designed a different approach that does not require generating histograms of broken triangles or artificially determining decision threshold.

In Algorithm 2, first, the incomplete graph $\mathscr{G} = (V, E)$ with the vertex set $V = 1, 2, \ldots, N$ and the edge set $E = \left\{(i, j) \mid d_{ij} > 0, i, j \in V\right\}$ is generated based on the observed EDM, and the one-hop neighbour set U_i of the $i - th$ node is defined, and each node in U_i has a one-hop ranging link with node i. For efficiency purposes, U_i contains only nodes with index IDs larger than i. The nodes in the intersection of U_i and U_j, defined as Set_{ij}, can form triangles with node (i, j). After finding all the triangles contained in the incomplete graph $\mathscr{G}$, in the last step, we use the function $\operatorname{Outlier Det}(\cdot)$ to determine whether a triangle is broken or not; specifically, the three edges of the triangle are defined as d_1, d_2, and d_3; without loss of generality, assuming that $d_3 > d_2 > d_1$, the result of the decision can be characterized using b:
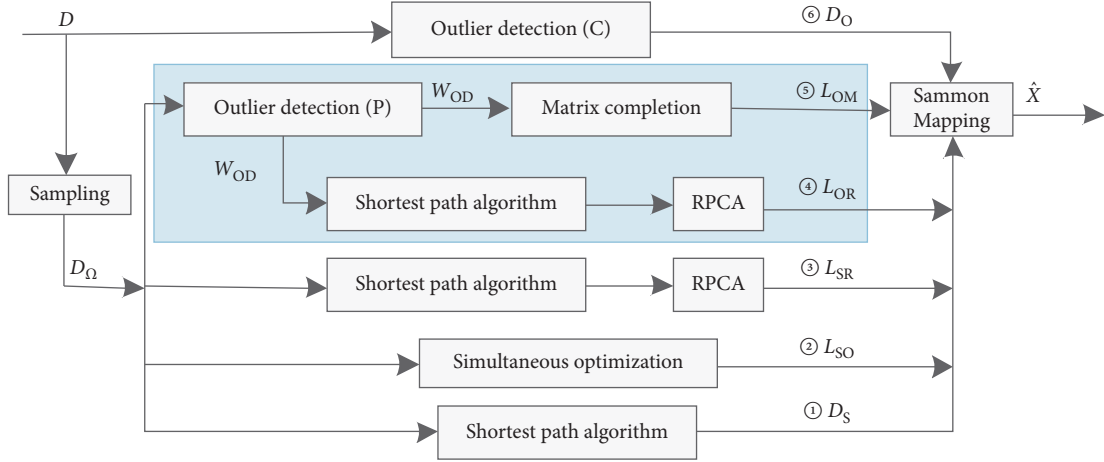
FIGURE 2: Existing robust localization schemes, where the shaded part is the work of this paper.

TABLE 1: Comparison of robust localization solutions.

Solutions	Outlier filtering	Matrix completion	Low-rank constraint
①	×	√	×
②	×	√	√
③	×	√	√
④	√	√	√
⑤	√	√	√
⑥	√	×	×

Input: observed EDM $D_\Omega = P_\Omega(D)$.
Output: the estimated coordinates $\widehat{X}$.
Step1: use Algorithm 2 to determine the index set of outliers and generate the masking matrix $\mathbf{W}_{OD}$.
Step2: implementing low-rank matrix approximation.
$\widehat{L} = \text{LRA}(\mathbf{D}_\Omega \odot \mathbf{W}_{OD})$.
Step3: using non-classical MDS to calculate the relative coordinates of the nodes.
$\widehat{X} = \text{NMDS}(\widehat{L})$.

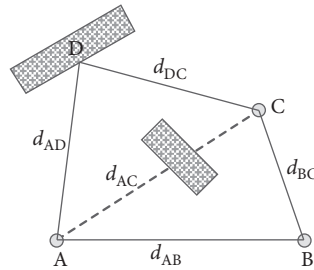ALGORITHM 1: Low-rank approximation-based localization.

FIGURE 3: An illustration of the scenario where the triangular inequality is broken; in this scenario, $\widehat{d_{AC}} > d_{AB} + d_{BC}$, a result that defies common sense and implies that outliers are introduced, at least on the edge AC.

$$b = \begin{cases} 1, & d_3 > d_2 + d_1, \\ 0, & \text{otherwise.} \end{cases} \qquad (3)$$

Moreover, OutlierIndex is an index that represents the position of the corresponding entry in the matrix for the detected outliers.

3.2. Matrix Completion. Recently, a considerable number of MC algorithms have been proposed, and they can be roughly classified into two categories. When the rank $r = \text{Rank}(L)$ of the matrix is unknown, the algorithms that can be used for matrix completion include nuclear norm minimization (NNM) via convex optimization [20], singular value

```
Input: observed EDM D_Ω.
Output: masking matrix W_OD.
Initialize: W_OD = ones(N, N).
Based on graph 𝒢 = (V, E), generate the set of one-hop neighbours for all nodes: U = U_{i=1}^N U_i.
for  i = 1; i < N; i + +  do
   for  all j ∈ U_i  do
               Set_{ij} = U_i ∩ U_j.
      for  k = 1: length(Set_{ij})  do
           Triset(k) = (node(i), node(j), node(Set_{ij}(k))).
           [b, Outlier Index] = Outlier Det(Triset(k)).
           if  (b == 1)  then
               W_OD(Outlier Index) = 0.
           else
                  W_OD = W_OD.
           end if
      end for
   end for
end for
```

ALGORITHM 2: Outlier detection.

thresholding (SVT) [21] and iteratively reweighted least squares (IRLS) [22]. When the rank is known or can be estimated, r can be utilized as a priori information, and in general, as verified by the compressed sensing problem, the introduction of priori information can improve the performance of matrix completion algorithms; for example, the performance of truncated NNM (TNNR) [23] is significantly improved over NNM, and similar examples can be found in [24, 25].

