Noisy range network localization based on distributed multidimensional scaling

Noisy Range Network Localization based on Distributed Multidimensional Scaling

Mingzhu Wei, Rosario Aragues, Carlos Sagues and Giuseppe C. Calafiore

***Abstract*—This paper considers the noisy range-only network localization problem, in which, measurements of relative distances between agents are used to estimate their positions in networked systems. When distance information is noisy, existence and uniqueness of location solution usually are not guaranteed. It is well known that in presence of distance measurement noise, a node may have discontinuous deformations (e.g. flip ambiguities and discontinuous flex ambiguities). Thus there are two issues that we consider in noisy localization problem. The first one is the location estimate error propagated from distance measurement noise. We compare two kinds of analytical location error computation methods by assuming that each distance is corrupted with independent Gaussian random noise. These analytical results help us to understand effects of the measurement noises on the position estimation accuracy. After that, based on multidimensional scaling theory, we propose a distributed localization algorithm to solve the noisy range network localization problem. Our approach is robust to distance measurement noise, and it can be implemented in any random case without considering the network setup constraints. Moreover, a refined version of distributed noisy range localization method is developed, which achieves a good trade-off between computational effort and global convergence especially in large-scale networks.**

***Index Terms*—Network localization, noisy range measurements, distributed algorithms, multidimensional scaling.**

I. Introduction

Estimating the positions of agents in networked systems is crucial for many applications ranging from robotics, autonomous vehicles navigation and sensor networks. Most of the applications require a precise knowledge of geometric position since the actions and observations are usually location-dependent. Often global positioning system (GPS) cannot be used since it cannot work indoors. Your system may be composed of tiny robots or that require more precision than the one given by the GPS or even your system may not operate on earth. Then in a network, the absolute node positions (with respect to a local or global reference frame) need to be estimated from partial relative measurements (e.g. distance) between nodes. There are several distance measurement techniques, which are easy to implement and low-cost, including time-difference-of-arrival (TDOA), time-of-arrival (TOA) and received signal strength (RSS) techniques [3]. Moreover, due to the spatial separation between the agents, or to occlusions, classically agents can only measure and exchange data with a subset of the team instead of the whole network. The range-only network localization problem consists of combining these distance measurements to build an estimate of the agent positions. If the internodal distance measurements are exact, the localization problem has a unique global solution. In practice, the internodal distances are often noisy, the existence and uniqueness of the network configuration cannot be guaranteed. It is well known that in presence of distance noises, a node may have discontinuous deformations (e.g. flip ambiguities and discontinuous flex ambiguities) [15], [25]. Thus there are two important considerations in noisy range network localization: i) the characterization of location error propagated from distance perturbations; ii) the design of localization algorithms with noisy range information.

Several works focus on the first issue, i.e. location error characterization. The Cramér-Rao Bound (CRB) is a popular tool for analyzing the location estimate errors. It is used for deriving lower bound position error from Gaussian and log-normal noisy distance models in time-of-arrival (TOA) and received signal strength (RSS) measurements [20]. CRB is also used to examine the behavior of error inducing parameters in multi-hop localization systems, such as, network density, anchor percentage and uncertainty of anchors [22]. Alternatively, there is a work that proposes a general location estimation error theory of noisy range based localization problem which states that if the distance noise is small enough, the noisy minimization problem has a unique solution, and the displacements of agent positions have a bound based on the distance noise [1].

There are more works discuss the second issue, i.e. network localization algorithm design with noisy range measurements. The general ways involve centralized methods and distributed methods [18]. In the centralized schemes, there is only one central unit to do the localization procedure for the whole networked sensors. For example, [5] proposes a successive refinement approach, in which Semi-Definite Programming (SDP) method tries to get a rude solution for the global network configuration in the noisy case and a gradient-descent method is added as a regularization term to have the final refinement solution. While Other works apply network multi-

This work was funded by National Science and Technology Program of China Grant 2011BAH24B06, Ministero dellIstruzione, dellUniversita e della Ricerca (MIUR) under PRIN Grant 20087W5P2K, project Ministerio de Economia y Competitividad DPI2012-32100 and grant MEC BES-2007-14772.

Mingzhu Wei is with Department of Signal Processing and System Engineering, China Electronics Technology Group Corporation No.38 Research Institute, Hefei, China. mingzhu1219@163.com

Rosario Aragues is with Departamento de Informatica e Ingenieria de Sistemas, Instituto de Investigacion en Ingenieria de Aragon, Universidad de Zaragoza, Zaragoza, Spain. raragues@unizar.es

Carlos Sagues is with Departamento de Informatica e Ingenieria de Sistemas, Instituto de Investigacion en Ingenieria de Aragon, Universidad de Zaragoza, Zaragoza, Spain. csagues@unizar.es

Giuseppe C. Calafiore is with Dipartimento di Automatica e Informatica, Politecnico di Torino, Torino, Italy. giuseppe.calafiore@polito.it

lateration techniques to search global solution, where each sensor estimates its multi-hop ranges to the nearest anchors[23]. When each sensor has multiple measurements to anchors, its coordinates can be calculated locally via multi-lateration procedure. [24] proposes two computationally efficient algorithms to refine the noisy distances based on the Cayley-Menger determinant in n-lateration networks and solves the new refined localization problem as in the noiseless case. In decentralized localization problems, each agent can implement its position estimation by itself in a network without the central computation unit and the communication cost can be reduced [10]. [13] and [14] describes fully distributed localization approaches, distributed iterative localization (DILOC) algorithm in noise-free range-based network localization problem, distributed localization random environments (DLER) algorithm in range noisy scenario and a fast distributed sensor localization with noisy distance measurements (DILAND) algorithm. Where the state of each sensor is updated with a convex combination of the states of its neighboring nodes. The coefficients are the barycentric coodinates of the convex hulls formed by the sensors and their neighbors. And the barycentric coodinates are determined from the Cayley-Menger determinants. Moreover the convergence of DLER and DILAND algorithms have been demonstrated against noisy distance measurements, communication link failures and communication channel noise. There is also a hybrid algorithm [16], which provides criteria (robust quadrilaterals) in the selection of a subgraph of the network to avoid flip ambiguities. Then the whole network is divided into small clusters. Inside each cluster, it uses a centralized optimization approach to get the agent positions. While all the clusters execute a distributed localization method in the network.

In a noisy range network localization problem, there are three kinds of input errors: range measurement errors, communication link failure errors and communication channel noises [9]. Usually the latter two errors could be removed during the algorithm iteration process. But the distance measurement errors in a sensor network will definitely propagate to the location error of each node. To reduce the location errors, accuracy distance measurement techniques and effective localization algorithms are needed. It is known that most of the existing noisy range only localization algorithms can be characterized into two classes: multi-lateration and successive refinement techniques. Better statistical performance is achieved by successive refinement methods compared to network multi-lateration approaches [18]. Therefore we are interested in searching successive refinement strategies in a distributed way. The SDP in [5] is a successive refinement approach, but it is implement in a centralized way. Which means the comptation cost in central unit will be a huge task in a large-scale network. The DILOC, DLRE in [13] and DILAND in [14] are distributed and iterative algorithms, which include two stages: triangulation and update. However they apply the idea of multi-lateration techniques in triangulation phase, in which each sensor tries to find a convex hull (triangulation) where it is included. And the update coefficients are derived from the triangulation setup. To find a convex hull, all connections in the convex hull are needed, including the distances between the sensors and vertices of the convex hull and the distances between the vertices. From this point, it is obviously that the connectivity is heavy in such a network configuration. As we already know that if nodes in a network can be located, network minimal rigidity property is the sufficient condition. While to triangulates each sensor in a network, its connectivity will be much heavier than a minimal rigid network. Moreover, the probability of each sensor triangulating itself depends on sensing range ability and nodes deployment density. Only at least one of these two parameters approaches to infinity, the probability can be close to 1. It implies that the triangulation condition is severe, i.e. if a network is minimal rigid, the network localization optimum exists, but because the connections are not enough for triangulating each sensor, the DILOC cannot be performed to get the solution. When there are distance measurement errors, the triangulation setup may fail, then the DILOC method is also ineffective. In addition, to execute DILOC, the position of anchors are restricted to form a convex hull which all sensors lie in. In practical case, the anchors may be in any arbitrary positions instead of the boundary of a sensing area.

To break through all of the limits, our contribution involves two issues in this paper. i) We will discuss the effect of range measurement noise in the final localization error by considering two different analytical error computation approaches. ii) We will propose fully distributed noisy range based network localization approaches, which belong to successive refinement methods. Unlike DILOC [13], our approaches do not impose any restriction on the anchor positions and can perform the localization in geometry global rigid graphs without heavy connectivity. Furthermore, by adding a fully distributed gradient algorithm, the refined version can achieve a good trade-off between computational effort and convergence performance.

