A Barycentric Coordinate based Distributed Localization Algorithm for Sensor Networks

Yingfei Diao, *Member, IEEE* Zhiyun Lin, *Senior Member, IEEE* and Minyue Fu, *Fellow, IEEE*

***Abstract*—This paper studies the problem of determining the sensor locations in a large sensor network using only relative distance (range) measurements. Based on a generalized barycentric coordinate representation, our work generalizes the DILOC algorithm to the localization problem under arbitrary deployments of sensor nodes and anchor nodes. First, a criterion and algorithm are developed to determine a generalized barycentric coordinate of a node with respect to its neighboring nodes, which do not require the node to be inside the convex hull of its neighbors. Next, for the localization problem based on the generalized barycentric coordinate representation, a necessary and sufficient condition for the localizability of a sensor network with a generic configuration is obtained. Finally, a new linear iterative algorithm is proposed to ensure distributed implementation as well as global convergence to the true coordinates.**

***Index Terms*—sensor network, localization, distributed algorithm.**

I. Introduction

LOCATION based services are major applications of sensor networks and they require to solve the localization problem. The localization problem consists of two steps: acquiring measurements and transforming them to coordinate information. In this paper, we consider the situation of using relative distance measurements only and focus on the second step, i.e., how to estimate the locations of sensor nodes in a distributed manner.

Existing work on localization can be divided into two classes [1]: sequential methods and concurrent methods. A sequential method begins with a set of anchor nodes and computes the locations of other nodes one by one or group by group. A prominent example is the so-called *trilateration* method [2]. Its advantage is that it is easy to implement, but it requires each (location-unknown) node to have three connections with (location-known) anchor nodes, which is a sufficient but not necessary condition for localizability. That is, the trilateration method can only work on *trilateration networks* [3]. A concurrent method starts with an initial estimate for the coordinate of every sensor node. Each node then updates its coordinate in a distributed or cooperative manner using relative distance measurements with its neighbors and the estimates of the neighbors' coordinates. The iterative process terminates when the estimates converge, hopefully to the true coordinates.

There are many localization algorithms proposed from the perspective of optimization, based on techniques of convex optimization [4], multi-dimensional scaling (MDS) [5], semi-definite programming (SDP) [6] [7], gradient methods [8]–[11] etc. With imprecise range measurements, [12] uses the Cayley-Menger determinant to describe the distance constraints among pairs of sensor nodes, and then computes the positions by solving a constrained optimization problem. However, due the non-convex nature of the resulting optimization problem, these algorithms are commonly sensitive to initial guess of the true coordinates and are not able to guarantee global convergence. Moreover, most of them can not be implemented in a distributed manner.

Recently, several distributed and concurrent methods are developed to address the localization problem. In [13], Khan *et al.* provide a distributed iterative localization (DILOC) method based on the barycentric coordinate representation for sensor locations. The distinct feature of this method is that the sensor locations can be expressed as a pseudo linear system, with all the nonlinearities hidden in the measured distances. Two important characteristics of DILOC are 1) all sensor nodes are assumed to lie inside the convex hull of the anchor nodes and 2) each node, other than the anchor nodes, can find three neighbors such that it lies in the convex hull of these neighbors. With these characteristics, the system matrix of the linear system can be expressed as a *substochastic matrix* whose nonzero entries are all positive with every row sum less than one. Thus, global convergence is ensured and the sensor locations can be obtained by the iterative algorithm. Since every sensor node selects only three neighbors in [13], the convex hull of its neighbors is a triangle in the plane. The DILOC algorithm is generalized in [14] by allowing to have more than three neighbors for each sensor and it is shown similarly that the algorithm also works if every sensor node is contained in the convex hull of its neighbors. On the other hand, [15] and [16] consider relative position measurements, including both distances and bearing angles, and develop a localization algorithm that first estimates the centroid position of all nodes and then obtains the position of each node relative to the estimated centroid. Moreover, in [17], Singer shows that the eigenvectors of a specific weight matrix, which represents the communication graph, exactly match the sensor locations. To construct such a weight matrix, all locally rigid subnetworks are detected by using the techniques of MDS or SDP, and then stitched together according to a local embedding rule.

Y. Diao is with the School of Electrical and Electronic Engineering, Nanyang Technological University, 639798, Singapore.

Z. Lin is with the State Key Laboratory of Industrial Control Technology, College of Electrical Engineering, Zhejiang University, Hangzhou, Zhejiang, 310007 China.

M. Fu is with the School of Electrical Engineering and Computer Science, University of Newcastle, Callaghan, NSW, 2308 Australia. He also holds a Qian-ren professorship at Zhejiang University, China. e-mail: minyue.fu@newcastle.edu.au.

Manuscript received xxxx; revised xxxx.

In this paper, we consider only range measurements between pairs of sensor nodes and convert the range measurements to a barycentric coordinate for every sensor node. Different from [13], we do not require each sensor to lie inside the convex hull of its neighbors. This is motivated by a more practical scenario that for a large sensor network, the anchor nodes may not sit on the boundaries of the network. Our main idea is to employ a general form of barycentric coordinate representation which allows the coordinate of each node to be expressed as a pseudo linear function of the coordinates of its neighbors no matter whether or not it lies inside the convex hull formed by these neighbors. A criterion is developed, by which the pseudo linear function is determined with only the knowledge of relative distances. The implication of this result is that all the sensor locations can be expressed as a pseudo linear system, just like in DILOC. However, unlike DILOC, this new linear system may have unstable eigenvalues, which creates a major difficulty for distributed and iterative calculation. To overcome this difficulty, we provide a new distributed and iterative algorithm. It is shown that the algorithm ensures global convergence with a proper scaling parameter, which can be pre-computed in a distributed manner. In addition to the iterative localization algorithm, we also derive a necessary and sufficient localizability condition for localization schemes based on the barycentric coordinate representation. That is, an entire sensor network is localizable if and only if every sensor node has at least three disjoint paths to the anchor nodes in the graph associated with the barycentric coordinate representation. Compared with our earlier conference paper [18], both the distributed localization algorithm and the necessary and sufficient localizability condition are new in this paper. Finally, a simulation result is provided to validate the effectiveness of our proposed algorithm. Also, a comparison with the well-known MDS based localization algorithm is given in the simulation, which shows the advantage of guaranteeing global convergence using our algorithm.

The rest of the paper is organized as follows. We introduce the preliminaries of barycentric coordinate systems and graphs as well as the problem formulation in Section II. In Section III, we study the generalized barycentric coordinate representation and present a criterion on how to determine the sign pattern of the barycentric coordinate using relative distance measurements. In Section IV, we provide a necessary and sufficient condition for localizability using barycentric coordinate representations. A distributed and iterative localization algorithm is given in Section V. We present a simulation result in Section VI and conclude our work in Section VII.

Notation. $\mathbb{R}$ denotes the set of real numbers. $\mathbf{1}_n$ represents the n-dimensional vector of ones and I_n denotes the identity matrix of order n. A bold font letter represents a vector and a capital letter represents a matrix. Δijk denotes a triangle formed by nodes i, j and k. ι is defined to be $\sqrt{-1}$, the imaginary unit.

II. Preliminaries and Problem Formulation

A. Barycentric Coordinate Representation

The barycentric coordinate, which was introduced by August Ferdinand Möbius in 1827 [19], is a geometric notion characterizing the relative position of a point with respect to other points. For a point, say l with its Euclidean coordinate p_l, and other three points, say i, j and k with their Euclidean coordinates p_i, p_j and p_k in the plane, the barycentric coordinate of point l with respect to i, j and k is $\{a_{li}, a_{lj}, a_{lk}\}$ that satisfies

$$p_l = a_{li}p_i + a_{lj}p_j + a_{lk}p_k. \tag{1}$$

Especially, when $a_{li}+a_{lj}+a_{lk} = 1$, the barycentric coordinate is called the *areal* coordinate because it can be expressed as a ratio of signed areas between specified triangles. For the example shown in Fig. 1(a), the barycentric coordinate $\{a_{li}, a_{lj}, a_{lk}\}$ is given by

$$\begin{cases} a_{li} = \frac{S_{\Delta ljk}}{S_{\Delta ijk}} \\ a_{lj} = \frac{S_{\Delta lki}}{S_{\Delta ijk}} \\ a_{lk} = \frac{S_{\Delta lij}}{S_{\Delta ijk}} \end{cases}, \tag{2}$$

where $S_{\Delta ljk}$, $S_{\Delta lki}$, $S_{\Delta lij}$ and $S_{\Delta ijk}$ are the signed areas of the corresponding triangles Δljk, Δlki, Δlij and Δijk. These areas can be calculated with pairwise internode distance measurements through *Cayley-Menger determinant* [20] [21]. That is,

$$S^2_{\Delta ljk} = -\frac{1}{16}\begin{vmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & d^2_{lj} & d^2_{lk} \\ 1 & d^2_{jl} & 0 & d^2_{jk} \\ 1 & d^2_{kl} & d^2_{kj} & 0 \end{vmatrix}, \tag{3}$$

where d_{lj}, d_{lk} and d_{jk} are the distance measurements among node l, j and k, respectively. The sign of $S_{\Delta ljk}$ is positive if node l is on the left-hand side when one moves from node j to k, and negative otherwise.