The matrix completion problem can be formulated mathematically as

$$\min_L \| L \|_* \text{ s.t. } P_\Omega(L) = P_\Omega(D), \quad (4)$$

where $\| L \|_* = \sum_{i=1}^N \sigma_i(L)$ is the nuclear norm of $\mathbf{L}$ and $\sigma_i(L)$ is the $i - th$ largest singular value of L. Since the nuclear norm minimization is equivalent to semidefinite programming [16], which is computationally burdensome and, in addition, when the condition number of the matrix is large, it is not easy to obtain the optimal result. Using a nonconvex regularization (e.g. Schatten-q norm) can achieve better results than the nuclear norm, but there is a risk of convergence to a local minimum [26].

In this paper, we apply various MC algorithms to LRAbL, among which MatrixIRLS [27], a recently proposed algorithm, has attracted our special attention. MatrixIRLS constructs a surrogate function $f_\varepsilon(\sigma)$:

$$f_\varepsilon(\sigma) = \begin{cases} \log|\sigma|, & \text{if } \sigma \geq \varepsilon, \\ \\ \log(\varepsilon) + \left(\frac{1}{2}\right)\left(\left(\frac{\sigma^2}{\varepsilon^2}\right) - 1\right), & \text{if } \sigma < \varepsilon. \end{cases} \quad (5)$$

To approximate the rank of the matrix, the objective function $F_\varepsilon: R^{N \times N} \longrightarrow R$ can be defined as

$$F_\varepsilon(\mathbf{L}) := \sum_{i=1}^N f_\varepsilon(\sigma_i(\mathbf{L})), \quad (6)$$

where $F_\varepsilon(\mathbf{L})$ is apparently a continuously differentiable function with ε_k^{-2}-Lipschitz gradient and $\nabla F_{\varepsilon_k}(\mathbf{L})$ can be calculated as (using the threshold parameter ϵ_k)

$$\nabla F_{\varepsilon_k}(\mathbf{L}) = U \text{diag}\left(\frac{\sigma_i(\mathbf{L})}{\max(\sigma_i(\mathbf{L}), \varepsilon_k)^2}\right)_{i=1}^N \mathbf{V}^T, \quad (7)$$

where $\mathbf{U}$ and $\mathbf{V}$ can be obtained by the SVD of $\mathbf{L}$.

Inspired by the modified Newton's method, a relatively easy to solve quadratic convex function $Q_{\varepsilon_k}(\mathbf{L}|\mathbf{L}^{(k)})$ is defined which allows us to compute $\mathbf{L}^{(k+1)}$ from the known $\mathbf{L}^{(k)}$ using an iterative style. $Q_{\varepsilon_k}(\mathbf{L} | \mathbf{L}^{(k)})$ can be defined as

$$L^{(k+1)} = \text{argmin}_L Q_{\varepsilon_k}\left(L|L^{(k)}\right)$$
$$= F_{\varepsilon_k}\left(L^{(k)}\right) + \left\langle \nabla F_{\varepsilon_k}\left(L^{(k)}\right), L - L^{(k)} \right\rangle \quad (8)$$
$$+ \frac{1}{2}\left\langle L - L^{(k)}, W^{(k)}\left(L - L^{(k)}\right)\right\rangle.$$

In fact, $Q_{\varepsilon_k}(\mathbf{L}|\mathbf{L}^{(k)})$ can be viewed as replacing the Hessian by an operator $W^{(k)}: R^{N \times N} \longrightarrow R^{N \times N}$, which is defined as

$$W^{(k)}(\mathbf{L}) = \mathbf{U}_{(k)}\left[\mathbf{H}_{(k)} \odot \left(\mathbf{U}_{(k)}^T \mathbf{L} \mathbf{V}_{(k)}\right)\right]\mathbf{V}_{(k)}^T, \quad (9)$$

where

$$\left(\mathbf{H}_{(k)}\right)_{ij} := \left(\max\left(\sigma_i^{(k)}, \varepsilon_k\right)\max\left(\sigma_j^{(k)}, \varepsilon_k\right)\right)^{-1}, \quad (10)$$

and $\mathbf{U}_{(k)}\mathbf{S}_{(k)}\mathbf{V}_{(k)}^T = \text{SVD}(\mathbf{L}^{(k)})$ and $\sigma_i^{(k)}$ is the $i - th$ singular value of $\mathbf{L}^{(k)}$. It should be noted that the original purpose of

designing $\mathbf{H}_k$ is to ensure that $Q_{\epsilon_k}(\mathbf{L}|\mathbf{L}^{(k)}) \geq F_{\epsilon_k}(\mathbf{L})$ and to maintain the convexity of the objective function; therefore, the global minima of $F_{\epsilon_k}(\mathbf{L})$ can be approximated by solving for $Q_{\epsilon_k}(\mathbf{L}|\mathbf{L}^{(k)})$.