The reminder of this paper is organized as follows. The problem setup is presented in Section II. The location error is analyzed through two different ways in Section III. A distributed multidimensional scaling based algorithm for solving noiseless network localization problem is derived as a basic technique for range only localization problem in Section IV. The two distributed noisy localization algorithms are proposed in Section V to solve the noisy range localization problem. Numerical simulation results are reported in Section VI and conclusions are drawn in Section VII.

II. Preliminary Knowledge

A. Problem Setup

Consider a network with n nodes, where m is the number of anchors which are given perfect priori position knowledge, and $n-m$ is the number of sensors whose position are unknown. Let $\mathcal{V}=\{v_1,\ldots,v_n\}$ be a set of n nodes (representing sensors, agents, robots, vehicles, etc.), and let $\mathcal{P}=\{p_1,\ldots,p_{n-m},p_{n-m+1},\ldots,p_n\}$ denote a corresponding set of positions on the Cartesian plane, where $p_i=[x_i\ y_i]^\top\in\mathbb{R}^2$ are the coordinates of the i-th node. We shall call $\mathcal{P}$ a *configuration* of nodes. Suppose that some pair of nodes, say nodes (i,j), have the possibility of measuring the relative

distance between them:

$$\|p_i - p_j\| = d_{ij}, \quad \forall\,(i,j) \in \mathcal{E}. \tag{1}$$

We denote $\mathcal{E}$ the set of unordered node pairs (i,j) such that a distance measurement exists between i and j. Let $p \doteq [p_1^\top \; p_2^\top \; \cdots \; p_n^\top]^\top$ denote the vector of node positions. Our objective is to determine a node configuration $\{p_1, \ldots, p_n\}$ that minimizes a Least-Squares of fit criterion

$$f(p) = \frac{1}{2} \sum_{(i,j)\in\mathcal{E}} \left(\|p_i - p_j\|^2 - d_{ij}^2\right)^2. \tag{2}$$

When the global minimum of f is zero, we say that exact matching is achieved, that means a configuration $\{p_1, \ldots, p_n\}$ is found that exactly matches the given distance measurements. Otherwise, no geometric node configuration can exactly match the given range data, and we say that approximate matching is achieved by the optimal configuration.

B. Distance Measurement Techniques

Since the absolute value of distances are the common information in range network localization algorithms, here we introduce two common methods that can obtain absolute distance measurements, the received signal strength (RSS) and the time of arrival (TOA).

a) Received signal strength (RSS): In wireless the signal power decays with a path-loss exponent n_p, which depends on the environment. Π_{ij} is the power received at agent i transmitted by agent j (in mW), which is modeled as log-normal. Thus the random variable $\Pi_{ij}^{RSS}(\text{dBm}) = 10\log 10\Pi_{ij}$ is Gaussian [19],

$$\Pi_{ij}^{RSS} \sim N(\bar{\Pi}_{ij}(d), \sigma_{dB}^2), \bar{\Pi}_{ij}(d) = \Pi_0^{RSS} - 10n_p \log 10(d_{ij} - d_0).$$

Where $\bar{\Pi}_{ij}(d)$ is the mean power in dBm, Π_0^{RSS} is the received power (expressed in dBm) at a short reference distance d_0. A bias-corrected estimator of distance can be defined as,

$$\hat{d}_{ij}^{RSS} = \frac{d_0}{c} 10^{\frac{\Pi_0^{RSS} - \Pi_{ij}^{RSS}}{10n_p}},$$

where c is a multiplicative bias factor, for typical channels, $c \approx 1.2$ [17]. Note that, $\hat{d}_{ij}^{RSS}$ has a log-normal distribution since $\log \hat{d}_{ij}^{RSS}$ has a Gaussian distribution. And $\hat{d}_{ij} = \log \hat{d}_{ij}^{RSS}$ is the distance measurement that we consider in the remaining of the paper.

b) Time of arrival (TOA): Time-of-arrival (TOA) is the measured time at which a signal (RF, acoustic, or other) first arrives at a receiver. The measured time delay, T_{ij}^{TOA} is the time for a signal to propagate from agent i to agent j. It can be modeled as a Gaussian distribution,

$$T_{ij}^{TOA} \sim N(d_{ij}/v_p + \mu_T, \sigma_T^2).$$

Then the distance measurement that we consider later is given by

$$\hat{d}_{ij} = (T_{ij}^{TOA} - \mu_T) \cdot v_p. \tag{3}$$

Where, v_p is the propagation velocity of signal. Based on calibration experiments and a priori knowledge of the environment, a precise estimates of the bias μ_T can be obtained, and wide-band direct-sequence spread-spectrum (DS-SS) measurements have shown $\mu_T = 10.9$ ns [18].

III. Location Estimation Error Characterization

From the typical distance measurements techniques received signal strength (RSS) and time-of-arrival (TOA), we note that the range estimates can be modeled as Gaussian distributions. So we assume the distance measurements are disturbed by Gaussian random noise and then analyze the distance noise propagation to the position estimation.

A. Linear Algebra Property of Globally Rigid Graph

The minimization objective (2) is rewritten as

$$f(p) = \frac{1}{2} \sum_{(i,j)\in\mathcal{E}} g_{ij}^2(p), \quad g_{ij}(p) \doteq \|p_i - p_j\|^2 - d_{ij}^2. \tag{4}$$

The gradient ∇g_{ij} is a row vector of n blocks, with each block composed of two entries. Thus $2n$ entries in total, and with the only non-zero terms corresponding to the blocks in position i and j:

$$\nabla g_{ij}(p) = 2[\mathbf{0}_2^\top \; \cdots \; \mathbf{0}_2^\top \; (p_i - p_j)^\top \; \mathbf{0}_2^\top \; \cdots \; \mathbf{0}_2^\top \; (p_j - p_i)^\top \; \mathbf{0}_2^\top \; \cdots \; \mathbf{0}_2^\top].$$

Then the *rigidity matrix* R of the network formation can be written:

$$R = R(p) \doteq \frac{1}{2} \begin{bmatrix} \nabla g_1(p) \\ \vdots \\ \nabla g_{|\mathcal{E}|}(p) \end{bmatrix} \in \mathbb{R}^{|\mathcal{E}|, 2|\mathcal{V}|}. \tag{5}$$

Notice that in the case when anchor nodes are present, by eliminating the columns of R corresponding to anchors and the rows of R corresponding to the edges between the anchors pairs, we have the *reduced rigidity matrix* R_r which is the submatrix obtained from R.

Lemma 1: [25] Assume that R is the rigidity matrix of a formation such that there is at least one selection of non-collinear anchors, $a_{i_1}, ..., a_{i_{|\mathcal{V}'|}}$, for $|\mathcal{V}'| \geq 3$, with the property that the formation obtained by adding inter-anchors edges is globally rigid. Then the reduced rigidity matrix R_r is of full column rank.

Let $\mathcal{E}_s$ and $\mathcal{V}_s$ be the set of sensors and the set of edges between sensors. According to lemma 1, we have $|\mathcal{E}_s| \geq 2|\mathcal{V}_s|$, which means the number of equations of eq.(1) is larger than the number of the unknown variables, i.e. with the assumption of generically globally rigid (*ggr*) communication graph, then the equation set constitutes an overdetermined system of simultaneous equations. In general, simultaneous equations have multiple solutions, but for the overdetermined set, if there is a solution, the solution is unique.

In the noiseless case, the localization problem is solvable if the communication graph is *ggr* [2]. In the noisy scenario, due to the overdetermined property of the constraint equation set, the unique global will not exist, except assuming the distance bound is typically small [1], so that the unique approximate solution can be found, which is close to the true localization. In the following paragraphs, we will investigate the measurements noise effect on the location error.

B. Analytical Location Estimation Error

In real scenarios, the distances measured by agents are noisy. We assume the distance measurements are corrupted with Gaussian random noise. The model is

$$\hat{d}_{ij} = d_{ij} + \omega_{ij}, (i,j) \in \mathcal{E}. \tag{6}$$

Where d_{ij} is the true distance between nodes i and j, $\hat{d}_{ij}$ is the measurement of d_{ij} and ω_{ij} is the independent Gaussian random measurement noise. For example, in TOA case, measurement noise is $\omega_{ij} \sim N(0, \sigma_d^2)$, with $\sigma_d^2 = v_p^2 \sigma_T^2$.

Now we consider the square of measured distances,

$$\hat{d}_{ij}^2 = d_{ij}^2 + 2 d_{ij} \omega_{ij} + \omega_{ij}^2, (i,j) \in \mathcal{E}.$$

The error part consists of two terms. One term $2 d_{ij} \omega_{ij}$ is Gaussian with mean 0 and variance $4 d_{ij}^2 \sigma_d^2$. While the second term ω_{ij}^2, that follows a Chi-square distribution with 1 degree of freedom multiplied by σ_d^2. If $|\omega| << d_{ij}$, i.e. small noises, the error part $2 d_{ij} \omega_{ij} + \omega_{ij}^2$ can be approximated to a Gaussian model. Thus the square of measured distance becomes a Gaussian model,

$$\hat{d}_{ij}^2 = d_{ij}^2 + \epsilon_{ij}, \epsilon_{ij} \sim N(0, 4 d_{ij}^2 \sigma_d^2) \tag{7}$$

Where ϵ_{ij} is the measurement noise in the square of true distance and is assumed as an independent Gaussian random model.