In the plane, barycentric coordinates $(a_{l1}, ..., a_{ln})$ that are defined with respect to more than three points are called *generalized barycentric coordinates*. That is,

$$p_l = a_{l1}p_1 + a_{l2}p_2 + \cdots + a_{ln}p_n,$$

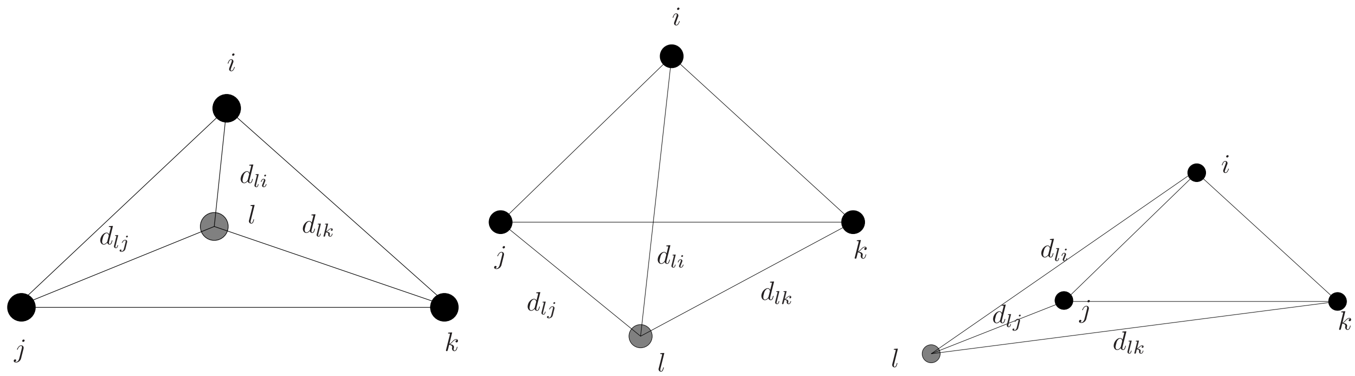
where p_l, p_1, p_2, and p_n are the Euclidean coordinates of these points.

B. Preliminaries on graphs and frameworks

An undirected graph (*graph* for short) $\mathcal{G} = (V, E)$ consists of a non-empty finite set V of elements called nodes and a finite set E of unordered pairs of nodes called edges. In graph theory, a sequence of edges, which connect a sequence of distinct nodes, is called a *path*. For a subset $U \subset V$ and a node $v \notin U$, we say v has k *disjoint paths* from U if there are k paths from nodes in U to v, which do not have a common node except v.

Next, we introduce several basic concepts in graph theory, which can be further explored in [22], [2], [23] and [24].

A *configuration* in the plane is a finite collection of n labeled points, $p = [p_1, \cdots, p_n]^T$, where $p_i \in \mathbb{R}^{1\times 2}$, $\forall i \in \{1, \cdots, n\}$. For convenience, we often represent each p_i by a complex value. A configuration is said to be *generic* if the coordinates $p_1, \ldots, p_n$ of the configuration do not satisfy any nontrivial algebraic equation with integer coefficients. A *framework* $(\mathcal{G}, \rho)$ in $\mathbb{R}^{1\times 2}$ is a graph $\mathcal{G} = (V, E)$ together with



(a) Node l lying inside the convex hull. (b) Case 1 of l lying outside the convex hull. (c) Case 2 of l lying outside the convex hull.

Fig. 1. Configurations of node l w.r.t. the convex hull of nodes i, j, k.

a configuration map $\rho : V \to \mathbb{R}^{1\times 2}$. Two frameworks ($\mathcal{G}$, ρ) and ($\mathcal{G}$, ζ) are said to be *congruent* if $\|\rho(i)-\rho(j)\| = \|\zeta(i)-\zeta(j)\|$ holds for all pairs $(i, j) \in E$. Two frameworks ($\mathcal{G}$, ρ) and ($\mathcal{G}$, ζ) are said to be *equivalent* if $\|\rho(i)-\rho(j)\| = \|\zeta(i)-\zeta(j)\|$ holds for all pairs $i, j \in V$. A framework ($\mathcal{G}$, ρ) is said to be *globally rigid* if every framework which is congruent to ($\mathcal{G}$, ρ) is equivalent to ($\mathcal{G}$, ρ). By abuse of the notion, we say $\mathcal{G}$ is globally rigid if the framework ($\mathcal{G}$, ρ) is globally rigid for any generic configuration $\rho(V)$.

C. Problem Formulation

The objective of the paper is to develop an efficient scheme for computing the coordinates of all sensor nodes in a network given a sufficient number of distance measurements between nodes and at least three known coordinates of nodes called *anchors*.

The commonly used trilateration scheme [2] sequentially computes the coordinate of a node based on at least three distance measurements to the nodes with known coordinates, that is, to solve a set of equations like

$$\begin{cases} d_{li} = \|p_l - p_i\| \\ d_{lj} = \|p_l - p_j\| \\ d_{lk} = \|p_l - p_k\| \end{cases}, \tag{4}$$

where p_u, $u \in \{i, j, k, l\}$, is the Euclidean coordinate of node u, and d_{uv}, $u, v \in \{i, j, k, l\}$, is the distance measurement between node u and v.

Instead of solving these nonlinear equations in a sequential way, Khan *et al.* developed a concurrent iterative algorithm [13], named $DILOC$, to compute the coordinates of all nodes in a network based on the barycentric coordinate presentation.

For a network of n nodes, without loss of generality, we assume there are three anchor nodes, whose coordinates are denoted by $p_1, p_2, p_3 \in \mathbb{R}^{1\times 2}$. Denote the coordinates of other nodes by $p_4, \ldots, p_n \in \mathbb{R}^{1\times 2}$. If we have a barycentric representation for every node (except the anchors) with respect to some other nodes, then we have the following pseudo linear representation for the whole network:

$$\begin{bmatrix} p_a \\ p_s \end{bmatrix} = \left[\begin{array}{c|c} I_3 & 0 \\ \hline B & C \end{array}\right] \begin{bmatrix} p_a \\ p_s \end{bmatrix}, \tag{5}$$

where $p_a = \begin{bmatrix} p_1 & p_2 & p_3 \end{bmatrix}^T$, $p_s = \begin{bmatrix} p_4 & \cdots & p_n \end{bmatrix}^T$, and the nonzero entries in each row of $\begin{bmatrix} B & C \end{bmatrix}$ are the barycentric coordinates of the node corresponding to the row. Equivalently, we can write as

$$p_s = Cp_s + Bp_a. \tag{6}$$

Then the DILOC algorithm [13] considers the following iterative form to compute the coordinates p_s.

$$\hat{p}_s(t+1) = C\hat{p}_s(t) + Bp_a, \tag{7}$$

where $\hat{p}_s(t)$ represents the estimate of the coordinates at step t. In order to guarantee that the iteration (7) is convergent, it is assumed in the DILOC algorithm [13] that every node should lie inside a triangle of other three nodes and all the sensor nodes to be localized should lie inside the triangle formed by the three anchor nodes. However, for practical sensor networks, these two assumptions are restrictive, especially when the communication range is limited and when the neighbors of each node cannot be arbitrarily arranged.

In this paper, we will give up the aforementioned assumptions and devise a new distributed algorithm called Extended Computation scHeme of cOordinate (ECHO). However, there are several challenges. Firstly, ECHO considers any possible configuration, in contrast to the case of each node lying inside a triangle of three other nodes. But for the case where a node lies outside of a triangle of three other nodes, the barycentric coordinate of this node is difficult to obtain because no known solution is available to determine the signs of the triangle area ratios used to compute the barycentric coordinate as in (2), by using only the local distance measurement information. Moreover, if there are sufficient distance measurement information such that a node can be represented using a generalized barycentric coordinate with respect to more than three other nodes, we should make full use of it without just selecting three from them. Secondly, due to the general barycentric coordinate representation, the matrix C in (7) may have eigenvalues with the modulus greater than 1 and thus the iteration (7) may diverge. One example configuration with unstable matrix C is shown in Fig. 2, where nodes $1, 2, 3$ are the anchor nodes, and nodes $4, 5$ are position-unknown nodes. Their Euclidean coordinates are $\iota, -4, 4, -2+\iota 2, 2+\iota 2$, respectively. The barycentric coordinate of node 4 with respect

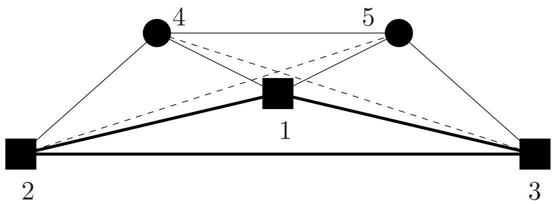


Fig. 2. A configuration of barycentric coordinate with unstable C

to nodes 1, 3 and 5 is $\{2, -4, 3\}$. Similarly, it can be obtained that the barycentric coordinate of node 5 with respect to nodes 2, 3 and 4 is $\{2, -4, 3\}$. Thus, we have

$$\begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \end{bmatrix} = \left[\begin{array}{ccc|cc} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ \hline 2 & 0 & -4 & 0 & 3 \\ 0 & 2 & -4 & 3 & 0 \end{array}\right] \begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \end{bmatrix}, \tag{8}$$

for which

$$C = \begin{bmatrix} 0 & 3 \\ 3 & 0 \end{bmatrix},$$

has spectral radius greater than 1.

Therefore, this paper aims at 1) developing an effective algorithm to compute a general barycentric coordinate of every node with respect to possibly more than three other nodes based on only local distance measurement information, and 2) developing a new distributed localization algorithm, based on the barycentric coordinate, that ensure global convergence. Moreover, we are going to explore the localizability condition for the whole network by using the distributed localization scheme based on a general barycentric coordinate representation.