The working process of MatrixIRLS can be summarized as shown in Algorithm 3, where Ω is a new version of the observable subset. In each update of λ_{k+1}, its value should be gradually decreasing, and depending on the specific problem, different update strategies can be chosen, e.g., $\lambda_{k+1} = \lambda_1 e^{-k\beta}$, where $\beta = \ln(\lambda_1/\lambda_k)/(K-1)$. The objective function of Algorithm 3 consists of two parts, namely, the low-rank constraint and the data fidelity term, and we have

$$R(\mathbf{L}) := \langle \mathbf{L}, W^{(k-1)}(\mathbf{L}) \rangle \tag{11}$$

It can be seen that $R(\mathbf{L})$ comes precisely from the minimization of $Q_{\epsilon_k}(\mathbf{L}|\mathbf{L}^{(\mathbf{k})})$. If the presence of noise is not considered, the final problem we need to solve can be expressed as

$$L^{(k)} = \underset{L:\ \mathrm{P}_\Omega(L)=L_\Omega}{\arg\min}\ \langle L, W^{(k-1)}(L) \rangle, \tag{12}$$

where $\mathbf{L}_\Omega$ is the matrix obtained by sampling from a subset Ω of $\mathbf{L}$.

We obtained the objective function of Algorithm 3 based on the modified Newton's method (which brings the benefit of saddle point escape), and a question that naturally arises is whether this approximated objective function can achieve the low-rank constraint? Here we give Theorem 2, which aims to establish an intuitive connection between the objective function of Algorithm 3 and the low-rank constraint commonly used in the literature.

Theorem 2. *Suppose $R(\mathbf{L})$, $H(\mathbf{L})$ are defined as in (11) and (10); when Algorithm 3 reaches the convergence state, the equation*

$$\begin{aligned} \widehat{L} &= \underset{L:\ \mathrm{P}_\Omega(L)=L_\Omega}{\arg\min}\ R(L) \\ &= \underset{L:\ \mathrm{P}_\Omega(L)=L_\Omega}{\arg\min}\ \mathrm{Rank}(L) \end{aligned} \tag{13}$$

holds. This means that $R(\mathbf{L})$ is essentially a low-rank constraint.

Proof. For completeness of presentation, rewrite $W^{(k)}(\mathbf{L})$ as follows:

$$W^{(k)}(\mathbf{L}) = \mathbf{U}_{(k)}\left[\mathbf{H}_{(k)} \odot \left(\mathbf{U}_{(k)}^T \mathbf{L} \mathbf{V}_{(k)}\right)\right]\mathbf{V}_{(k)}^T. \tag{14}$$

Assume that Algorithm 3 reaches convergence when $k = K$, at which point the operator $W^{(K)}$ can be expressed as

$$W^{(K)}\left(\mathbf{L}^{(K)}\right) = \mathbf{U}_{(K)}\left[\mathbf{H}_{(K)} \odot \left(\mathbf{U}_{(K)}^T \mathbf{L}^{(K)} \mathbf{V}_{(K)}\right)\right]\mathbf{V}_{(K)}^T. \tag{15}$$

By definition, $\mathbf{H}_{(K)}$ can be expressed as

$$\mathbf{H}_{(K)} = \begin{bmatrix} \mathbf{H}_{11} & \mathbf{H}_{12} \\ \mathbf{H}_{21} & \mathbf{H}_{22} \end{bmatrix}, \tag{16}$$

where $\mathbf{H}_{11} \in R^{r \times r}$ with $(\mathbf{H}_{11})_{ij} = 1/\sigma_i\sigma_j$, $\mathbf{H}_{12} \in R^{r \times (N-r)}$ with $(\mathbf{H}_{12})_{ij} = 1/\sigma_{i+r}\varepsilon_K$, $\mathbf{H}_{21} \in R^{(N-r) \times r}$ is similar to $\mathbf{H}_{12}$, with $(\mathbf{H}_{21})_{ij} = 1/\varepsilon_K\sigma_{j+r}$, and $\mathbf{H}_{22} \in R^{(N-r) \times (N-r)}$ is slightly different in that all its entries are ε_K^{-2}.

Considering the low-rank property of $\mathbf{L}$, it is not difficult to obtain

$$\mathbf{S}^{(K)} = \mathbf{U}_{(K)}^T \mathbf{L}^{(K)} \mathbf{V}_{(K)}, \tag{17}$$

where $\mathbf{S}^{(K)}$ is a diagonal matrix composed of singular values of $\mathbf{L}^{(k)}$ whose last $N - r$ entries take the value 0. It can be deduced that $\Sigma = \mathbf{H}_{(K)} \odot (\mathbf{U}_{(K)}^T \mathbf{L}^{(K)} \mathbf{V}_{(K)})$ is also a diagonal matrix and can be expressed as

$$\Sigma = \mathrm{diag}\left(\sigma_1^{-1}, \ldots, \sigma_r^{-1}, 0, \ldots, 0\right). \tag{18}$$

This means that Σ is essentially the Moore–Penrose pseudo-inverse of $\mathbf{S}^{(K)}$; therefore, we get

$$W^{(K)}\left(\mathbf{L}^{(K)}\right) = \mathbf{U}_{(K)}\Sigma\mathbf{V}_{(K)}^T = \left(\left(\mathbf{L}^{(K)}\right)^\dagger\right)^T, \tag{19}$$

where $(\mathbf{L}^{(K)})^\dagger = \mathbf{V}_{(K)}\Sigma\mathbf{U}_{(K)}^T$ is the Moore–Penrose pseudo-inverse of $\mathbf{L}^{(\mathbf{K})}$.

For a matrix $\mathbf{A}$ of non-full rank, it is easy to prove that $\mathbf{P} := (\mathbf{A}^\dagger\mathbf{A})^T$ is an idempotent matrix; furthermore, we can prove that

$$\mathrm{Trace}(\mathbf{P}) = \mathrm{Rank}(\mathbf{P}) = \mathrm{Rank}(\mathbf{A}). \tag{20}$$

Returning to our problem, when the algorithm converges, it is not hard to see that

$$\begin{aligned} R\left(\mathbf{L}^{(K)}\right) &= \left\langle \mathbf{L}^{(K)}, W^{(K)}\left(\mathbf{L}^{(K)}\right) \right\rangle, \\ &= \mathrm{Trace}\left(\left(\mathbf{L}^{(K)}\right)^T W^{(K)}\left(\mathbf{L}^{(K)}\right)\right), \\ &= \mathrm{Trace}\left(\left(\left(\mathbf{L}^{(K)}\right)^\dagger \mathbf{L}^{(K)}\right)^T\right), \\ &= \mathrm{Rank}\left(\mathbf{L}^{(K)}\right). \end{aligned} \tag{21}$$

The proof is accomplished at this point.