There are two kinds of methods to check the measurement noise propagation characterization. One is based on Cramér-Rao Bound (CRB), and another is based on approximation of residual function [25]. We will derive the CRB error of our problem and compare it with the approximated location error in [25].

1) Position estimate error analysis based on Cramér-Rao Bound: The Cramér-Rao Bound (CRB) is an algorithm-independent way to express a lower bound on the covariance of an unbiased estimator for deterministic parameters. Here we derive the CRB of position estimation when the distance measurement is modeled as eq.(6).

Let the k-th edge be the distance between nodes i and j in a global rigid graph, the location estimation problem becomes,

$$\hat{d}_k = \hat{d}_{ij} = \|p_i - p_j\| + \omega_{ij}, k = 1, \cdots, |\mathcal{E}_s|.$$

Where p_i is the true position of node i. Let $M_k(p) = M_{ij}(p) = \|p_i - p_j\|$. We stack all the equations corresponding to all the nodes and have $D = M(p) + \omega$, where $D = [\hat{d}_1, ..., \hat{d}_{|\mathcal{E}_s|}]^T$, $M(p) = [M_1(p), ..., M_{|\mathcal{E}_s|}(p)]^T$ and $\omega = [\omega_1, ..., \omega_{|\mathcal{E}_s|}]^T$.

Under the assumption of Gaussian independent random distance error, measurement probability density function (pdf) of the distance $\hat{d}_{ij}$ is

$$f_{\hat{d}_k|p}(\hat{d}_k, p) = \frac{1}{\sqrt{2\pi} \cdot \sigma_d} \cdot \exp(-\frac{1}{2\sigma_d^2} \cdot (\hat{d}_k - \|p_i - p_j\|)^2), \sigma_k = \sigma_d.$$

Since we assume that the distance measurements are uncorrelated, the covariance matrix Σ_ω is a diagonal matrix with the nonzero elements $\Sigma_\omega(k,k) = \sigma_k^2 = \sigma_d^2$. Then the measurement pdf is the vector Gaussian pdf:

$$\begin{aligned} f_{D|p}(D,p) &= \prod_{(i,j)\in\mathcal{E}_s} f_{\hat{d}_k|p}(\hat{d}_k, p) \\ &= \frac{1}{(\sqrt{2\pi})^{|\mathcal{E}_s|} \cdot \prod_{(i,j)\in\mathcal{E}_s} \sigma_d} \\ &\cdot \exp(\sum_{(i,j)\in\mathcal{E}_s} \frac{-1}{2\sigma_d^2} \cdot (\hat{d}_k - \|p_i - p_j\|)^2) \\ &= \frac{1}{(\sqrt{2\pi})^{|\mathcal{E}_s|} \cdot |\Sigma_\omega|^{\frac{1}{2}}} \\ &\cdot \exp(-\frac{1}{2}[D - M(p)]^T \Sigma_\omega^{-1} [D - M(p)]). \end{aligned} \tag{8}$$

Based on eq.(8), we find the Fisher information matrix :

$$\begin{aligned} J(p) &= E[(\nabla_p \ln f_{D|p}(p)) \cdot (\nabla_p \ln f_{D|p}(p))^T] \\ &= [G'(p)]^T \Sigma_\omega^{-1} [G'(p)], \end{aligned}$$

where

$$[G'(p)]_{k,n} = \frac{\partial M_k(p)}{\partial p_n} = \begin{cases} \frac{x_i - x_j}{d_{ij}}, p_n = x_i; \\ \frac{y_i - y_j}{d_{ij}}, p_n = y_i; \\ \frac{x_j - x_i}{d_{ij}}, p_n = x_j; \\ \frac{y_j - y_i}{d_{ij}}, p_n = y_j; \\ 0, \text{otherwise.} \end{cases}$$

According to the definition of reduced rigidity matrix R_r, we have $G'(p) = R_r \cdot \Sigma_d^{-1}$,where Σ_d is a diagonal matrix with the nonzero elements $\Sigma_d(k,k) = d_{ij}$. The Fisher information matrix is

$$J(p) = R_r^T \cdot \Sigma^{-1} \cdot R_r, \tag{9}$$

where $\Sigma = \Sigma_d \cdot \Sigma_\omega \cdot \Sigma_d$.

So the Cramér-Rao Bound matrix is

$$C_{\text{CRB}} = J^{-1}(p) = [R_r^T \cdot \Sigma^{-1} \cdot R_r]^{-1}. \tag{10}$$

The formal position estimate error covariance matrix defined as $C_p = E[(\hat{p} - p)(\hat{p} - p)^T]$, which is lower bounded by the Cramér-Rao Bound, i.e. $C_p \geq C_{\text{CRB}}$.

2) Position estimate error analysis based on cost function linearization: By linearizing the cost function (2) in the noisy case, [25] derive the approximate location error characterization. They define the following residual function, $\vartheta_{ij}(p)$ as

$$\vartheta_{ij}(p) = \|\hat{p}_i - \hat{p}_j\|^2 - \hat{d}_{ij}^2, \quad \hat{p}_i = p_i + \delta_{p_i},$$

where δ_{p_i} is the position estimate error of agent i in the noisy case. Then

$$\begin{aligned} \vartheta_{ij}(p) &= \|(p_i + \delta_{p_i}) - (p_j + \delta_{p_j})\|^2 - (d_{ij}^2 + \epsilon_{ij}) \\ &= 2(p_i - p_j)^T (\delta_{p_i} - \delta_{p_j}) + \|\delta_{p_i} - \delta_{p_j}\|^2 - \epsilon_{ij}. \end{aligned}$$

Based on [1, Theorem 3.2], with sufficiently small distance measurement error, the location estimation error is bounded in a typical small vicinity of the true position. By omitting the effect of the second order term, $\|\delta_{p_i} - \delta_{p_j}\|^2$, they have

$$\theta_{ij}(p) = (p_i - p_j)(\delta_{p_i} - \delta_{p_j})^T - \epsilon_{ij}/2.$$

The two summands ϑ and θ are the same for a particular edge (i,j) by neglecting the factor of 2 and the second order term.

Stacking all the equalities in lexico-graphical order, then

$$\theta(p) = R \cdot \delta_p - \epsilon/2, \tag{11}$$

In the noisy range localization problem, by minimizing the following problem,

$$\min_{\delta_p} \|R \cdot \delta_p - \frac{\epsilon}{2}\|^2, \tag{12}$$

The solution δ_p^* can be found, which satisfies $R \cdot \delta_p^* = \epsilon/2$.

Rigidity matrix R is often not square in general, then it is not invertible, thus the error in the agent position caused by the distance measurement noise will make no sense. But if we put some constraint conditions (for example, some agents are fixed, some distances will not change), a submatrix of R will map the position estimate errors to the distance measurement errors. If the distance error is deterministic, then the deterministic position error is

$$\delta_p^* = (R_r^T R_r)^{-1} R_r^T \cdot \epsilon_r/2. \tag{13}$$

Where R_r is reduce rigid matrix, i.e. the submatrix of R. ϵ_r is the error vector in the square of the length of edges (connecting those edges with at least one non-anchor). Actually each element of ϵ_r is in the form of eq.(7).

With the Gaussian independent random distance noise, whose covariance Σ_ϵ is a diagonal matrix with the nonzero elements $\Sigma_\epsilon(k,k) = 4d_k^2\sigma_d^2$, the covariance of approximate position estimate error $cov(\delta_p^*)$ is derived as follows,

$$\begin{aligned} cov(\delta_p^*) &= \frac{1}{4}(R_r^T R_r)^{-1} R_r^T cov(\epsilon_r)((R_r^T R_r)^{-1} R_r^T)^T \\ &= \frac{1}{4}(R_r^T R_r)^{-1} R_r^T \cdot \Sigma_\epsilon \cdot ((R_r^T R_r)^{-1} R_r^T)^T \\ &= (R_r^T R_r)^{-1} R_r^T \cdot \Sigma \cdot ((R_r^T R_r)^{-1} R_r^T)^T. \end{aligned} \tag{14}$$

Proposition 1: In a generally global rigid communication graph G, with the assumption of independent and identically distributed (i.i.d.) Gaussian random distance measurement noise, if the distance errors are typically small, the approximate location error is equal to CRB error, i.e. the CRB error can be used as a benchmark of localization accuracy for particular algorithms.