III. Algorithms for Computing Generalized Barycentric Coordinates

As we discussed in the preceding section, in applying barycentric coordinates to solve distance-based localization problems, a major technical difficulty is the computation of the generalized barycentric coordinates given only local distance measurement information, in particular when a node may not lie inside of the triangle formed by its neighbors or when a node may have more than three neighbors. Two examples are shown in Fig. 1(b) and 1(c), for which a node l lies outside the triangle Δ_{ijk} of its three neighbors. This section aims at providing an effective scheme to compute generalized barycentric coordinates for all these cases, but starts with computing the signed barycentric coordinates on triangles.

A. Signed barycentric coordinates on triangles

Consider a node l together with three neighbors i, j and k. Suppose the pairwise distance measurements between any two in $\{l, i, j, k\}$ are available to node l. Then the absolute values of the barycentric coordinates a_{li}, a_{lj} and a_{lk} can be computed as

$$\begin{cases} |a_{li}| = \frac{|S_{\Delta ljk}|}{|S_{\Delta ijk}|} \\ |a_{lj}| = \frac{|S_{\Delta lki}|}{|S_{\Delta ijk}|} \\ |a_{lk}| = \frac{|S_{\Delta lij}|}{|S_{\Delta ijk}|} \end{cases}, \tag{9}$$

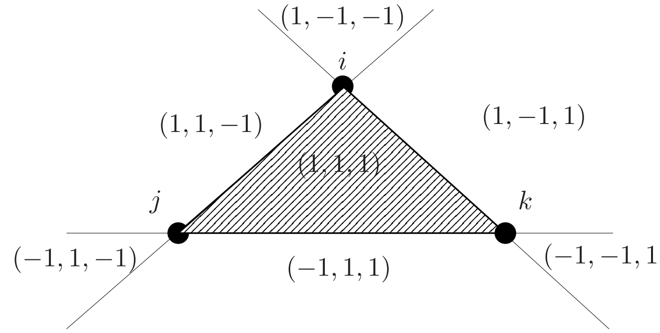


Fig. 3. Seven possible sign patterns for $(\sigma_{li}, \sigma_{lj}, \sigma_{lk})$.

where $|S_{\Delta ljk}|$ and others can be solved according to (3). Thus, it remains to determine the signs of the barycentric coordinates to compute the barycentric coordinates. Moreover, note that $a_{li} + a_{lj} + a_{lk} = 1$. Thus, given $|a_{li}|, |a_{lj}|$ and $|a_{lk}|$, the problem of determining the sign pattern of the barycentric coordinate is equivalent to solve the following equation:

$$\sigma_{li}|a_{li}| + \sigma_{lj}|a_{lj}| + \sigma_{lk}|a_{lk}| = 1, \tag{10}$$

where σ_{li}, σ_{lj} and σ_{lk} take values of either 1 or -1. The information available for determining the signs should be limited to the pairwise distance measurements between nodes within the set $\{i, j, k, l\}$.

To avoid the case that $S_{\Delta ijk} = 0$, we need an assumption on the positions of the node ls neighbors as below.

Assumption 1: For each node in the network, any three of its neighbors are not collinear.

There are only 7 possible sign patterns $(\sigma_{li}, \sigma_{lj}, \sigma_{lk})$, as shown in Fig. 3 (because the pattern $(-1, -1, -1)$ is not a possible solution for (10)). It turns out that sometimes the sign pattern can be uniquely determined from (10), but sometimes this cannot be done. When the sign pattern cannot be uniquely solved from (10), we consider two cases in the following.

In the first case, one of $|a_{li}|, |a_{lj}|, |a_{lk}|$ equals to zero. That is, a node l lies on the line aligned with one of three edges of the triangle formed by its three neighbors, according to (9). Without loss of generality, say $a_{li} = 0$. For this case, we can take $\sigma_{li} = 1$ (without loss of generality). Then the other two signs σ_{lj} and σ_{lk} can be determined according to the following criterion.

$$(\sigma_{li}, \sigma_{lj}, \sigma_{lk}) = \begin{cases} (1,1,1) & \text{if } |a_{lj}|, |a_{lk}| \le 1, \\ (1,1,-1) & \text{if } |a_{lj}| > 1, |a_{lj}| > |a_{lk}|, \\ (1,-1,1) & \text{if } |a_{lk}| > 1, |a_{lk}| > |a_{lj}|. \end{cases} \tag{11}$$

In the following lemma, we characterize the remaining cases where the sign pattern can not be uniquely determined from (10).

Lemma 1: Suppose$|a_{li}| \neq 0$, $|a_{lj}| \neq 0$, and $|a_{lk}| \neq 0$. The solution of (10) does not result in a unique sign pattern $(\sigma_{li}, \sigma_{lj}, \sigma_{lk})$ if and only if one of them, saying a_{li}, satisfies $|a_{li}| = 1$, and $|a_{lj}| = |a_{lk}|$.

Proof: (Sufficiency) If $|a_{li}| = 1$ and $|a_{lj}| = |a_{lk}|$, it can be inferred from (10) that $\{\sigma_{li}, \sigma_{lj}, \sigma_{lk}\} = \{1, 1, -1\}$ or $\{\sigma_{li}, \sigma_{lj}, \sigma_{lk}\} = \{1, -1, 1\}$. That is, (10) does not result in a unique sign pattern. Fig. 5 shows such an example where

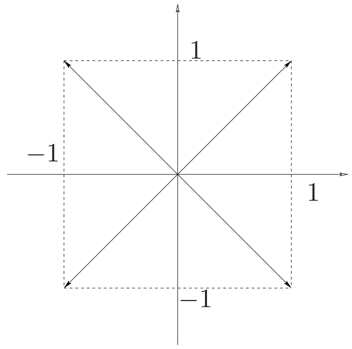


Fig. 4. An illustration for the necessity proof.

node l lies on the line of e_1 parallel to the line of nodes j and k. The two possible positions for l are $l^{'}$ and $l^{''}$ on e_1.

(Necessity) Suppose there are two sign patterns both satisfying (10). That is, it holds that

$$\begin{bmatrix}|a_{li}| & |a_{lj}| & |a_{lk}|\end{bmatrix}\begin{bmatrix}\mathbf{v}_1\\ \mathbf{v}_2\\ \mathbf{v}_3\end{bmatrix} = \begin{bmatrix}1 & 1\end{bmatrix}, \tag{12}$$

where $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3 \in \{[1\ 1], [-1\ 1], [-1\ \ -1], [1\ \ -1]\}$. This means a positive combination of $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ equals to $[1\ 1]$ (see Fig. 4). Consequently, there must be $[1\ 1]$ for one of $\mathbf{v}_1, \mathbf{v}_2$, and $\mathbf{v}_3$. Without loss of generality, we assume $\mathbf{v}_1 = [1\ 1]$. Next, we consider different choice of $\mathbf{v}_2$. If $\mathbf{v}_2$ equals to $[1, 1]$ or $[-1, -1]$, we will have $\mathbf{v}_3$ equal to $[-1\ \ -1]$ or $[1\ 1]$. In this way, two solutions $\{\sigma_{li}, \sigma_{lj}, \sigma_{lk}\}$ are actually identical. If $\mathbf{v}_2$ equals to $[-1\ 1]$ or $[1\ \ -1]$, $\mathbf{v}_3$ must equal to $[1\ \ -1]$ or $[-1\ 1]$. Then according to (10), we know that $|a_{li}| = 1$ and $|a_{lj}| = |a_{lk}|$. ∎

Next, we present a result on how to determine the sign pattern using the distance based information when it can not be uniquely solved from (10).

Lemma 2: Given $|a_{li}| = 1$ and $|a_{lj}| = |a_{lk}| \neq 0$, suppose $\angle ijk$ is an acute angle[1].

1) $\sigma_{li} = -1$ if and only if

$$d_{jl} = d_{ik},\ d_{kl} = d_{ij},\ \text{and}\ d_{il}^2 = 2d_{ij}^2 + 2d_{ik}^2 - d_{jk}^2.$$

2) If $\sigma_{li} = -1$, then $\{\sigma_{li}, \sigma_{lj}, \sigma_{lk}\} = \{-1, 1, 1\}$.
3) If $\sigma_{li} = 1$ and $d_{jl}^2 < d_{ij}^2 + d_{il}^2$, then $(\sigma_{li}, \sigma_{lj}, \sigma_{lk}) = \{1, 1, -1\}$.
4) If $\sigma_{li} = 1$ and $d_{jl}^2 > d_{ij}^2 + d_{il}^2$, then $(\sigma_{li}, \sigma_{lj}, \sigma_{lk}) = \{1, -1, 1\}$.

Proof: 1) (Necessity) If $\sigma_{li} = -1$, then we have $a_{li} = -1$. Moreover, since $|a_{lj}| = |a_{lk}|$, it follows from (10) that $a_{lj} = a_{lk} = 1$. Thus, $S_{\Delta ljk} = -S_{\Delta ijk}$, $S_{\Delta lij} = S_{\Delta ijk}$ and $S_{\Delta lki} = S_{\Delta ijk}$. Comparing the sign pattern with the ones described in Fig. 3, we know that the only option of l is at the location of $l^{'''}$ in Fig. 5, which forms a parallelogram together with nodes i, j and k. Hence, we can obtain directly that $d_{jl} = d_{ik}$, $d_{kl} = d_{ij}$. Furthermore, according to the parallelogram law, we have

$$d_{il}^2 = 2d_{ij}^2 + 2d_{ik}^2 - d_{jk}^2. \tag{13}$$

[1]If $\angle ijk$ is not acute, then $\angle ikj$ must be acute and the conditions in the lemma can be modified accordingly.