Theorem 2 shows that by continuously updating the weighting operator $\mathbf{W}^{(K)}$, MatrixIRLS eventually achieves the minimization of the matrix rank, thereby recovering the missing entries in the EDM. □

3.3. Robust Principal Component Analysis and Non-Classical MDS.
RPCA has achieved remarkable success in the field of image processing, such as foreground extraction and video denoising. In this section, we use RPCA to implement the second step of Algorithm 1. Mathematically, RPCA can be expressed in a canonical form:

$$\min_{L,S} \frac{1}{2}\|L + S - D\|_F^2 + \lambda\|L\|_* + \tau\|S\|_1. \tag{22}$$

By solving (22), it is possible to decompose the observation matrix $\mathbf{D}$ into three components:

$$\mathbf{D} = \mathbf{L} + \mathbf{S} + \mathbf{N}, \tag{23}$$

Input: observed EDM $P_\Omega(D) = D_\Omega \odot W_{O\,D}$.
Output: restructured EDM $\widehat{L}$.
Initialize: $k = 1$, $K = 100$, $\lambda_1 = 0.1$, $\varepsilon_1 = 10$, $r = 4$.
for $K = 1\,to\,K$ **do**
    Solve optimization problem:
        $L^{(k)} = \arg\min_L R(L) + \lambda_k/2 \parallel P_\Omega(L) - P_\Omega(D)\parallel_F^2$.
    Update smoothing parameter:
        $\varepsilon_{k+1} = \min(\varepsilon_{(k)}, \sigma_{r+1}(\mathbf{L}^{(k)}))$.
    Generate the matrices needed for the weighting operator $W^{(k)}(\mathbf{L})$.
        $\mathbf{U}_{(k)}, \mathbf{H}_{(k)}, \mathbf{V}_{(k)}$.
    Update λ_{k+1}.
end for
        $\widehat{L} = \mathbf{L}^{(K)}$.

ALGORITHM 3: MatrixIRLS.

where $\mathbf{N}$ is the Gaussian additive noise generated during ranging. Pay attention to ④ in Figure 2; although both use the shortest distance algorithm to reconstruct the matrix, LRAbL-RPCA is superior to MDS-MAP in that it not only filters outliers but also exploits the low-rank property of EDM.

It should also be noted that in the final step of Algorithm 1, we use non-classical MDS, namely, Sammon mapping (SM), to calculate the coordinates of the nodes. The objective function to be solved is

$$\text{Stress}(\widehat{X}) := \sum_{i \neq j} \frac{\left(D_{ij} - \left\|X_i - X_j\right\|\right)^2}{D_{ij}}, \tag{24}$$

and the SM was chosen because Blouvshtein and Cohen-Or [6] have demonstrated that it has the ability to suppress outliers.

4. Numerical Evaluation

In this section, we use numerical simulations to verify the effectiveness of the proposed localization framework. In all simulations, we consider a sensor network containing $N = 50$ nodes randomly deployed in a $20\,\text{m} \times 20\,\text{m} = 400\,\text{m}^2$ square area. Given that the performance of SM and the sensitivity of classical MDS to outliers have been demonstrated in [6], we do not show it again. The simulation results show that the value of λ has little impact on the performance if it is relatively small, so we fix $\lambda = 0.1$.

4.1. Incomplete Matrix-Based Localization. To confirm the effectiveness of the proposed strategy, we compared the performance of MDS-MAP with that of LRAbL (using MatrixIRLS to complete the matrix). The EDM used is an incomplete EDM containing 413 observable entries, and the simulation results are presented in the Shepard diagram in Figure 4, where the X-axis represents the real distance between the nodes and the Y-axis represents the estimated value, and ideally the two should agree, i.e., all the scatter points lie on the ideal curve.

Figure 4 shows that LRAbL-MatrixIRLS displays a noticeable superiority over MDS-MAP in that it is able to accurately estimate the distance between nodes, whereas MDS-MAP has poorer performance due to the limited accuracy of the estimated EDM used.

4.2. The Necessity of Outlier Filtering. In the previous section, we did not consider the presence of outliers in the incomplete EDM; in this part, we will demonstrate the necessity of outlier filtering, the first step of LRAbL, through simulation. We set up two scenarios for simulation, and the parameters involved are listed in Table 2, where S_R denotes the sampling rate parameter, based on which $m = \lfloor rS_R(2N - r)\rfloor$ observable entries can be generated; moreover, $\lfloor \cdot \rfloor$ is actually the floor function, i.e., it will output the largest integer that is not greater than the input.

We use MatrixIRLS for matrix completion and generate outliers by sampling Gaussian white noise, more specifically, Gaussian white noise with variance Var, of which the signals with absolute values greater than T_O are selected as outliers to be added to the incomplete EDM. Using each set of parameters, we run 500 Monte Carlo simulations to evaluate the localization performance, and the result of the $i - \text{th}$ simulation is defined as

$$s_i = \begin{cases} 1, & \|\mathbf{X} - \widehat{\mathbf{X}}\|_F \leq T_S, \\ 0, & \text{otherwise}, \end{cases} \tag{25}$$

where $\mathbf{X}$ and $\widehat{\mathbf{X}}$ are the real and estimated coordinates of all nodes, respectively, and the localization success rate is expressed as $S_R = \sum_{i=1}^{500} s_i/500$.