Proof. With Gaussian random distance measurement noise, according to the equations (10) and (14), in order to prove $C_{\text{CRB}} = cov(\delta_p^*)$, we need to prove

$$(R_r^T \cdot \Sigma^{-1} \cdot R_r) \cdot [(R_r^T R_r)^{-1} R_r^T \cdot \Sigma \cdot ((R_r^T R_r)^{-1} R_r^T)^T] = I. \tag{15}$$

The leftside of eq.(15) is reformed as,

$$\begin{aligned} &(R_r^T \cdot \Sigma^{-1} \cdot R_r) \cdot (R_r^T R_r)^{-1} R_r^T \cdot \Sigma \cdot ((R_r^T R_r)^{-1} R_r^T)^T \\ &= R_r^T \cdot \Sigma^{-1} \cdot [R_r \cdot (R_r^T R_r)^{-1} R_r^T)] \cdot \Sigma \cdot ((R_r^T R_r)^{-1} R_r^T)^T. \end{aligned} \tag{16}$$

Let $A = R_r \cdot (R_r^T R_r)^{-1} R_r^T$. It is easy to indicate that $A \cdot A = A$. A is non-singular matrix since the determinants of R_r and $(R_r^T R_r)^{-1} R_r^T$ are non-zero. Then $A = I$ and the eq.(16) is

$$\begin{aligned} &(R_r^T \cdot \Sigma^{-1} \cdot R_r) \cdot (R_r^T R_r)^{-1} R_r^T \cdot \Sigma \cdot ((R_r^T R_r)^{-1} R_r^T)^T \\ &= R_r^T \cdot \Sigma^{-1} \cdot [R_r \cdot (R_r^T R_r)^{-1} R_r^T)] \cdot \Sigma \cdot ((R_r^T R_r)^{-1} R_r^T)^T \\ &= R_r^T \cdot \Sigma^{-1} \cdot I \cdot \Sigma \cdot R_r (R_r^T R_r)^{-1} \\ &= R_r^T R_r (R_r^T R_r)^{-1} \\ &= I. \end{aligned}$$

Then, $C_p = C_{\text{CRB}} = cov(\delta_p^*)$ is proved. □

Remark 1: As addressed in proposition 1, CRB can evaluate the noise propagation in the case where the distance noise is typically small. Under Gaussian random distance error model and generally global rigid graph assumption, CRB error is approximately linear with error variance σ_d^2. Since the reduced rigidity matrix R_r in covariance matrix defined by the communication graph, the network parameters (such as, network size or anchor number) have crucial effects on CRB error, so as the approximate location error in [21].

IV. Distributed Multidimensional Scaling based Algorithm

In Multidimensional Scaling (MDS) theory, the data is quantitative and the proximities of objects are treated as distances in a Euclidean space [6], [12]. The goal of multidimensional scaling method is to find a configuration of points in a multidimensional space such that the internodal distances are related to the provided proximities by some transformation (e.g. a linear transformation) [12]. If the proximity data were measured without error in a Euclidean space, then classical metric MDS would exactly recreate the configuration of points [2], [15]. In practice, the technique tolerates error gracefully due to the overdetermined nature of the solution. The MDS algorithm is used to regulate the local formation-shape to the global one in formation-shape stabilization problem, and it is implemented in a decentralized way [9]. Based on the idea of [9], we will address how to use it in a range only network localization problem to find the global network configuration with random initial position instead of local solutions. Since the error tolerance property of the MDS, we will use it to the localization problem where the distance estimates can be very rough indeed in the section V.

A. Classical Multidimensional Scaling Algorithm

Inspired by the multidimensional scaling techniques, we note that the desired network configuration is the global minimizer of the raw stress function $S(p) : (R^2)^n \to R$,

$$S(p) = \frac{1}{2} \sum_{(i,j)\in\mathcal{E}} \left(\|p_i - p_j\| - d_{ij}\right)^2. \tag{17}$$

The S-Stress function $S_s(p)$ of $S(p)$ is in fact the cost function $f(p)$ in eq.(2), i.e. $S_s(p) = f(p)$. Since the stress function $S(p)$ has the same global minimizers as the S-Stress function $S_s(p)$, so as $f(p)$. Here instead we focus on searching the solution of the raw stress function $S(p)$. If we compute the partial derivative and design a gradient-descent coordinate algorithm to minimize the stress function $S(p)$. It is prone to local minima. An alternative strategy involves the construction of majorization functions that are easier to optimize.

Given any column vector of agents' positions, $z = [z_1^T, ..., z_n^T]^T \in R^{2n}$, and a communication graph $\mathcal{G}$, we let $\mathcal{G}^z$ be the weighted graph with adjacency matrix $A(\mathcal{G}^z)$ with entries,

$$a_{ij} = \begin{cases} \frac{d_{ij}}{\|z_i - z_j\|}, & (i,j) \in \mathcal{E}, \\ 0, & (i,j) \notin \mathcal{E}. \end{cases}$$

Then the stress majorization function $F^z(p) : R^{2n} \to R$ is

$$F^z(p) = \mathrm{tr}(p^T L(\mathcal{G})p) - 2\mathrm{tr}(p^T L(\mathcal{G}^z)p) + \frac{1}{2}\sum_{(i,j)\in\mathcal{E}} d_{ij}^2. \quad (18)$$

Where $L(\mathcal{G})$ is the Laplacian matrix related to communication graph $\mathcal{G}$ with the element l_{ij} satisfying

$$\begin{cases} l_{ij} = \begin{cases} -1, & (i,j) \in \mathcal{E}, \\ 0, & \text{otherwise}, \end{cases} \\ l_{ii} = |\mathcal{N}_i|. \end{cases}$$

And $L(\mathcal{G}^z)$ is the Laplacian matrix of the weighted graph $\mathcal{G}^z$ with the element l_{ij}^z satisfying

$$\begin{cases} l_{ij}^z = \begin{cases} -\frac{d_{ij}}{\|z_i - z_j\|}, & (i,j) \in \mathcal{E}, \\ 0, & \text{otherwise}, \end{cases} \\ l_{ii}^z = \sum_{j\in\mathcal{N}_i} \frac{d_{ij}}{\|z_i - z_j\|}. \end{cases}$$

Alternatively the Kronecker product of $F^z(p)$ can be expressed as

$$F^z(p) = p^T(L(\mathcal{G}) \otimes I_2)p - 2p^T(L(\mathcal{G}^z) \otimes I_2)z + \frac{1}{2}\sum_{(i,j)\in\mathcal{E}} d_{ij}^2. \quad (19)$$

Proposition 2: [9] Given z and a communication graph $\mathcal{G}$, the following statements hold:
1) The gradient and Hessian of $F^z(p)$ are respectively,

$$\bigtriangledown F^z = 2(L(\mathcal{G}) \otimes I_2) \cdot p - 2(L(\mathcal{G}^z) \otimes I_2) \cdot z,$$
$$\bigtriangledown^2 F^z = 2L(\mathcal{G}) \otimes I_2.$$

2) The function $F^z(p)$ is global convex.
3) $p \in R^{2n}$ is a global minimizer of $F^z(p)$, if and only if

$$(L(\mathcal{G}) \otimes I_2) \cdot p = (L(\mathcal{G}^z) \otimes I_2) \cdot z.$$

Proposition 3: [9] Given an undirected graph $\mathcal{G}$, for any sets $\mathcal{P} = \{p_1, ..., p_n\} \in R^{2n}$ and $\mathcal{Z} = \{z_1, ..., z_n\} \in R^{2n}$, $S(p) \le F^z(p)$. Moreover, if $p = z$, then $F^p(p) = S(p)$.

From proposition 3, we can know if the global solution of $F^z(p)$ is achieved, then it is the solution of $S(p)$. Based on proposition 2, it is easy to deduce an update formula of the SMACOF (Scaling by MAjorizing a Complicated Function) algorithm [12],

$$(L(\mathcal{G}) \otimes I_2) \cdot p = b(z). \quad (20)$$

Where $b(z)$ is a matrix function of current node position estimation z and $b(z) = (L(\mathcal{G}^z) \otimes I_2) \cdot z$.

B. Distributed Multidimensional Scaling Algorithm

Given an invertible matrix $U \in R^{n\times n}$ and a vector $b \in R^n$, we consider a linear system $U \cdot x = b$. With Jacobi Over Relaxation (JOR) method [9], the $\tau+1$ iteration value of i-th element of x, represented by symbol $x_i(\tau+1)$ is obtained using the following equation

$$x_i(\tau+1) = (1-h)x_i(\tau) + h \cdot \frac{1}{u_{ii}}(b_i - \sum_{j\neq i} u_{ij}x_j(\tau)). \quad (21)$$

with $\tau \in \mathbb{Z} \ge 0, x(0) \in R^n$, and a relaxation factor $h, h \in (0, 1)$.

The convergence properties of the JOR method can be characterized in terms of eigenvalues of the matrix describing the linear iteration [4].