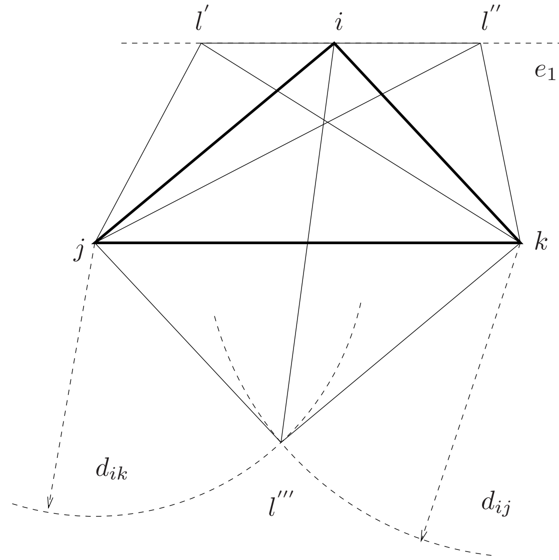


Fig. 5. An example of Δijk and node l for Lemmas 1 and 2.

(Sufficiency) If $d_{jl} = d_{ik}$ and $d_{kl} = d_{ij}$, we can draw two circles centered at j and k with radius d_{ik} and d_{ij}, respectively. These two circles will have two intersection points. One of the two intersection points is $l^{'''}$ and we denote the other by l^*. From the necessity proof, it is known that when node l is at the location of $l^{'''}$, it satisfies (13). On the other hand, we will show that when node l is at the location of l^*, it does not satisfy (13). (To see this, it remains to show that $d_{il'''} \neq d_{il^*}$. Suppose by contradiction that $d_{il'''} = d_{il^*}$. Then, recalling the fact $d_{jl'''} = d_{jl^*}$ and $d_{kl'''} = d_{kl^*}$, we have that nodes i, j and k are on the perpendicular bisector of the line segment $l^{'''}l^*$ and so they are colinear, a contradiction to Assumption 1. Therefore, it can be concluded that due to the condition $d_{il}^2 = 2d_{ij}^2 + 2d_{ik}^2 - d_{jk}^2$, node l must lie at the location of $l^{'''}$. Thus, according to the sign patterns described in Fig. 3, we can obtain that $\sigma_{li} = -1$.

2) If $\sigma_{li} = -1$, then $\{\sigma_{li}, \sigma_{lj}, \sigma_{lk}) = (-1, 1, 1)$, which is shown in the necessity proof of 1).

3) If $\sigma_{li} = 1$, then we have $a_{li} = 1$. For this case, $S_{\Delta ljk} = S_{\Delta ijk}$ according to (2). Therefore, node l must be on the line that is parallel to the edge jk and crosses node i. On this line, there are two nodes, saying node $l^{'}$ and $l^{''}$ as shown in Fig. 5, whose distances to node i are equal to d_{il}.

For the triangle $\Delta ijl^{'}$, according to the cosine law, it holds that

$$d_{jl'}^2 = d_{ij}^2 + d_{il'}^2 - 2d_{ij}d_{il'}\cos\angle jil^{'}.$$

Since $l^{'}l^{''}$ is parallel with jk, we have $\angle jil^{'} = \angle ijk$. Then we know $d_{jl'}^2 < d_{ij}^2 + d_{il'}^2$ because $\angle ijk$ is an acute angle. Similarly, for the triangle $\Delta ijl^{''}$, we could obtain that $d_{jl''}^2 > d_{ij}^2 + d_{il''}^2$.

Therefore, if $d_{jl}^2 < d_{ij}^2 + d_{il}^2$, then node l must be at the locatio of l'. Thus, according to the sign patterns described in Fig. 3, we obtain that $(\sigma_{li}, \sigma_{lj}, \sigma_{lk}) = (1, 1, -1)$.

4) Following the argument in 3), we know that if $d_{jl}^2 > d_{ij}^2 + d_{il}^2$, then node l must lie at the location of $l^{''}$. Then

again from the sign patterns described in Fig. 3, we obtain that $(\sigma_{li}, \sigma_{lj}, \sigma_{lk}) = (1, -1, 1)$. ■

Finally, we summarize the above results to provide an algorithm of determining the sign pattern based on the distance measurement information. The pseudo code is given in Algorithm 1.

Algorithm 1 Determining the sign pattern of node l's barycentric coordinate.

Input: $|a_{li}|$, $|a_{lj}|$, $|a_{lk}|$, d_{li}, d_{lj}, d_{lk}, d_{ij}, d_{ik}, d_{jk}.
Output: $(\sigma_{li}, \sigma_{lj}, \sigma_{lk})$.

Solve eq. (10).
if the solution is unique, **then**
    Return $\sigma_{li}, \sigma_{lj}, \sigma_{lk}$.
else if one of $|a_{li}|$, $|a_{lj}|$, $|a_{lk}|$ equals to 0, **then**
    Determine $(\sigma_{li}, \sigma_{lj}, \sigma_{lk})$ according to (11).
else if $d_{jl} = d_{ik}$, $d_{kl} = d_{ij}$ and $d_{il}^2 = 2d_{ij}^2 + 2d_{ik}^2 - d_{jk}^2$, **then**
    $(\sigma_{li}, \sigma_{lj}, \sigma_{lk}) = (-1, 1, 1)$.
else if $d_{jl}^2 < d_{ij}^2 + d_{il}^2$, **then**
    $(\sigma_{li}, \sigma_{lj}, \sigma_{lk}) = (1, 1, -1)$.
else if $d_{jl}^2 > d_{ij}^2 + d_{il}^2$, **then**
    $(\sigma_{li}, \sigma_{lj}, \sigma_{lk}) = (1, -1, 1)$.
end if

B. Generalized barycentric coordinates

For a sensor network of n nodes, we use an undirected graph $\mathcal{G} = (V, E)$ to model it with each node $i \in V$ corresponding to a sensor node and each edge $(i, j) \in E$ indicating that sensor node i and j are in the communication range and the distance between i and j is available to both nodes.

Recall a necessary localizable condition from [25] that every node in $\mathcal{G}$ should have at least three disjoint paths to the set of anchor nodes. This implies locally each node has at least three neighbors in $\mathcal{G}$. For any node $l \in V$, denote by $\mathcal{N}_l$ the set of all its neighbors in $\mathcal{G}$. We provide an algorithm below to compute a generalized barycentric coordinate of a node with respect to its neighbors for a given graph $\mathcal{G}$.

From the formula to compute the generalized barycentric coordinate in Algorithm 2, it can be simply verified that

$$p_l = \sum_{i \in \mathcal{N}_l} a_{li} p_i, \tag{14}$$

$$\sum_{i \in \mathcal{N}_l} a_{li} = 1. \tag{15}$$

However, it should be pointed out that some a_{li}, $i \in \mathcal{N}_l$, may be zero. An example is given in Fig. 6, where nodes $1, \dots, 4$ are all neighbors of l, but $a_{l4} = 0$ in this case since there is no combination of three neighbors containing node 4, which are mutually neighbors.

Suppose without loss of generality that the three anchor nodes in $\mathcal{G}$ are labeled as 1, 2, and 3. Then we can write the generalized barycentric coordinates of all nodes in a matrix form as

$$A = \left[\begin{array}{c|c} \triangle_{3\times 3} & * \\ \hline B & C \end{array} \right], \tag{16}$$

Algorithm 2 Computing node l's generalized barycentric coordinate.

Input: $\mathcal{N}_l$, the distances d_{li} for $i \in \mathcal{N}_l$, and the distances d_{ij} for $i, j \in \mathcal{N}_l$ and $(i, j) \in E$.
Output: a_{li}, $i \in \mathcal{N}_l$.

Set $m = 0$;
while there is still a combination of three neighbors in $\mathcal{N}_l$ that have not been selected before **do**
    Choose a combination of three neighbors from $\mathcal{N}_l$, say i, j, and k.
    if node i, j, and k are mutually neighbors **then**
        (1) Update $m := m + 1$;
        (2) Compute the barycentric coordinate $a_{li}^{(m)}$, $a_{lj}^{(m)}$, $a_{lk}^{(m)}$ of l with respect to i, j, and k according to the result in the preceding section;
        (3) Set $a_{ls}^{(m)} = 0$ for $s \in \mathcal{N}_l - \{i, j, k\}$.
    end if
end while
return $a_{li} = \frac{1}{m} \sum_{r=1}^{m} a_{li}^{(m)}$ for any $i \in \mathcal{N}_l$.

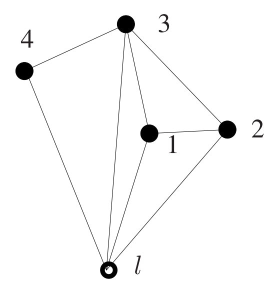


Fig. 6. An example that the derived generalized barycentric coordinate may have zero components with respect to its neighbors in $\mathcal{G}$.

where the off-diagonal entries at (i, j) are the generalized barycentric coordinate if $j \in \mathcal{N}_i$ and must be zero otherwise.

Now we define a new graph $\mathcal{G}_A$ associated to the matrix A such that there is an edge (i, j) if and only if the (i, j)th entry of A is nonzero. Note from the example in Fig. 6 that the new graph $\mathcal{G}_A$ describing the interconnection of nodes in terms of the generalized barycentric coordinates might be different from the graph $\mathcal{G}$ but certainly it is a subgraph of $\mathcal{G}$ with the same node set.