In Figure 5, we consider adding outliers of small amplitude, setting $\text{Var} = [3, 6, 9, 12]$ and $T_O = 13$ and seeking a high-accuracy localization performance, i.e., $T_S = 3$. We compare two strategies, that is, low-rank approximation with and without outlier filtering, and it can be seen that when Var is small, the advantage of outlier filtering is not very obvious because the number of outliers added is small, but as Var becomes larger and the number of outliers added increases, the gain from filtering becomes very obvious.
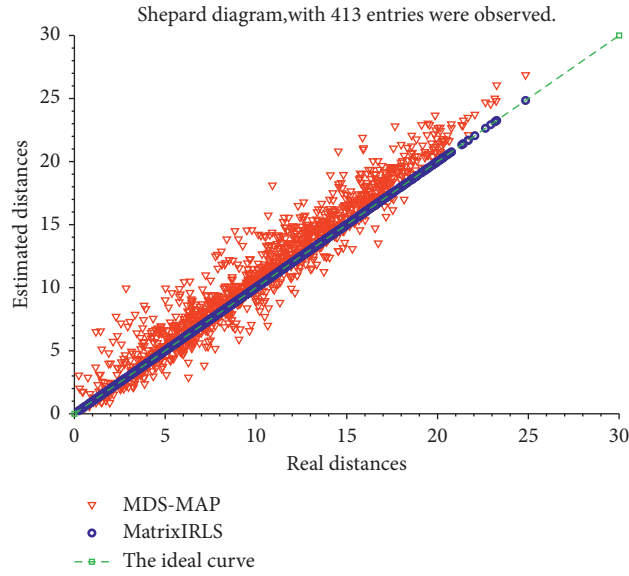
Figure 4: Shepard diagram of localization results using MDS-MAP and LRAbL-MatrixIRLS.

Table 2: Description of the parameters used in the numerical simulation.

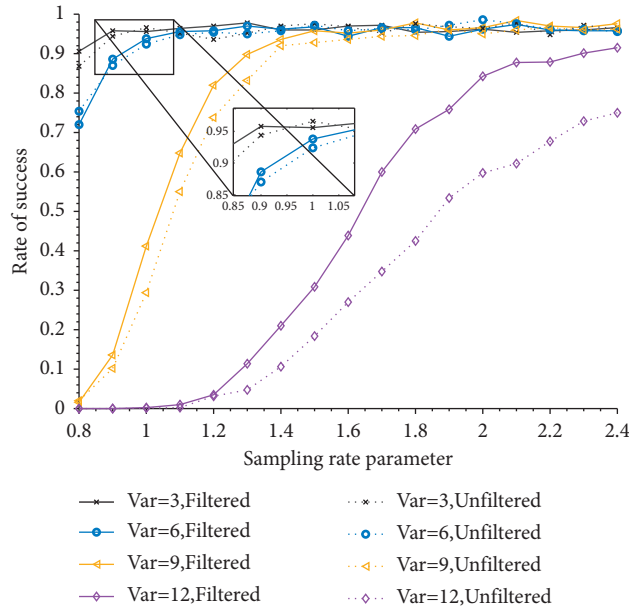| Parameters	Description
S_R	Sampling rate parameter
Var	Variance of outlier
T_O	Threshold for outlier generation
T_S	Threshold for successful localization
MSE	Mean square error of matrix completion

Figure 5: Localization success ratio versus sampling parameters for small-amplitude outliers ($T_O = 13$) and high localization accuracy constraint ($T_S = 3$). The dashed lines represent the localization performance of MatrixIRLS without filtering outliers and the solid lines represent the proposed framework.
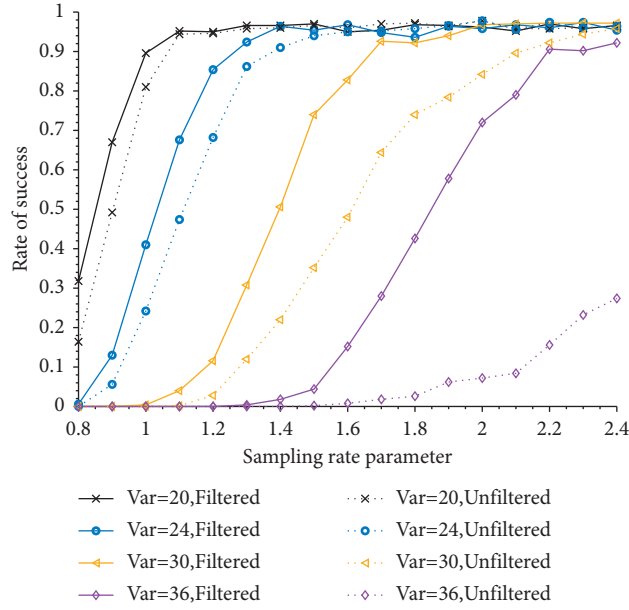
FIGURE 6: Localization success ratio versus sampling parameters for large magnitude outliers ($T_O = 36$) and low localization accuracy constraint ($T_S = 6$). The dashed lines represent the localization performance of MatrixIRLS without filtering outliers and the solid lines represent the proposed framework.

TABLE 3: Matrix completion methods used in the simulation.