Proposition 4 (DMDS Method): Given an undirected communication graph $\mathcal{G}$ and an initial position guess, each node converges to true position by executing the distributed multidimensional scaling (DMDS) method with the exact distance measurements $d_{ij}, (i,j) \in \mathcal{E}$. The update rule is

$$p_i(\tau+1) = p_i(\tau) - \frac{h}{|\mathcal{N}_i|} \cdot \sum_{j\in\mathcal{N}_i} (1 - \frac{d_{ij}}{\|p_i(\tau) - p_j(\tau)\|}) \cdot (p_i(\tau) - p_j(\tau)). \quad (22)$$

Proof. From eq.(20), the centralized update formula at iteration τ with position estimation $p(\tau)$ reads,

$$(L(\mathcal{G}) \otimes I_2) \cdot p(\tau+1) = b(p(\tau)). \quad (23)$$

As stated before, JOR method can solve a linear function with a distributed way and keep the global convergence. If we can implement our problem with the JOR method, it will persist the above property.

Give the communication graph $\mathcal{G}$, each node i has current access of $b_i(p(\tau))$ and l_{ii}. Moreover it collects l_{ij} and $l_{ij}^{p(\tau+1)}$ by communicating with its neighbors $j \in \mathcal{N}_i$. Then distributed implementation for solving eq.(23) is in the following update form,

$$p_i(\tau+1) = (1-h) \cdot p_i(\tau) - \frac{h}{l_{ii}} \cdot [\sum_{j\in\mathcal{N}_i} l_{ij} \cdot p_j(\tau) - b_i(p(\tau))]. \quad (24)$$

Where the weighted Laplacian matrix $L(\mathcal{G}^{p(\tau)})$ is corresponding to the position estimation $p(\tau)$ at iteration τ, with the entries,

$$\begin{cases} l_{ij}^{p(\tau)} = \begin{cases} -\frac{d_{ij}}{\|p_i(\tau) - p_j(\tau)\|}, & (i,j) \in \mathcal{E}, \\ 0, & \text{otherwise}. \end{cases} \\ l_{ii}^{p(\tau)} = \sum_{j\in\mathcal{N}_i} \frac{d_{ij}}{\|p_i(\tau) - p_j(\tau)\|}. \end{cases} \quad (25)$$

Then, the eq.(24) is rewritten as eq.(26).
So the proposition 4 is proved. □

V. Distributed Range Localization Approaches in Noisy Case

In previous section, we derived the DMDS approach in the noiseless range based network localization. In this section, we consider the localization problem where the distances are noisy. We apply a distance information estimation scheme, which provide more accurate distance measurements to the distributed localization algorithms. To accelerate the convergence speed in actual applications, we propose a refined distributed noisy range localization algorithm.

A. Distributed MDS Method in Noisy Scenario

a) Distance estimation: The localization schemes exclusively use current measurements of internodal distances to compute the position estimation. So if distance information is high uncertain, the position estimation accuracy will be affected. Here we use the information from past distance measurements to compute a more accurate version of the distance

$$\begin{aligned} p_i(\tau+1) &= (1-h)\cdot p_i(\tau) - h\cdot\frac{1}{l_{ii}}\cdot(\sum_{j\in\mathcal{N}_i} l_{ij}\cdot p_j(\tau) - \sum_{j\in\mathcal{N}_i}\frac{d_{ij}}{\|p_i(\tau)-p_j(\tau)\|}(p_i(\tau)-p_j(\tau))) \\ &= p_i(\tau) - h\cdot\frac{1}{|\mathcal{N}_i|}\cdot\sum_{j\in\mathcal{N}_i}(1-\frac{d_{ij}}{\|p_i(\tau)-p_j(\tau)\|})\cdot(p_i(\tau)-p_j(\tau)). \end{aligned} \tag{26}$$

$\bar{d}_{ij}(\tau)$ at each update iteration τ. These estimated distance $\bar{d}_{ij}(\tau)$ is thus a function of the entire past measurements $\hat{d}_{ij}(s)_{s\le\tau}$ [14].

Since at time τ, we have knowledge of previous measurements $\hat{d}_{ij}(s)_{s\le\tau}$ between nodes i and j. With $\bar{d}_{ij}(0)=\hat{d}_{ij}(0)$, we use the following update rule to estimate an accurate distance estimate $\bar{d}_{ij}(\tau)$ which is the one used by the localization algorithm,

$$\bar{d}_{ij}(\tau) = \frac{1}{\tau}\sum_{s\le\tau}\hat{d}_{ij}(s) = \frac{\tau-1}{\tau}\bar{d}_{ij}(\tau-1) + \frac{1}{\tau}\hat{d}_{ij}(\tau). \tag{27}$$

Assumption 1: [14] Let $\{K(\tau)_{\tau\ge0}\}$ be any sequence of internodal distance measurements collected over time, then there exist a sequence of estimates $\bar{d}(\tau)_{\tau\ge0}$, such that for all τ, $\bar{d}(\tau)$ can be computed efficiently from $\{K(s)_{s\le\tau}\}$ and have $P[\lim_{\tau\to\infty}\bar{d}(\tau)=d]=1$, where d is the true distance measurement.

b) Relaxation parameter: The relaxation parameter h is chosen as a time varying function, $h(\tau)$, satisfing $h(\tau)\ge0$, $\lim_{\tau\to\infty}h(\tau)=0$, and $\sum_\tau h(\tau)=\infty$. In particular, we consider the following choice: for $h>0$ and $0<\delta\le1$,

$$h(\tau) = \frac{h}{(\tau+1)^\delta}. \tag{28}$$

Where the weights decay to zero, but not too fast, which is common in the adaptive control and signal processing.

We apply the distance estimate scheme and time-varying relaxation parameter to distributed multidimensional scaling algorithm eq.(22) to solve the noisy range based network localization problem. The new algorithm is summarized as Distributed Noisy Range Localization (DNRL) Method.

Algorithm 1: DNRL method

Inputs

- $p^{(0)}$: Initial guess of node positions;
- $\tau_{\max}$: Maximum number of update iterations;
- η_{abstol}: Absolute update tolerance.

Initialization Let $\tau=0$, $p_i(\tau)=p_i(0)$, for all $i=1,...,n-m$; let the initial distance information be equal to the initial measurement, i.e. $\bar{d}_{ij}(0)=\hat{d}_{ij}(0),\forall(i,j)\in\mathcal{E}$; mark all states of nodes as active.

Repeat while $\tau<\tau_{\max}$, and at least one node is active:

For i = 1 to $n-m$,

- Compute update coefficients $l_{ij}^{p(\tau)}$ and $l_{ii}^{p(\tau)}$ as in eq.(25);
- Update the node position estimate according to eq.(29), but with $h(\tau)$ satisfying eq.(28),

$$p_i(\tau+1) = p_i(\tau) - \frac{h(\tau)}{\mathcal{N}_i}\cdot\sum_{j\in\mathcal{N}_i}(1 - \frac{\bar{d}_{ij}(\tau)}{\|p_i(\tau)-p_j(\tau)\|})\cdot(p_i(\tau)-p_j(\tau)). \tag{29}$$

- If $\|p_i(\tau)-p_i(\tau-1)\|\le\eta_{\text{abstol}}$, then mark node i as "inactive";

end for.

Let $\tau=\tau+1$.

Update distance sequence $\bar{d}_{ij}(\tau)$ as formula eq.(27).

End repeat

B. Refinement by a distributed gradient based method

The DNRL approach almost surely converges to the optimal solution, but its convergence speed is slow so that it needs costly computational effort. In the practical application, we hope to apply a method to get global solution with fast convergence speed. We combine a distributed gradient method with our DNRL algorithm to achieve this goal.

c) Distributed Gradient based Approach: Gradient-based approaches are based on successive refinement steps to find the global minimum of a non-linear cost function associated to a network localization problem. The distributed version of gradient method that we proposed in [7] is an iterative method with fast convergence speed, which only requires first-order information. The distributed gradient (DG) based method is proved to converge to the same solution as its centralized counterpart. But it provides the benefits of a fully decentralized scheme that can be implemented autonomously by the networked agents.

It is known that $f(p)$ has the same global minimizer as $S(p)$. After executing the DNRL algorithm, the raw position estimation is close to the global solution. DG method can converge to the true position estimation with a good initial guess and exact distance measurement information, . Therefore the distributed gradient refinement approach is used to regulate this raw node position estimation to the almost true position, avoiding the local minimizer.

The DG method is composed of a consensus phase and an update phase. In the consensus phase, distributed implementation of Barzilai-Borwein stepsizes of each node is in the following form,

$$\alpha_i(\tau(t)) \doteq \frac{\rho_i(\tau(t))}{\psi_i(\tau(t))}. \tag{30}$$

It is an approximation of the common step size α_τ, and t is the consensus iteration.