Remark 1: For graph $\mathcal{G}_A$, if a node has a neighbor, then it must have at least three neighbors because the barycentric coordinates are derived triangle by triangle as described in Algorithm 2. In other words, if j is a neighbor of l in $\mathcal{G}_A$, then in graph $\mathcal{G}$, j is also a neighbor of l and moreover there must exist another two neighbors of l, which are both neighbors of j, to form a triangle together with j (see for example, nodes 1, 2, and 3 in Fig. 6. Also, from the above observation, we can infer that if j is a neighbor of l in $\mathcal{G}_A$, then l is also a neighbor of j since in $\mathcal{G}$, j must have three neighbors (namely, node l, and node l's another two neighbors that form a triangle together with j). Therefore, $\mathcal{G}_A$ is also an undirected graph though the matrix A is not symmetric.

Remark 2: The generalized barycentric coordinates computed by Algorithm 2 are more general than in [13] and [14].

In [13], each node needs to be inside a simplex of three neighbors and, in [14], each node needs to be inside the polygon of its neighbors. But Algorithm 2 does not require the convex hull assumption.

IV. Localizability using barycentric coordinate representation

For the generalized barycentric coordinates given in (16), we have the following pseudo linear representation for the Euclidean coordinates of the sensor network:

$$p_s = Cp_s + Bp_a, \tag{17}$$

where p_a and p_s are the stacked coordinate vector of anchor nodes and sensor nodes respectively. From (17), it is certain that the non-singularity of the matrix $I - C$ implies the localizability of a sensor network using the barycentric coordinate representation. Thus, the localizability for the approach based on the barycentric coordinate representation relates to the topological connectivity of $\mathcal{G}_A$ rather than $\mathcal{G}$. In the following, we aim to provide the topological connectivity condition of $\mathcal{G}_A$ to ensure the solvability of the localization problem using this approach.

Before developing our main result for localizability, we give a preliminary result on graph Laplacian. Consider a simple graph $\mathcal{G} = (V, E)$ and let A denote a matrix of specific zero/nonzero pattern associated with $\mathcal{G}$. That is, the (i, j)th entry is nonzero if and only if (i, j) is an edge in $\mathcal{G}$. Define a diagonal matrix D with its diagonal entry to be the corresponding row sum of A. Then we denote by $\mathcal{L}(\mathcal{G})$ the set of all matrices $D - A$. It is certain that for any matrix $L \in \mathcal{L}(\mathcal{G})$, $L\mathbf{1} = 0$. Let R be a subset of V and let L_R be the sub-matrix of $L \in \mathcal{L}(\mathcal{G})$ with the rows and columns corresponding to nodes in R crossed out. The following lemma provides the relationship between nonzero principal minors of $L \in \mathcal{L}(\mathcal{G})$ and the topological connectedness of $\mathcal{G}$.

Lemma 3: Consider generic $\xi, \zeta, \in \mathbb{R}^n$. The following two are equivalent.

1) There are three disjoint paths from $R = \{r_1, r_2, r_3\}$ to every other node in $\mathcal{G}$.
2) For almost all $L \in \{L \in \mathcal{L}(\mathcal{G}) : L\xi = 0, L\zeta = 0\}$, all principal minors of L_R are distinct from zero.

The proof of this lemma is given in Appendix.

Now we are ready to present a necessary and sufficient localizability condition for localization schemes based on the barycentric coordinate representation.

Theorem 1: A sensor network with a generic configuration $[p_a^T\ p_s^T]$ is localizable using the barycentric coordinate representation by solving (17), i.e., the matrix $I - C$ is nonsingular, if and only if every node to be localized has at least three disjoint paths from the set of anchor nodes in $\mathcal{G}_A$.

Proof: Write

$$L = I - A,$$

where A is the matrix defined in (16) in terms of the generalized barycentric coordinates. From (15) we know that $L\mathbf{1} = 0$ with $\mathbf{1}$ being a vector of all one's. Moreover, let x and y be the real and imaginary part of the vector $[p_a^T\ p_s^T]^T$, respectively. Then from (14) we know that $Lx = 0$ and $Ly = 0$. So if every node to be localized has at least three disjoint paths from the set of anchor nodes in $\mathcal{G}_A$, then by Lemma 3 it is known that $\det(I - C) \neq 0$. Therefore, the sensor network is localizable using the barycentric coordinate representation by solving (17).

On the other hand, if the sensor network is localizable using the barycentric coordinate representation by solving (17), then $\det(I - C) \neq 0$. Thus, again by Lemma 3 it follows that $\mathcal{G}_A$ has three disjoint paths from the set of anchor nodes to any other node. ■

V. An extended computation scheme of coordinates

As discussed in Subsection II-C, the matrix C in (17) for the generalized barycentric coordinates may not be stable. So the DILOC algorithm may not converge. In the following, we present an alternative approach to compute the coordinates with ensured global convergence. An extended distributed computation scheme of coordinates, called ECHO, is developed.

Let $M = I - C$. Then eq. (17) can be re-organized as

$$Mp_s = Bp_a.$$

Multiplying the matrix ϵM^T (with a scalar $\epsilon > 0$ to be specified later) to both sides of the above equation, we then obtain that

$$\epsilon M^T M p_s = \epsilon M^T B p_a. \tag{18}$$

Thus, p_s can be solved from (18) by the Richardson iteration [26]

$$\hat{p}_s(t+1) = (I - \epsilon M^T M)\hat{p}_s(t) + \epsilon M^T B p_a. \tag{19}$$

Notice that $M^T M$ is positive definite. So for sufficiently small ϵ, the roots of $\epsilon M^T M$ come to the open unit disk centered at $(1, 0)$. More specifically, the eigenvalues of $M^T M$ are all real and thus the eigenvalues of $\epsilon M^T M$ are on the interval $(0, 2)$ for $0 < \epsilon < 2/\lambda_{\max}(M^T M)$, where $\lambda_{\max}(M^T M)$ represents the maximum eigenvalue of $M^T M$. Therefore, the system matrix in the iteration algorithm (19) is stable and (19) is convergent for arbitrary initial conditions. Moreover, in order to have the fastest convergence rate, ϵ should be selected as

$$\epsilon = \frac{2}{\lambda_{\max}(M^T M) + \lambda_{\min}(M^T M)}\ , \tag{20}$$

where $\lambda_{\min}(M^T M)$ is minimum eigenvalue of $M^T M$ [27, Section 4.2]. This value can be solved in a distributed way by distributed power iteration algorithms (see for example [28]) or other distributed techniques of estimating the eigenvalues, such as the methods in [29], [30] and [31].

Next we consider the distributed implementation of the iteration algorithm (19). Note that (19) can be written as a two-step iteration form by introducing an intermediate variable η:

$$\left\{ \begin{array}{l} \eta(2t+1) = M\hat{p}_s(2t) - B\hat{p}_s(2t) \\ \hat{p}_s(2t+2) = \hat{p}_s(2t) - \epsilon M^T \eta(2t+1), \end{array} \right. \quad t = 0, 1, 2, \cdots. \tag{21}$$

As discussed in Remark 1, though M is not symmetric, $\mathcal{G}_A$ however is an undirected graph, which means each row of

M^T has nonzero entries in the same locations as M. Thus, the nonzero entries of row i of M^T can also be available to node i from its neighbors $j \in \mathcal{N}_i^*$ where $\mathcal{N}_i^*$ represents the neighbor set of node i in the graph $\mathcal{G}_A$ rather than $\mathcal{G}$. Therefore, (21) can be formulated as a distributed algorithm (Algorithm 3) implemented on each node, for which communications are required only between neighbors.

Algorithm 3 ECHO

1: Each node i transmits its own estimate $\hat{p}_i$ to its neighbors in $\mathcal{N}_i^*$.
2: Each node i receives $\hat{p}_j$ from its neighbors in $\mathcal{N}_i^*$ and then computes $\eta_i = \hat{p}_i - \sum_{j\in\mathcal{N}_i^*} a_{ij}\hat{p}_j$.
3: Each node i transmits η_i to its neighbors in $\mathcal{N}_i^*$.
4: Each node i receives η_j from its neighbors in $\mathcal{N}_i^*$ and then updates $\hat{p}_i := \hat{p}_i - \epsilon \sum_{j\in\mathcal{N}_i^*} a_{ji}\eta_j$.
5: Go back to 1.

VI. Simulations

A. Localization by ECHO

We provide simulation results to show the effectiveness of ECHO. A network consisting of 8 nodes is considered, which is deployed in an 80×80 unit area, as shown in Fig. 7(a). The anchor nodes are labeled by 1, 2, and 3. The remaining nodes are the ones to be localized. The edges in Fig. 7(a) indicate that the distance information between the two nodes on both sides of an edge is available. This is denoted as $\mathcal{G}$. It can be checked from the example that the assumptions in [13] are not satisfied and thus the DILOC algorithm is not applicable. More specifically,

1) The triangle of the three anchor nodes does not contain all sensor nodes inside;
2) Some sensor node may not lie inside the convex hull of their neighbors.