Algorithms	Description
MatrixIRLS [27]	An iterative weighted least squares (IRLS) algorithm, which can be interpreted as smoothing Newton's method applied on a non-convex rank surrogate cost function, has the ability of escaping saddle points.
MDS-MAP [13]	The shortest path algorithm, i.e., Dijkstra, is used to estimate the missing ranging data.
AD [15]	An alternating coordinate descent method, where a single coordinate of a particular point is updated each time and the optimal solution is obtained iteratively.
Lmafit [28]	An efficient non-linear successive relaxation strategy is used, where only one linear least squares problem needs to be solved per iteration instead of singular value decomposition (SVD).
TNNR [23]	Using truncated nuclear norm (TNN) regularization, where TNN is defined as the nuclear norm subtracted by the sum of the largest few singular values.
LRM-CG [17]	The low-rank matrix completion is transformed into an unconstrained minimization problem in Riemannian manifolds. The definition of differentiability is also given, and the modified conjugate gradient algorithm is used to solve the problem.

The simulation setup in Figure 6 is similar to Figure 5, except that it uses a larger Var, i.e., Var = [20, 24, 30, 36], $T_O = 36$, and the localization accuracy is suitably reduced by setting $T_S = 6$, which is also acceptable in practice. As expected, LRAbL using outlier filtering performs better than direct MC without filtering.

4.3. Comparison of the Performance of Different Matrix Completion Algorithms. Having confirmed the effectiveness of the proposed framework, in this section, we compare the performance of different matrix completion algorithms, and the algorithms involved are shown in Table 3.

We considered two scenarios, Var = 12 and Var = 9, and the performances of the six algorithms are shown in Figure 7. It can be seen that MatrixIRLS shows a significant advantage in both scenarios, especially when the network links are very sparse. For example, when $S_R = 0.8$ and Var = 9, there are only almost 300 entries that can be

observed in the network. MatrixIRLS still maintains an 80% success rate of localization, which is unachievable by other algorithms.

To further analyze the performance of each algorithm, we defined the mean square error (MSE) as

$$\text{MSE} = \frac{1}{N}\|\widehat{L} - \mathbf{L}\|_F. \tag{26}$$

The MSE is then used to evaluate the accuracy of the matrix completion algorithms, and the simulation results are shown in Figure 8. In this simulation, we added the alternating rank (AR) [15] algorithm for comparison, and the reason why AR was not used in the previous comparison of localization performance is that the matrix reconstructed by the AR algorithm is sometimes not an EDM and therefore cannot be used directly to calculate the coordinates of the nodes. It should also be noted that for MDS-MAP, the MSE is infinite when $S_R = 0.8$ and cannot be shown in the figure.
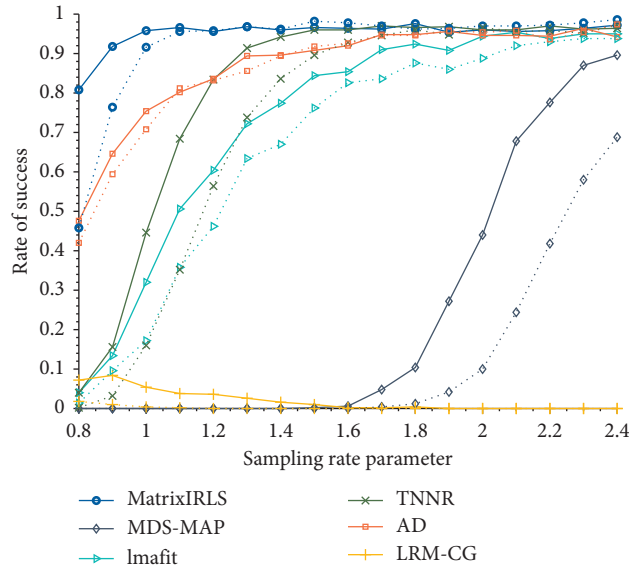
FIGURE 7: The success rate of localization versus the sampling rate parameter, and we show two scenarios, i.e., Var = 12 and Var = 9, which are represented by dashed and solid lines, respectively; in addition, $T_O = 13$ and $T_S = 6$.
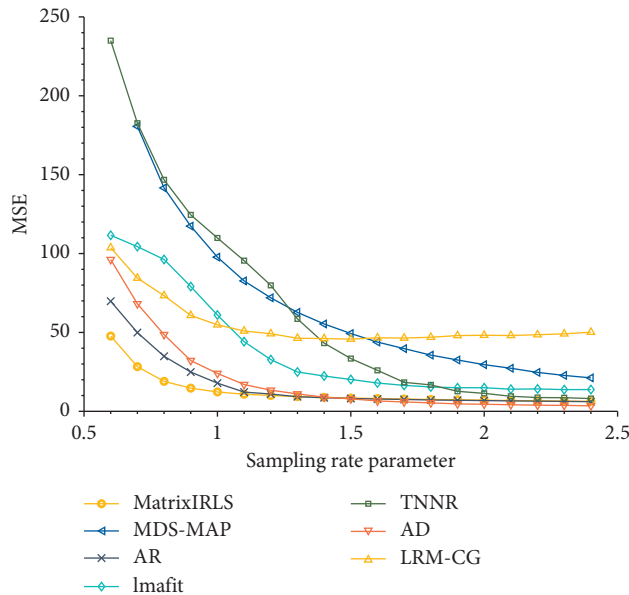

FIGURE 8: MSE versus sampling parameters (Var = 12, $T_O = 13$).

The simulation results in Figure 8 confirm the conclusions of Figure 7. In all scenarios, MatrixIRLS has the highest accuracy for matrix completion, and thus it achieves the best localization results.

4.4. Robust Localization Based on RPCA. In this part, we investigate LRAbL-RPCA using numerical experiments, i.e., LRAbL using robust principal component analysis (RPCA) as a means of low-rank approximation (LRA). Figure 9 compares the localization performance using three strategies, i.e., the classical MDS-MAP and RPCA with (LRAbL-RPCA) and without (shortest path-RPCA) outlier filtering,

and here we used the classical singular value thresholding (SVT) algorithm [21].