The numerator and denominator of $\alpha_i(\tau(t))$ are defined as follows: at "time" $\tau(0)$ each node i, $i=1,\ldots,n$, initializes two scalar values:

$$\rho_i(\tau(0)) = \|p_i^{(\tau)}-p_i^{(\tau-1)}\|^2, \tag{31}$$

$$\psi_i(\tau(0)) = (p_i^{(\tau)}-p_i^{(\tau-1)})^\top(\nabla_i f(p^{(\tau)})-\nabla_i f(p^{(\tau-1)})), \tag{32}$$

and then starts a series of consensus iterations, exchanging data with its neighbors:

$$\rho_i(\tau(t+1)) = W_{ii}\rho_i(\tau(t)) + \sum_{j\in\mathcal{N}_i} W_{ij}\rho_j(\tau(t)), \tag{33}$$

$$\psi_i(\tau(t+1)) = W_{ii}\psi_i(\tau(t)) + \sum_{j\in\mathcal{N}_i} W_{ij}\psi_j(\tau(t)), \quad t = 0, 1, \ldots, \tag{34}$$

where $\nabla_i f(p)$ denote the i-th 1×2 block in the gradient $\nabla f(p)$,

$$\nabla_i f(p) = \sum_{j\in\mathcal{N}_i} (p_i - p_j)^\top g_{ij}(p). \tag{35}$$

W is a symmetric doubly stochastic matrix, compatible with graph $\mathcal{G}$, with its weights as follows:

$$W_{ij} = \begin{cases} \dfrac{1}{\max(|\mathcal{N}_i|, |\mathcal{N}_j|)} & \text{if } (i,j)\in\mathcal{E},\ i\neq j, \\ 1 - \displaystyle\sum_{j\in\mathcal{N}_i\setminus i} W_{ij} & \text{if } i = j, \\ 0 & \text{otherwise,} \end{cases}$$

where $|\mathcal{N}_i|$ denotes the cardinality of $\mathcal{N}_i$.

During the update phase, this common parameter α_τ is used to actually update the current position estimate, then each node would be able to locally update its estimated position according to the distributed gradient rule [8]:

$$p_i^{(\tau+1)} = p_i^{(\tau)} - \alpha_\tau \nabla_i f(p^{(\tau)}), i = 1, \ldots, n. \tag{36}$$

The refined DNRL (R-DNRL) algorithm is a combination of DNRL and DG algorithms. First we run Algorithm 1 with small fixed iteration number, $\tau_{1_{\max}}$, then output the raw position estimation p_{DNRL} and the distance estimation information $\bar{d}_{\text{DNRL}}$ which gets closer to the true value d. Then we run the DG algorithm, where the initial position guess is p_{DNRL} and the required distance information is fixed as $\bar{d}_{\text{DNRL}}$. The practical distributed noisy range localization algorithm works as follows.

Algorithm 2: R-DNRL method

Inputs

- $p^{(0)}$: Initial guess of node positions;
- $\tau_{1_{\max}}$: Maximum number of update iterations in DNRL phase;
- $\tau_{\max}$: Maximum number of update iterations in DG refinement phase;
- η_{abstol}: Absolute update tolerance.

Initialization Let $\tau = 0$, $p_i(\tau) = p_i(0)$, for all $i = 1, ..., n - m$; let the initial distance information be equal to the initial measurement, i.e. $\bar{d}_{ij}(0) = \hat{d}_{ij}(0), \forall (i,j)\in\mathcal{E}$; mark all states of nodes as active.

DNRL procedure:

Execute the DNRL algorithm for $\tau_{1_{\max}}$ iterations.

Outputs

- p_{DNRL}: Raw position estimation;
- $\bar{d}_{\text{DNRL}}$: Approximated distance information used in DG algorithm.

DG refinement:

Inputs

- $t_{\max}$: Maximum number of consensus iterations;
- τ_{wup}: Maximum number of warm-up iterations;

Warm-up

Set the position estimates $p(\tau) = p_{\text{DNRL}}$; let the required distance $\bar{d} = \bar{d}_{\text{DNRL}}$; mark all nodes as "active".

- Fix $\alpha_\tau = \tilde{\alpha}$ (a small number, and equal for all nodes);
- Update all node position estimates according to (36) for τ_{wup} iterations ;

Repeat while $\tau \geq \tau_{\max}$, and at least one node is active.

(Stepsize computation) Repeat while $t < t_{\max}$:

- Initialize $\rho_i(\tau(0))$ and $\psi_i(\tau(0))$ as eq.(31) and eq.(32);
- Update $\rho_i(\tau(t))$, $\psi_i(\tau(t))$ according to eq.(33) and eq. (34) for $t_{\max}$ iterations;

(Gradient update) At each active node:

- Compute α_τ according to eq.(30);
- Update position estimate according to eq.(36); If $\|p_i(\tau) - p_i(\tau - 1)\| \leq \eta_{\text{abstol}}$, then mark node i as "inactive";

End repeat

Remark 2: DNRL approach can converge to almost true position estimation avoiding the local minima, but it needs onerous iterations. When we select a small number of DNRL iterations, DNRL approach outputs raw global position estimations and relatively accurate distance estimations. Since DG method can accelerate the convergence speed with noise-free distance information, DG refinement could find the optimum network configuration where location errors are in the tolerance bound by using the almost true distance estimations. Thus the proposed R-DNRL scheme can work well with a good trade-off between computational effort and location error.

VI. Numerical Experiments

We now present some numerical tests on the proposed decentralized network localization methods. In first experiment, we study the performance of distributed multidimensional scaling (DMDS) approach. In second experiment, we choose a small-scale network to investigate the properties of the distributed noise range network localization (DNRL) approach and the refined version (R-DNRL technique). In third experiment, we extend to large-scale networks and evaluate the performance of R-DNRL technique.

We assume the distance measurement $\hat{d}_{ij}$ between nodes i and j is corrupted by Gaussian noise with model as eq.(6). Where σ_d is the standard deviation of distance measurement. The initial guess of each sensor position in networks for algorithms is drawn from a multivariate normal distribution centered at the true node positions with standard deviation σ_p.

A. Experiment 1

In this experiment we investigate the error propagation characterizations of three methods, SDP, DILOC and DMDS by comparing their localization errors with CRB errors. If a network is generically global rigid, it can find the optimum localization solution. SDP and DMDS methods can achieve the only optimum configuration in such a generically global rigid graph. But for DILOC method, to triangulate each sensor, the network connectivity is much heavier than the generically global rigid graph. Moreover all sensors must stay in the convex hull formed by anchors. Instead there is no anchor position limit during SDP and DMDS methods running. To compare the three localization algorithms, we will consider a

graph with heavier edge connections than generically global rigid graph so that it can be located by all the three algorithms. Fig. 1 plots such a network where $n = 50$ nodes are distributed in a triangle $[0, 1] \times [0, 1]$, $m = 3$ anchors are at the external vertices of the formation and all sensors are triangulated with the solid lines.

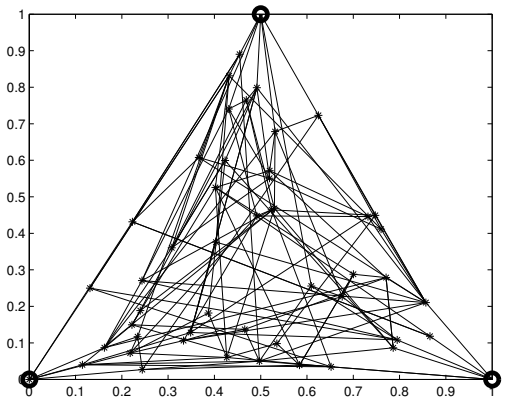

Fig. 1. The $N = 50$ network graph and the respective triangulation set. The stars represent true unknown sensor positions; the circles represent the $m = 3$ anchor positions; the lines represent triangulation set connections.

We define the root-mean-square (RMS) positioning error ϕ for a n-nodes network where $n - m$ sensor positions need to be determined,

$$\phi = \sqrt{\frac{\sum_{i=1}^{n-m} \|\hat{p}_i - p_i\|^2}{n-m}}, \tag{37}$$

where p_i is the true position of node $i, i \in 1, .., n-m$, $\hat{p}_i$ is the corresponding estimated position and m is the number of anchors.

The CRB error is given by

$$\phi_{\text{CRB}} = \sqrt{\frac{\sum_{i=1}^{n-m}([C_{\text{CRB}}]_{2i-1,2i-1} + [C_{\text{CRB}}]_{2i,2i})}{n-m}}. \tag{38}$$

It is known that if the location error nearly achieves to the CRB error, there is no need to improve the accuracy of a localization algorithm. Here we compare CRB errors with the location errors of DMDS, SDP and DILOC approaches. We run 100 Monte-Carlo simulations for distance measurement errors σ_d =0.005, 0.01, 0.03, 0.05, 0.07, 0.1. Fig. 2 shows that the location errors of SDP and DMDS algorithms are close to the CRB errors especially with small distance noise. As the distance measurement errors increase, the location errors of SDP and DMDS approaches linearly increase as what CRB errors do. However the location errors of DILOC method are much greater than CRB errors and do not increase linearly when distance errors become greater. It implies that DILOC is more sensitive to the distance errors than SDP and DMDS methods, because DILOC is not a successive refinement method and it applies the trilateration technique at setup phase. Thus SDP and DMDS methods have much less algorithm errors than DILOC method in noisy range network localization.