Using Algorithm 2, we obtain generalized barycentric coordinates as follows:

$$p_s = \begin{bmatrix} B & C \end{bmatrix} \begin{bmatrix} p_a \\ p_s \end{bmatrix},$$

where

$$B = \begin{bmatrix} 0 & -0.9599 & 0 \\ 0.5292 & -0.7956 & 1.2663 \\ 0 & -1.7515 & 0.9345 \\ 0 & -0.0211 & 0 \\ 0.9886 & -0.4040 & 0 \end{bmatrix},$$

and

$$C = \begin{bmatrix} 0 & 0 & 0 & 1.3811 & 0.5788 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.8170 & 0 \\ 0.4868 & 0 & 0.5344 & 0 & 0 \\ 0.4154 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

The graph used to describe the interconnections between nodes in terms of the generalized barycentric coordinates is $\mathcal{G}_A$, which is shown in Fig. 7(b). Compared to $\mathcal{G}$, the graph $\mathcal{G}_A$ has less edges. However, it can be checked that $\mathcal{G}_A$ has three disjoint paths from the set of anchor nodes $\{1, 2, 3\}$ to every other node. So every node in the network is localizable by Theorem 1.

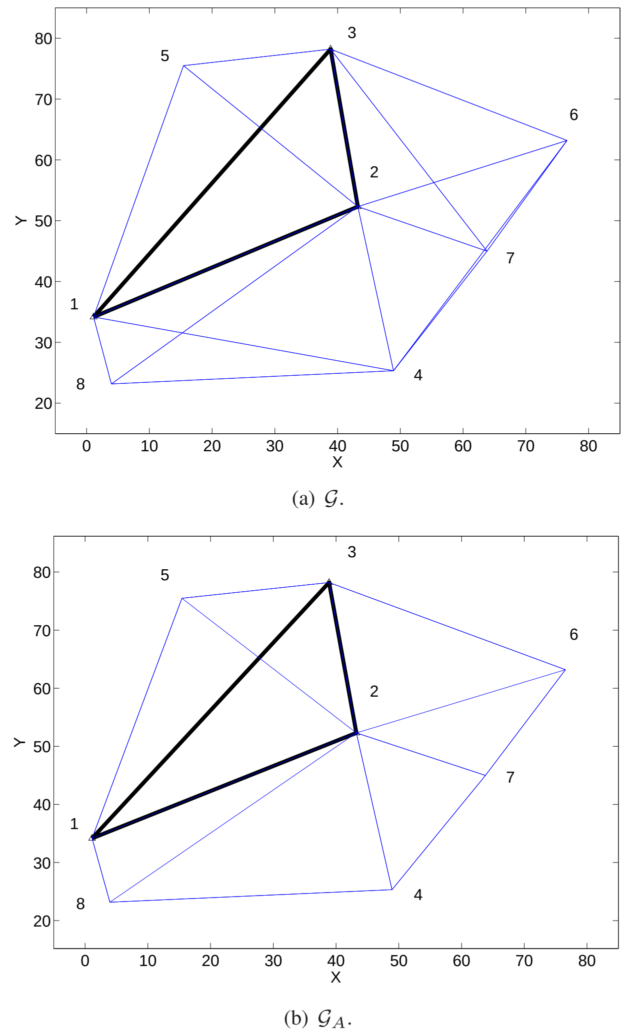
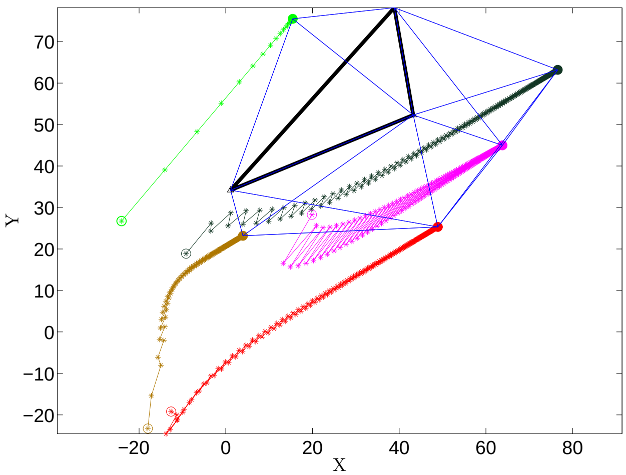


Fig. 7. A sensor network of 8 nodes. X and Y are horizontal and vertical coordinates.

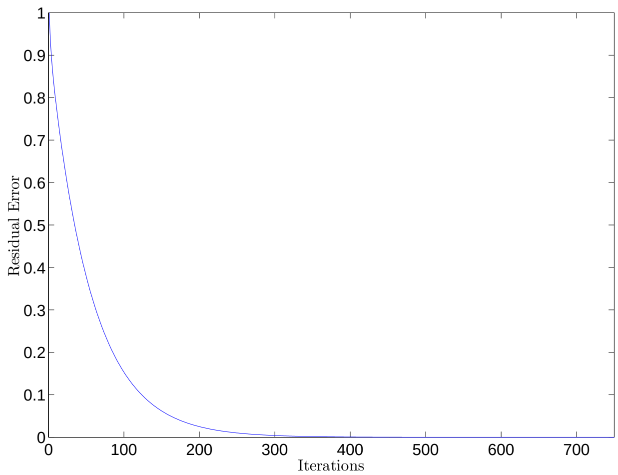
In this example, it can be computed that taking $\epsilon = 0.2527$ achieves the fastest convergence rate of (19). The trajectories of the coordinate computation using ECHO with $\epsilon = 0.2527$ are given in Fig. 8(a). The estimation error ratio $\frac{||p_s(t)-p_s||}{||p_s(0)-p_s||}$ is plotted in Fig. 8(b). From the simulation result, ECHO exponentially converges and leads to true coordinates of the sensor nodes to be localized.

B. Performance evaluation compared with MDS

To show the effectiveness of the ECHO algorithm, we compare with the well-known MDS method [5]. A sensor network of 80 nodes deployed in an area of 100×100 square meters is considered in the simulation. The configuration is plotted in Fig. 9(a), in which the three anchor nodes with known locations are represented by red cycles while the others are sensor nodes with unknown locations. The lines between pairs of nodes in the figure indicate that the distances are available for these pairs.



(a) Trajectories of the coordinate estimates.



(b) $\frac{||\hat{p}_s(t)-p_s||}{||\hat{p}_s(0)-p_s||}$ w.r.t. t.

Fig. 8. Localization by ECHO.

A simulation result with the estimates of sensor positions obtained by ECHO is shown in Fig. 9(b). The red stars represent the estimates while the blue circles are the true positions. The simulation result shows that the estimates perfectly matches the true positions of the sensor nodes. Two simulation results with the estimates of sensor positions obtained by MDS are plotted in Fig. 9(c) and 9(d). The line segments linked between red stars and blue circles in Fig. 9(c) and 9(d) indicate the localization errors. Similar to other gradient based optimization methods (e.g., [10] [11]), the MDS algorithm is sensitive to the choice of initial values. Fig. 9(c) and 9(d) show two different localization results obtained by the MDS algorithm with different initial values. As shown in Fig. 9(d), even without noises, the MDS algorithm may still not converge to the exact positions of the sensor nodes with a bad choice of initial values, which is also observed in [5]. In contrast, our proposed distributed iterative algorithm (ECHO) can globally exponentially converge to the true locations.

Note that the running of MDS depends on a distance matrix containing the distance measurements between every pair of nodes in the network while our algorithm only uses local information of neighbors. For a fair comparison, these un-localizable nodes are removed in the simulations before we use either ECHO or MDS for localization. For example, the node at the left bottom of Fig. 9(a) is removed since it does not have three paths to the anchor nodes. Details on how to detect and remove un-localizable nodes can be found in [2] [32].

VII. Conclusion

In this paper, we developed a criterion and algorithm to determine the sign pattern of a barycentric coordinate for a sensor node with arbitrary deployment of neighbors. Based on the generalized barycentric coordinate representation, we then developed a new localization algorithm for randomly deployed sensor networks, without restrictive assumptions on anchor nodes or neighboring nodes. The proposed localization algorithm is implemented in an iterative and distributed manner. A particular advantage of such implementation is that is robust against topological changes of the network, i.e., the iterative solutions adapt naturally when the sensor nodes and their distance measurements are added or dropped over time. In addition to the localization scheme, a localizability condition was also obtained, which shows that the existence of at least three disjoint paths from any node to the set of anchor nodes in the graph associated with the generalized barycentric coordinate representation is necessary and sufficient.

Acknowledgment

This work is supported by the National Natural Science Foundation of China Key Program 61134001 and General Program 61273113 as well as Zhejiang Provincial Natural Science Foundation of China LR13F030002.