As shown in Figure 9, the classical MDS-MAP has the worst performance and the LRAbL-RPCA shows the best performance, which is not difficult to understand because the LRAbL-RPCA first filters out the outliers with large amplitudes and then subsequently completes EDM using Dijkstra and thus suffers the least interference. Nevertheless, it should be noted that the performance of LRAbL-RPCA is much inferior to that of LRAbL-MC when the observation matrix is very sparse, and one possible reason is that we use the rank information of EDM in LRAbL-MC, while LRAbL-RPCA does not make use of this a priori information.
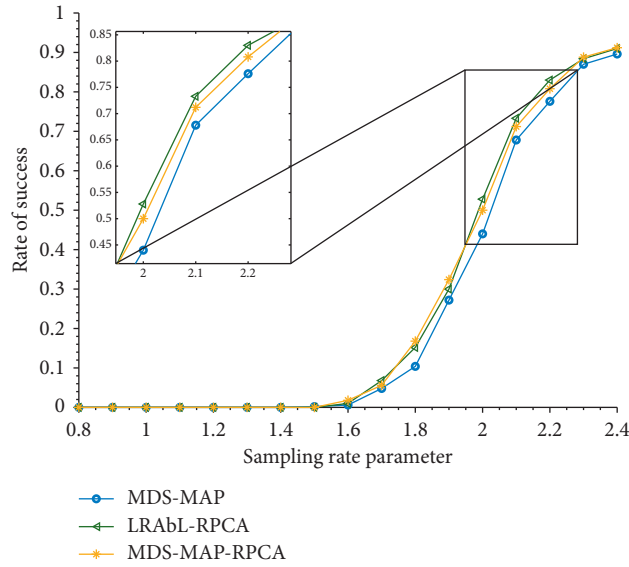
FIGURE 9: Localization success ratio versus sampling parameters for small magnitude outliers (Var = 9, T_O = 13) and low localization accuracy constraint (T_S = 6).
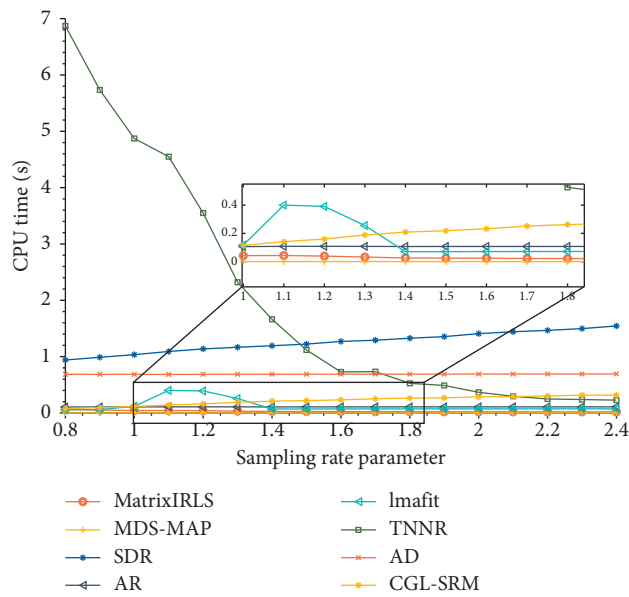

FIGURE 10: The running time of different MC algorithms versus the sampling rate parameter (Var = 12, T_O = 13).

4.5. Computational Complexity.

In practical application scenarios, in addition to performance, we need to balance the complexity of the algorithms, especially in IoT systems where nodes have limited battery capacity and computing power, and we should aim to use a light computational burden to achieve high performance.

We compared the runtimes of several mainstream MC algorithms, some of which we used in our localization performance analysis and some of which we did not use because they could not consistently generate EDM; all algorithms follow the parameter settings suggested by the authors, and the simulation results are shown in Figure 10. All the experiments are conducted on a PC running Windows 10 64 bit operating system with Intel Core i5-9500 @

3.0 GHz CPU and 8.0 GB RAM. Because some algorithms, such as AD and SDR [29], do not have stable operation times, 500 Monte Carlo experiments are performed to calculate the average running time for each sampling rate parameter.

It can be seen that despite the different convergence conditions set by the algorithms, MatrixIRLS shows a clear advantage, with only MDS-MAP (uses the shortest path algorithm) having a slight computational advantage over it among all the algorithms, but the performance gap between the two is very significant. The main reason, as pointed out in Theorem 3.1 of [27], is that MatrixIRLS only needs to operate on $N \times r$ matrices, while its competitors need to store and update $N \times N$ matrices.

5. Conclusion and Discussion

To meet the needs of future smart cities, this paper presents a robust localization framework LRAbL, with low-rank matrix approximation as the main technique. Specifically, the proposed LRAbL consists of three steps: (1) outliers in the incomplete EDM are detected and filtered out to obtain a new observed EDM; (2) the complete EDM is obtained using a low-rank matrix approximation algorithm; and (3) the coordinates of the nodes are calculated using non-classical MDS. Extensive numerical simulations confirm that the proposed framework is effective in improving localization performance, especially when the IoT network works in a very complex environment; when the ranging links are sparse and contain a large number of outliers, LRAbL-MatrixIRLS can achieve localization with sufficient accuracy with a high probability.

In this paper, we use a centralized approach to detect outliers, and in the follow-up work, we will design a distributed outlier detection scheme in order to improve the efficiency of detection and make LRAbL more applicable to IoT systems. In summary, our work provides a promising solution for future ubiquitous IoT localization needs.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

References

[1] V. Havyarimana, Z. Xiao, A. Sibomana, D. Wu, and J. Bai, "A fusion framework based on sparse Gaussian-wigner prediction for vehicle localization using GDOP of GPS satellites," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 2, pp. 680–689, 2020.