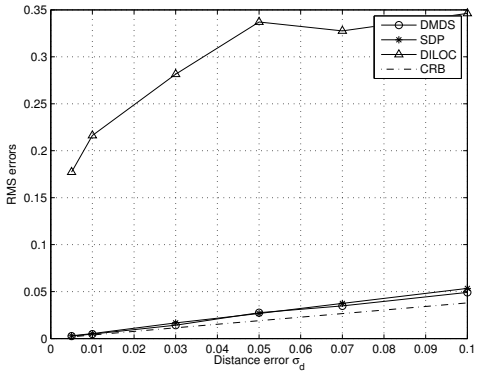

Fig. 2. Location error comparison of SDP, DILOC and DMDS algorithms with the Cramér-Rao Bound errors. Where initial position guess $\sigma_p = 0.1$.

Table I shows the computational efforts of the three techniques. As a centralized localization algorithm SDP need much more time to obtain the optimum solution than the distributed ones, DMDS and DILOC. DILOC needs more computational efforts than DMDS. Moreover as the distance errors increase, the location errors of DILOC become larger. But the location errors of DMDS change little with distance noise. It shows that DMDS is more robust than DILOC in noisy range case. In addition, when we take account of network connectivity and anchor position limits, we do not think DILOC is a common available localization method. About SDP method, though it is as good as DMDS from aspect of minor localization algorithm errors, it is a centralized algorithm which needs much computation cost than a distributed algorithm. Therefore it is obviously DMDS is a suitable localization method in the distributed noisy range based network localization system.

B. Experiment 2

In order to work on more realistic graphs, we use random geometric graphs, that are graphs in which nodes are deployed at random in the square plane, and an edge exists between a pair of nodes if and only if their geometrical distance is smaller than sensing radius, r. It has been proved in [11] that if $r > 2\sqrt{2}\sqrt{\frac{\log(n)}{n}}$, the graphs produced by the previous technique in a square $[0, 1] \times [0, 1]$ are geometry globally rigid with high probability.

In this section, we pick a small-scale network to investigate the detailed properties of DNRL and R-DNRL algorithms in the case where the distances are noisy. In a geometry globally rigid graph, the only optimum global configuration can be obtained. But if we apply DIRE and DILAND algorithms (two variations of DILOC method in noisy range case), their triangulation setup phase can not be implemented because there is not enough connections between nodes as what they need so that these two algorithms are ineffective in such a network. However DNRL and R-DNRL algorithms can work well. In the numerical simulations, we consider geometry globally rigid graphs with $n = 30$ nodes that are distributed in a square $[0, 1] \times [0, 1]$, where the sensing range is $r = 0.6$ and the $m = 3$ anchor nodes are selected randomly in the network. An example of random geometric graph is shown in Fig. 3.

TABLE I
COMPUTATION EFFORT OF SDP, DILOC AND DMDS METHODS

σ_d	0.005	0.01	0.03	0.05	0.07	0.1
SDP method (seconds)	192.7494	232.6691	184.2966	183.0561	212.7865	239.5150
DILOC method (seconds)	0.6333	0.7316	0.5623	0.5565	15.3241	17.0462
DMDS method (seconds)	0.0619	0.0626	0.0728	0.0733	0.0743	0.0750

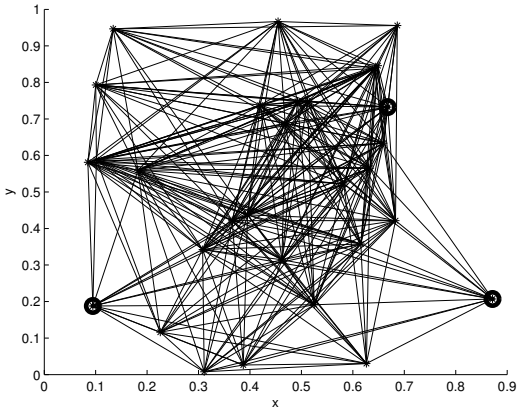

Fig. 3. The $N = 30$ random geometric communication graph. The stars represent true unknown sensor positions; the circles represent the anchor positions; the lines represent distance connections.

In order to measure the localization effectiveness of each node in the noisy range case, we define the local position error at the each node as the Euclidean distance between the estimated position at iteration τ and the true position of the node. We assume that distance measurement $\hat{d}_{ij}(\tau)$ at time τ between nodes i and j is corrupted by Gaussian noise σ_d. The required distance information $\bar{d}_{ij}(\tau)$ in localization is estimated using eq. (27). The time varying relaxation parameter is set as $h(\tau) = 0.6/(\tau+1)^{0.5}$ and the initial guess parameter is set as $\sigma_p = 0.1$. Here we change the distance error σ_d with different test values $\sigma_d = 0.005$, 0.01, 0.03, 0.05, 0.07, 0.1.

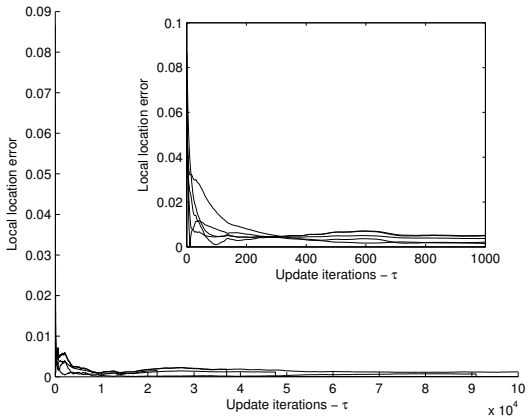

Fig. 4. The local position error convergences of five nodes by using DNRL algorithm. Where initial position guess $\sigma_p = 0.1$ and distance measurement error $\sigma_d = 0.1$.

The DNRL algorithm can converge to the almost true global configuration, but with great computational effort. For example, in Fig. 4 the random picked 5 nodes of the network can finally converge to the true position with a suitable iteration number, but iteration number is large which implies the great computational effort. Instead when the R-DNRL method is used, the convergence of each node to the true position can be guaranteed with much less computational effort. Table II compares the simulation results of the two methods. Even though the location errors increase as distance noise increase, the network configuration still converges to the almost true value. It means that the location errors of DNRL and R-DNRL algorithms are robust with the distance measurement noise in a tolerant convergence bound. For the practical noisy range localization problem in a large-scale network, because R-DNRL method can accelerate the convergence speed, it could achieve great performance to balance localization convergence and computational effort.

C. Experiment 3

In third experiment we study the performance of R-DNRL localization algorithm when the network becomes large-scale. We use graphs with $n = 25, 100, 400$ nodes disposed in lattice configuration on the unit square $[0, 1]\times[0, 1]$. An example with $n = 100$ nodes is shown in Fig. 5. Four anchor nodes are selected at the external vertices of the unit square. Diagonal edges are added to the lattice structure for guaranteeing rigidity of the underlying graph, hence enabling the nodes to retrieve the correct configuration.

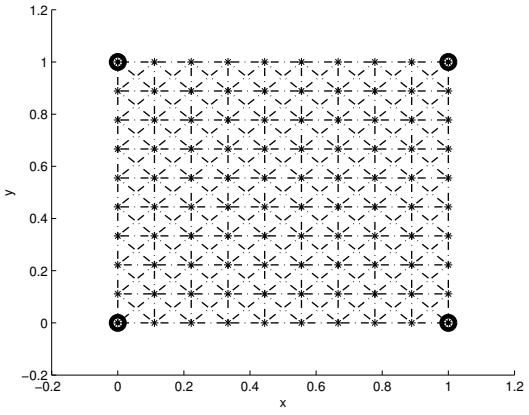

Fig. 5. Lattice configuration ($n = 100$).

In this experiment, we test the convergence of the R-DNRL method against different distance noise factors $n_f = 0.005$, 0.01, 0.03, 0.05, 0.07, 0.1 and different network sizes $n = 25, 100, 400$. To keep the distance errors vary in the same scale for different network sizes, the noise factor n_f is different

TABLE II
COMPARISON OF DNRL AND R-DNRL APPROACHES

Distance errors σ_d		0.01	0.03	0.05	0.07	0.1
DNRL	Location errors	0.0003	0.0002	0.0003	0.0004	0.00048
	Computational efforts (seconds)	36.9177	23.0370	21.9075	23.4849	27.0800
R-DNRL	Location errors	0.0001	0.0003	0.0006	0.0007	0.0009
	Computational efforts (seconds)	1.4018	1.2657	1.4550	1.2529	1.4289

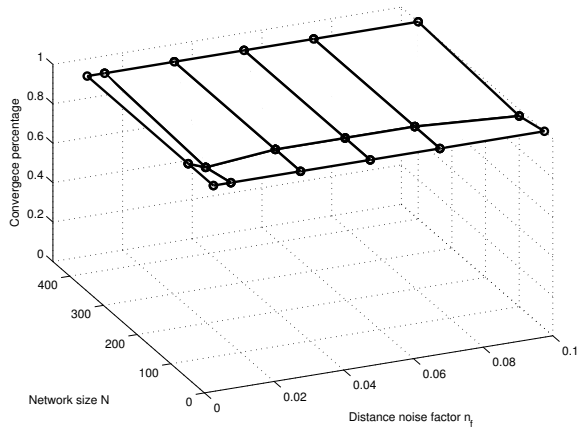

Fig. 6. Percentage of convergence test depending on network size and goodness of initial guess for the R-DNRL approach.

from the standard deviation of distance measurement σ_d. n_f is defined in eq.(39),

$$\hat{d}_{ij} = d_{ij}(1 + N(0,1) * n_f), (i,j) \in \mathcal{E}. \tag{39}$$

The initial guess parameter is fixed as $\sigma_p = 0.1$. Fig. 6 reports that through 100 Monte Carlo simulations, the convergence percentages are all greater than 96% in different network size and distance noise factor setups. Table III illuminates that with the distance noise robustness and tractable localization time, localization convergence of R-DNRL algorithm is not affected by the network size. We can conclude that R-DNRL algorithm is a feasible and efficient distributed localization algorithm in networks with different configurations, different network sizes and different distance measurement noises.