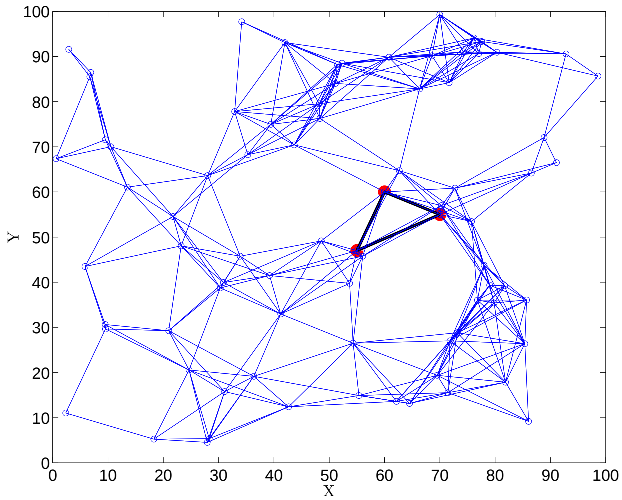
Appendix

Proof of Lemma 3: 1) $\Longrightarrow$ 2) If 1) holds, then by removing extra edges for each node $i \in V - R$ except the three edges on three disjoint paths from R to i, it results in a subgraph of $\mathcal{G}$, denoted by $\mathcal{T} = (V, \bar{E})$. If necessary, we re-label the nodes such that $r_1 = n-2$, $r_2 = n-1$, and $r_3 = n$. Then it is clear that each row of $L \in \{L \in \mathcal{L}(\mathcal{T}) : L\xi = 0, L\zeta = 0\}$ from 1 to $n-3$ has exactly four non-zero entries and thus is only one degree of freedom. I.e., it must satisfy

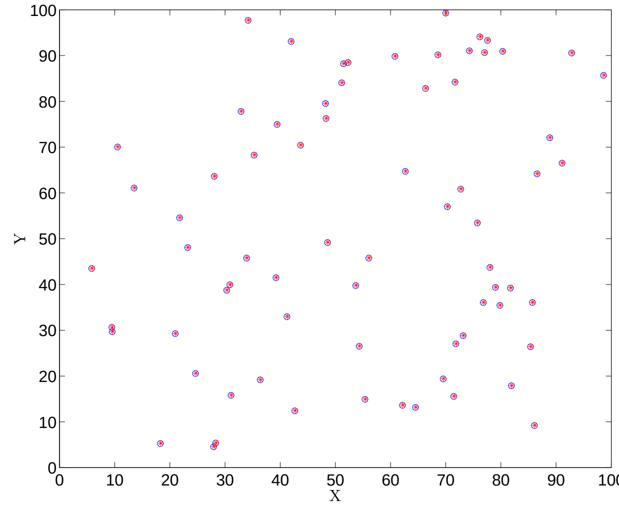
$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ \xi_i & \xi_{j_i^1} & \xi_{j_i^2} & \xi_{j_i^3} \\ \zeta_i & \zeta_{j_i^1} & \zeta_{j_i^2} & \zeta_{j_i^3} \end{bmatrix} \begin{bmatrix} L_{ii} \\ L_{ij_i^1} \\ L_{ij_i^2} \\ L_{ij_i^3} \end{bmatrix} = 0, \tag{22}$$

for $i = 1, \ldots, n-3$, where L_{ii} and $L_{ij_i^k}$ $(k = 1, 2, 3)$ are the four nonzero entries of the ith row of L, and ξ_k and ζ_k for $k = i, j_i^1, j_i^2, j_i^3$ are the corresponding components of ξ and ζ respectively. Moreover, j_i^1, j_i^2, and j_i^3 are the three neighbors of i.

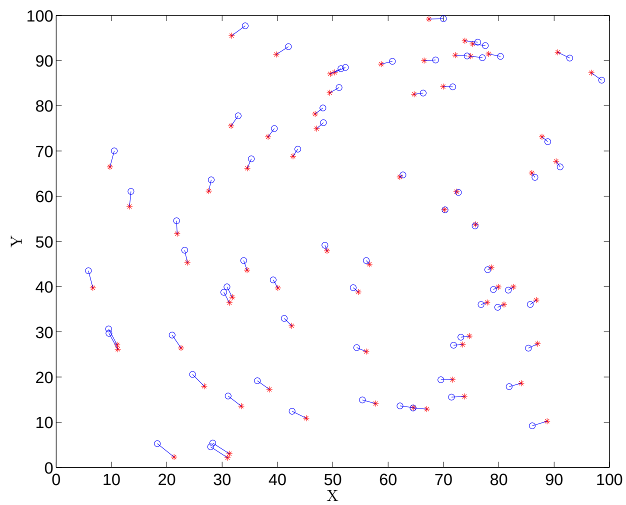
First, we show that for any $L \in \{L \in \mathcal{L}(\mathcal{T}) : L\xi = 0, L\zeta = 0\}$ with ξ and ζ being generic, all principal minors of L_R are distinct from zero. Let M be an s-th order principal sub-matrix of L_R corresponding to the subset of nodes $U \subset V - R$. If there exists a subset of nodes in U such that every of them



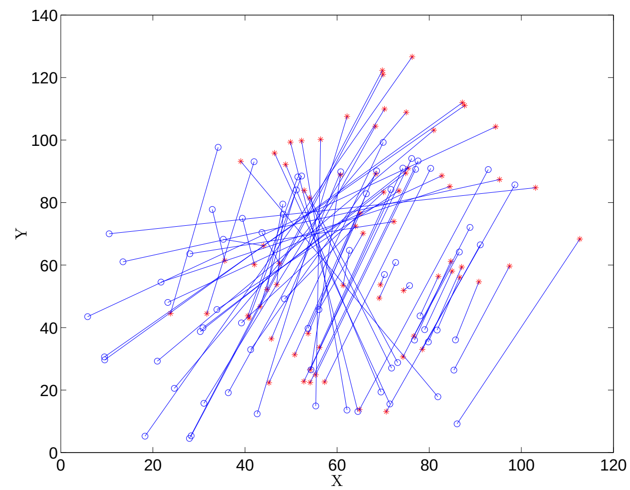
(a) A configuration of sensor network of 80 nodes, where the three anchor nodes are represented by red circles while the others are sensor nodes with unknown locations. The lines between pairs of nodes indicate that the distances are available for these pairs.



(b) A localization result by ECHO, which shows the estimates of the sensor positions (represented by red stars) perfectly match the true positions (blue circles).



(c) One localization result by MDS, which converges to the true positions for an initial condition.



(d) Another localization result by MDS, which does not converge to the true positions for another initial condition.

Fig. 9. Performance comparison between ECHO and MDS.

has no neighbors in U, then the matrix M must be of the following form by re-labeling these nodes as $1, \ldots, k$.

$$M = \begin{bmatrix} M_1 & 0 \\ 0 & M_2 \end{bmatrix},$$

where $M_1 \in \mathbb{R}^{k \times k}$ is a diagonal matrix and M_2 is a square matrix. Notice that the diagonal entries of M_1 are all nonzero. So for this particular case, to show $\det(M) \neq 0$, we only need to show M_2 has non-zero determinant, whose corresponding nodes all have neighbors in U. Therefore, without loss of generality, in the following argument we assume that every node in U has neighbors in U.

Now suppose by contradiction that $\det(M) = 0$. Thus we have a nonzero vector $\eta_s \in \mathbb{R}^s$ such that $M\eta_s = 0$. Denote by η the n-dimensional vector whose entries corresponding to the nodes not in U are zero and whose entries corresponding to the nodes in U are the corresponding components of η_s. Then it can be known that for $i \in U$,

$$\begin{bmatrix} \eta_i & \eta_{j_i^1} & \eta_{j_i^2} & \eta_{j_i^3} \end{bmatrix} \begin{bmatrix} L_{ii} \\ L_{ij_i^1} \\ L_{ij_i^2} \\ L_{ij_i^3} \end{bmatrix} = 0, \tag{23}$$

where j_i^1, j_i^2, and j_i^3 are the three neighbors of node i. Recall that $[L_{ii}\ L_{ij_i^1}\ L_{ij_i^2}\ L_{ij_i^3}]$ has only one degree of freedom, so it follows from (22) and (23) that for $i \in U$ there must exist α_i, β_i and γ_i (not all zeros) such that

$$\begin{bmatrix} \eta_i \\ \eta_{j_i^1} \\ \eta_{j_i^2} \\ \eta_{j_i^3} \end{bmatrix} = \alpha_i \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} + \beta_i \begin{bmatrix} \xi_i \\ \xi_{j_i^1} \\ \xi_{j_i^2} \\ \xi_{j_i^3} \end{bmatrix} + \gamma_i \begin{bmatrix} \zeta_i \\ \zeta_{j_i^1} \\ \zeta_{j_i^2} \\ \zeta_{j_i^3} \end{bmatrix}. \tag{24}$$

Recall that if j_i^k $(k = 1, 2, 3)$ is not in U, then $\eta_{j_i^k} = 0$. So there is one type of constraint from (24) as the following form

$$\alpha_i + \beta_i \xi_{j_i^k} + \gamma_i \zeta_{j_i^k} = 0. \tag{25}$$

Moreover, if j_i^k $(k = 1, 2, 3)$ is in U, then (24) results in another type constraint

$$\alpha_i + \beta_i \xi_{j_i^k} + \gamma_i \zeta_{j_i^k} = \alpha_{j_i^k} + \beta_{j_i^k} \xi_{j_i^k} + \gamma_{j_i^k} \zeta_{j_i^k}, \tag{26}$$

since both sides are equal to $\eta_{j_i^k}$.