[2] Z. Xiao, D. Xiao, V. Havyarimana et al., "Toward accurate vehicle state estimation under non-Gaussian noises," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10652–10664, 2019.

[3] F. Chan and H. C. So, "Efficient weighted multidimensional scaling for wireless sensor network localization," *IEEE Transactions on Signal Processing*, vol. 57, no. 11, pp. 4548–4553, 2009.

[4] J. W. Sammon, "A nonlinear mapping for data structure analysis," *IEEE Transactions on Computers*, vol. C-18, no. 5, pp. 401–409, 1969.

[5] P. A. Forero and G. B. Giannakis, "Sparsity-exploiting robust multidimensional scaling," *IEEE Transactions on Signal Processing*, vol. 60, no. 8, pp. 4118–4134, 2012.

[6] L. Blouvshtein and D. Cohen-Or, "Outlier detection for robust multi-dimensional scaling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 9, pp. 2273–2279, 2018.

[7] Y. Zhu, T. Ma, Z. Li et al., "NLOS identification and correction based on multidimensional scaling and quasi-accurate detection," *IEEE Access*, vol. 7, pp. 53977–53987, 2019.

[8] Z. Xiao, H. Wen, A. Markham, N. Trigoni, P. Blunsom, and J. Frolik, "Non-line-of-sight identification and mitigation using received signal strength," *IEEE Transactions on Wireless Communications*, vol. 14, no. 3, pp. 1689–1702, 2015.

[9] A. A. Momtaz, F. Behnia, R. Amiri, and F. Marvasti, "NLOS identification in range-based source localization: statistical approach," *IEEE Sensors Journal*, vol. 14, no. 3, pp. 3745–3751, 2018.

[10] T. V Nguyen, Y. Jeong, H. Shin, and M. Z. Win, "Machine learning for wideband localization," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 7, pp. 1357–1380, 2015.

[11] J. S. Choi, W. H. Lee, J. H. Lee, J.-H. Lee, and S.-C. Kim, "Deep learning based NLOS identification with commodity WLAN devices," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 4, pp. 3295–3303, 2018.

[12] J. Fan and A. S. Awan, "Non-Line-of-Sight identification based on unsupervised machine learning in ultra wideband systems," *IEEE Access*, vol. 7, pp. 32464–32471, 2019.

[13] Y. Shang, W. Rumi, Y. Zhang, and M. Fromherz, "Localization from connectivity in sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 15, no. 11, pp. 961–974, 2004.

[14] H. Zhang, Y. Liu, and H. Lei, "Localization from incomplete euclidean distance matrix: performance analysis for the SVD-MDS approach," *IEEE Transactions on Signal Processing*, vol. 67, no. 8, pp. 2196–2209, 2019.

[15] I. Dokmanic, R. Parhizkar, J. Ranieri, and M. Vetterli, "Euclidean distance matrices: essential theory, algorithms, and applications," *IEEE Signal Processing Magazine*, vol. 32, no. 6, pp. 12–30, 2015.

[16] L. T. Nguyen, J. Kim, and B. Shim, "Low-rank matrix completion: a contemporary survey," *IEEE Access*, vol. 7, pp. 94215–94237, 2019.

[17] L. T. Nguyen, J. Kim, S. Kim, and B. Shim, "Localization of IoT networks via low-rank matrix completion," *IEEE Transactions on Communications*, vol. 67, no. 8, pp. 5833–5847, 2019.

[18] F. Xiao, W. Liu, Z. Li, L. Chen, and R. Wang, "Noise-tolerant wireless sensor networks localization via multinorms regularized matrix completion," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 3, pp. 2409–2419, 2018.

[19] X. Guo and Z. Lin, "Low-rank matrix recovery via robust outlier estimation," *IEEE Transactions on Image Processing*, vol. 27, no. 11, pp. 5316–5327, 2018.

[20] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Foundations of Computational Mathematics*, vol. 9, no. 6, pp. 717–772, 2009.

[21] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20, no. 4, pp. 1956–1982, 2010.

[22] M. Fornasier, H. Rauhut, and R. Ward, "Low-rank matrix recovery via iteratively reweighted least squares minimization," *SIAM Journal on Optimization*, vol. 21, no. 4, pp. 1614–1640, 2011.

[23] Y. Hu, D. Zhang, J. Ye, X. Li, and X. He, "Fast and accurate matrix completion via truncated nuclear norm regularization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 9, pp. 2117–2130, 2013.

[24] K. Lee and Y. Bresler, "ADMiRA: atomic decomposition for minimum rank approximation," *IEEE Transactions on Information Theory*, vol. 56, no. 9, pp. 4402–4416, 2010.

[25] J. Tanner and K. Wei, "Low rank matrix completion by alternating steepest descent methods," *Applied and Computational Harmonic Analysis*, vol. 40, no. 2, pp. 417–429, 2016.

[26] F. Wen, L. Chu, P. Liu, and R. C. Qiu, "A survey on nonconvex regularization-based sparse and low-rank recovery in signal processing, statistics, and machine learning," *IEEE Access*, vol. 6, pp. 69883–69906, 2018.

[27] C. Kümmerle and C. M. Verdun, "Escaping saddle points in ill-conditioned matrix completion with a scalable second order method," 2020, https://arxiv.org/abs/2009.02905.

[28] Z. Wen, W. Yin, and Y. Zhang, "Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm," *Mathematical Programming Computation*, vol. 4, no. 4, pp. 333–361, 2012.

[29] A. Y. Alfakih, A. Khandani, and H. Wolkowicz, "Solving euclidean distance matrix completion problems via semidefinite programming," *Computational Optimization*, vol. 12, no. 1, pp. 13–30, 1999.