VII. CONCLUSIONS

In this paper, we have discussed two issues in noisy range-only network localization. First one is analysis of distance measurement noise effect on location estimation errors. We applied two different mathematical ways to derive the analytical location estimation errors. The formal position estimation error is lower bounded by Cramér-Rao bound (CRB). If the distance noise is typically small and the communication graph is generally globally rigid, an approximate location error computed by cost function linearization is proved to be equal to CRB error. That is an evidence that CRB error can be used as a benchmark to evaluate the performance of our particular localization algorithm. After understand of range measurement noise propagation, based on multidimensional scaling theory and Jacobian over relaxation algorithm, we deduced a distributed multidimensional scaling localization (DMDS) algorithm which is prove to achieve perfect convergence through comparing with Cramér-Rao bound (CRB). After that, we adopted a distance estimation scheme and a time varying relaxation parameter to obtain a distributed noisy range localization (DNRL) algorithm which is used in noisy case. The DNRL algorithm has good convergence and robustness in noisy range network localization problems. But it needs large computational effort, especially in large-scale networks. By adding a distributed gradient method to the DNRL algorithm, we derived a refined distributed noisy range localization (R-DNRL) algorithm which is robust to the distance measurement noise and achieves a good trade-off between computational effort and convergence error in practical applications. To sum up, in a localization problem, if the network size is small, we can choose the DNRL algorithm, nevertheless if the network size becomes large, the R-DNRL method would be a better choice.

The localization algorithms that we considered in this paper is applied in a stationary network, while the mobility of nodes would make the localization problem much complicated. Thus the distributed localization algorithm in a mobile network will be our future direction. Mobility creates the problem for locating and tracking the moving sensors in real time, and the opportunity to improve sensor location, such as considering the update rate, i.e. how often location must be re-estimated. If sensors are completely stationary, localization might just be done at startup. In networks with some changes over time, it should not be necessary to continually make and broadcast pair-wise measurements between stationary sensors. There might be a distributed algorithm which detects sensor motion (by monitoring changing pair-wise measurements) and then updates the location estimates only those sensors in motion, in order to save communication and computation. The trade-offs between measurement requirement, communication and accuracy should be more explicitly explored.

REFERENCES

[1] B. D. O. Anderson, I. Shames, G. Mao, and B. Fidan. Formal theory of noisy sensor network localization. *SIAM Journal of Discrete Mathematics*, 24(2):684–698, 2010.
[2] B.D.O. Anderson, P.N. Belhumeur, T. Eren, D.K. Goldenberg, A.S. Morse, W. Whiteley, and Y.R. Yang. Graphical properties of easily localizable sensor networks. *Wireless Networks*, 15(2):177–191, 2009.
[3] J. Bachrach and C. Taylor. Localization in sensor networks. *Handbook of sensor networks*, pages 277–310, 2005.
[4] D.P. Bertsekas and J.N. Tsitsiklis. *Parallel and distributed computation*. Old Tappan, NJ (USA); Prentice Hall Inc., 1989.
[5] P. Biswas, T.C. Liang, K.C. Toh, Y. Ye, and T.C. Wang. Semidefinite programming approaches for sensor network localization with noisy distance measurements. *IEEE Transactions on Automation Science and Engineering*, 3(4):360–371, 2006.
[6] I. Borg and P. Groenen. Modern multidimensional scaling: theory and applications. *Journal of Educational Measurement*, 40(3):277–280, 2003.
[7] G.C. Calafiore, L. Carlone, and M. Wei. A distributed gradient method for localization of formations using relative range measurements. In *Computer-Aided Control System Design (CACSD), 2010 IEEE International Symposium*, pages 1146–1151, 2010.

TABLE III
COMPUTATIONAL EFFORTS OF R-DNRL ALGORITHM (SECONDS)

Network size	$n_f = 0.01$	$n_f = 0.03$	$n_f = 0.05$	$n_f = 0.07$	$n_f = 0.1$
N=25	0.4871	0.4887	0.4921	0.4897	0.5223
N=100	1.8292	1.5619	1.6788	1.5881	0.8971
N=400	2.0228	1.9952	1.8818	2.8147	1.9734

[8] G.C. Calafiore, L. Carlone, and M. Wei. A distributed technique for localization of agent formations from relative range measurements. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 42(5):1065–1076, 2012.
[9] J. Cortés. Global and robust formation-shape stabilization of relative sensing networks. *Automatica*, 45(12):2754–2762, 2009.
[10] M. Deghat, I. Shames, B.D.O. Anderson, and J.M.F. Moura. Distributed localization via barycentric coordinates: Finite-time convergence. In *Proceedings of the 18th World Congress of the International Federation of Automatic Control (IFAC)*, 2011.
[11] T. Eren, D.K. Goldenberg, W. Whiteley, Y.R. Yang, A.S. Morse, B.D.O. Anderson, and P.N. Belhumeur. Rigidity, computation, and randomization in network localization. In *IEEE INFOCOM*, volume 4, pages 2673–2684, 2004.
[12] X. Ji and H. Zha. Sensor positioning in wireless ad-hoc sensor networks using multidimensional scaling. In *Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 4, pages 2652–2661, 2004.
[13] U.A. Khan, S. Kar, and J.M.F. Moura. Distributed sensor localization in random environments using minimal number of anchor nodes. *IEEE Transactions on Signal Processing*, 57(5):2000–2016, 2009.
[14] U.A. Khan, S. Kar, and J.M.F. Moura. Diland: An algorithm for distributed sensor localization with noisy distance measurements. *IEEE Transactions on Signal Processing*, 58(3):1940–1947, 2010.
[15] G. Mao and B. Fidanand B. Anderson. Wireless sensor network localization techniques. *Computer Networks*, 51(10):2529–2553, 2007.
[16] D. Moore, J. Leonard, D. Rus, and S. Teller. Robust distributed network localization with noisy range measurements. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 50–61, 2004.
[17] N. Patwari. Location estimation in sensor networks. *Ph.D. Thesis*, Citeseer, 2007.
[18] N. Patwari, J.N. Ash, S. Kyperountas, A.O. Hero III, R.L. Moses, and N.S. Correal. Locating the nodes: cooperative localization in wireless sensor networks. *Signal Processing Magazine*, 22(4):54–69, 2005.
[19] N. Patwari and A.O. Hero. Location estimation accuracy in wireless sensor networks. In *Conference Record of the Thirty-Sixth Asilomar Conference on Signals, Systems and Computers*, volume 2, pages 1523–1527, 2002.
[20] N. Patwari, A.O. Hero III, M. Perkins, N.S. Correal, and R.J. O'dea. Relative location estimation in wireless sensor networks. *IEEE Transactions on Signal Processing*, 51(8):2137–2148, 2003.
[21] A. Savvides, W. Garber, S. Adlakha, R. Moses, and M. Srivastava. On the error characteristics of multihop node localization in ad-hoc sensor networks. In *Information Processing in Sensor Networks*, pages 317–332. Springer, 2003.
[22] A. Savvides, W.L. Garber, R.L. Moses, and M.B. Srivastava. An analysis of error inducing parameters in multihop sensor node localization. *IEEE Transactions on Mobile Computing*, pages 567–577, 2005.
[23] A. Savvides, H. Park, and M.B. Srivastava. The bits and flops of the n-hop multilateration primitive for node localization problems. In *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pages 112–121. ACM, 2002.
[24] I. Shames and A.N. Bishop. Noisy network localization via optimal measurement refinement part 2: Distance-only network localization. In *Proceedings of the 18th World Congress of the International Federation of Automatic Control (IFAC)*, 2011.
[25] I. Shames, B.I. Fidan, and B. Anderson. Minimization of the effect of noisy measurements on localization of multi-agent autonomous formations. *Automatica*, 45(4):1058–1065, 2009.