By the condition that every node $i \notin R$ has three disjoint paths from R in $\mathcal{T}$, we then know for any node $i \in U$ there exist three disjoint paths from the set $V - U$ to i. We decompose $\mathcal{T}$ into three subgraphs, denoted by $\mathcal{T}^1$, $\mathcal{T}^2$ and $\mathcal{T}^3$ of the same node set V, such that every node other than the nodes in R has only one neighbor and every node in U is reachable from a node outside of U. For any $i \in U$, denote by j_i^1, j_i^2, and j_i^3 the neighbor of node i in $\mathcal{T}^1$, $\mathcal{T}^2$, and $\mathcal{T}^3$, respectively. We let $N_1 \in \mathbb{R}^{s \times s}$ be the sub-matrix of a binary Laplacian for $\mathcal{T}^1$ with the rows and columns corresponding to nodes in $V - U$ crossed out. That is, the diagonal entries of N_1 are 1's, the (i, j_i^1)-th entry is -1 only if $j_i^1 \in U$, and all other off-diagonal entries are zeros. Furthermore, we define N_2 and N_3 in $\mathbb{R}^{s \times s}$ in the same way according to $\mathcal{T}^2$ and $\mathcal{T}^3$ respectively. Moreover, we define N_1^ξ and N_1^ζ in $\mathbb{R}^{s \times s}$ of the same zero/nonzero pattern as N_1, whose i-th diagonal entries are $\xi_{j_i^1}$ and $\zeta_{j_i^1}$ respectively, and whose (i, j_i^1)-th off-diagonal entries are $-\xi_{j_i^1}$ and $-\zeta_{j_i^1}$ only if $j_i^1 \in U$. Similarly, N_2^ξ and N_2^ζ in $\mathbb{R}^{s \times s}$ are defined of the same zero/nonzero pattern as N_2, whose i-th diagonal entries are $\xi_{j_i^2}$ and $\zeta_{j_i^2}$, and whose (i, j_i^2)-th off-diagonal entries are $-\xi_{j_i^2}$ and $-\zeta_{j_i^2}$ only if $j_i^2 \in U$. Finally, N_3^ξ and N_3^ζ in $\mathbb{R}^{s \times s}$ are defined of the same zero/nonzero pattern as N_3 by the same manner (i.e., the nonzero weights relate to $\xi_{j_i^3}$ and $\zeta_{j_i^3}$ respectively). Then we can write the linear constraints in (25) and (26) in a concise form as

$$\begin{bmatrix} N_1 & N_1^\xi & N_1^\zeta \\ N_2 & N_2^\xi & N_2^\zeta \\ N_3 & N_3^\xi & N_3^\zeta \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} = 0, \tag{27}$$

where $\alpha = [\alpha_1, \ \cdots, \ \alpha_s]^T$, $\beta = [\beta_1, \cdots, \ \beta_s]^T$, and $\gamma = [\gamma_1, \cdots, \ \gamma_s]^T$.

Since every node in U is reachable from some nodes outside of U in terms of $\mathcal{T}^1$, it is obtained from algebraic graph theory that N_1 has a non-zero determinant, and so are N_1^ξ and N_1^ζ for generic ξ and ζ due to the same zero/nonzero pattern as N_1. For the same reason, we can obtain that $\det(N_2) \neq 0$, $\det(N_3) \neq 0$, $\det(N_2^\xi) \neq 0$ and $\det(N_3^\xi) \neq 0$ for a generic ξ, and $\det(N_2^\zeta) \neq 0$ and $\det(N_3^\zeta) \neq 0$ for a generic ζ. Denote

$$N = \begin{bmatrix} N_1 \\ N_2 \\ N_3 \end{bmatrix}, \ N^\xi = \begin{bmatrix} N_1^\xi \\ N_2^\xi \\ N_3^\xi \end{bmatrix} \text{ and } N^\zeta = \begin{bmatrix} N_1^\zeta \\ N_2^\zeta \\ N_3^\zeta \end{bmatrix}.$$

We use l_i, l_i^ξ, and l_i^ζ to represent the i-th column of N, N^ξ, and N^ζ, respectively. So N is column linearly independent. Moreover, note that l_i^ξ has the same zero/nonzero pattern as l_i. So we can know that once l_i^ξ for a choice of generic ξ is linearly independent with l_i, then $[N \ l_i^\xi]$ is column linearly independent. Recall by our assumption that every node $i \in U$ has neighbors in U, which means the column of N^ξ corresponding to node i contains a free variable ξ_i. Also, notice that each column i has at least two non-zero entries, so by varying ξ_i we can make $[N \ l_i^\xi]$ column linearly independent. Repeating the same argument, we can eventually reach the conclusion that $[N \ N^\xi]$ is column linearly independent for a generic ξ. For the same reason, we can show that $[N \ N^\xi \ N^\zeta]$ is column linearly independent for generic ξ and ζ. Therefore, eq. (27) has a unique solution, i.e., $\alpha = \beta = \gamma = 0$, which contradicts to the assumption that α_i, β_i and γ_i are not identically zero. So we conclude that all principal minors of L_R are distinct from zero for $L \in \{L \in \mathcal{L}(\mathcal{T}) : L\xi = 0, L\zeta = 0\}$.

Second, since we already showed that for any $L \in \{L \in \mathcal{L}(\mathcal{T}) \ : \ L\xi = 0, L\zeta = 0\}$ with ξ and ζ being generic, all principal minors of L_R are distinct from zero, it follows straightforward that for almost all $L \in \{L \in \mathcal{L}(\mathcal{G}) : L\xi = 0, L\zeta = 0\}$ with ξ and ζ being generic, all principal minors of L_R are distinct from zero as well since some nonzero entries are added into L associated to graph $\mathcal{G}$.

2) $\Longrightarrow$ 1) We prove it in a contrapositive form. Suppose that there exists a node $i \notin R$ such that there are no three disjoint paths in $\mathcal{G}$ from R to i. That is, after removing two nodes, without loss of generality, say k_1 and k_2, a subset W of nodes becomes not reachable from R. Denote the set of remaining nodes as $\bar{W}$, which are still reachable from R after removing k_1 and k_2. It is certain that $R \in \bar{W}$ and after removing nodes k_1 and k_2, the nodes in W are not reachable from any node in $\bar{W}$. Suppose the total number of nodes in W is m. If necessary, re-label the nodes in W as $1, \cdots, m$, change the labels of node k_1 and k_2 to $m + 1$ and $m + 2$, and re-label the nodes in $\bar{W}$ as $m + 3, \cdots, n$. Then the matrix L after re-labeling satisfies $L(i, j) = 0$ for $i \in W$ and $j \in \bar{W}$. That is, L is of the following form

$$\left[\begin{array}{c|c|c|c} L_w & l_1 & l_2 & 0 \\ \hline * & * & * & * \end{array}\right],$$

where $L_w \in \mathbb{R}^{m \times m}$ and $l_1, l_2 \in \mathbb{R}^m$. Re-order the components of ξ and ζ in the same way as relabeling the nodes, and denote the resulting vectors by $[\xi_a^T, \ \xi_b^T]^T$ where $\xi_a \in \mathbb{R}^{m+2}$ and $\xi_b \in \mathbb{R}^{(n-m-2)}$, and $[\zeta_a^T, \ \zeta_b^T]^T$ where $\zeta_a \in \mathbb{R}^{m+2}$ and $\zeta_b \in \mathbb{R}^{(n-m-2)}$. According to the definition of L, we have

$$[L_w \ l_1 \ l_2]\mathbf{1}_{m+2} = 0, \ [L_w \ l_1 \ l_2]\xi_a = 0 \text{ and } [L_w \ l_1 \ l_2]\zeta_a = 0.$$

As $\mathbf{1}_{m+2}$, ξ_a, and ζ_a are linearly independent for generic ξ and ζ, we then know that $[L_w \ l_1 \ l_2]$ must be row linearly dependent, which means $\det(L_R) = 0$ for any $L \in \{L \in \mathcal{L}(\mathcal{G}) : L\xi = 0, L\zeta = 0\}$. ■

References

[1] N. Priyantha, "The cricket indoor location system," Ph.D. dissertation, Massachusetts Institute of Technology, 2005.

[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*, vol. 15, no. 2, pp. 177–191, Feb. 2009.

[3] 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 *Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies*, Hong Kong, Mar. 2004, pp. 2673–2684.

[4] L. Doherty, K. S. J. Pister, and L. El Ghaoui, "Convex position estimation in wireless sensor networks," in *Proceedings of the 20th Annual Joint Conference of the IEEE Computer and Communications Societies*, Anchorage, AK, Apr. 2001, pp. 1655–1663.

[5] X. Ji and H. Zha, "Sensor positioning in wireless ad-hoc sensor networks using multidimensional scaling," in *Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies*, Mar. 2004, pp. 2652–2661.

[6] A. M.-C. So and Y. Ye, "Theory of semidefinite programming for sensor network localization," *Mathematical Programming*, vol. 109, no. 2-3, pp. 367–384, Mar. 2007.

[7] P. Biswas, T.-C. Lian, T.-C. Wang, and Y. Ye, "Semidefinite programming based algorithms for sensor network localization," *ACM Transactions on Sensor Networks*, vol. 2, no. 2, pp. 188–220, May 2006.

[8] 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*, Baltimore, MD, Nov. 2004, pp. 50–61.

[9] D. Niculescu and B. Nath, "DV based positioning in ad hoc networks," *Telecommunication Systems*, vol. 22, no. 1-4, pp. 267–280, Jan. 2003.

[10] M. Rabbat and R. Nowak, "Distributed optimization in sensor networks," in *Proceedings of the 3rd international symposium on Information processing in sensor networks*, Berkeley, CA, Apr. 2004, pp. 20–27.

[11] B. H. Cheng, L. Vandenberghe, and K. Yao, "Distributed algorithm for node localization in wireless ad-hoc networks," *ACM Transactions on Sensor Networks*, vol. 6, no. 1, p. 8, Dec. 2009.

[12] M. Cao, B. D. O. Anderson, and A. S. Morse, "Sensor network localization with imprecise distances," *Systems & control letters*, vol. 55, no. 11, pp. 887–893, Nov. 2006.

[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*, vol. 57, no. 5, pp. 2000–2016, May 2009.

[14] U. A. Khan, S. Kar, B. Sinopoli, and J. M. F. Moura, "Distributed sensor localization in euclidean spaces: Dynamic environments," in *Proceedings of the 46th Annual Allerton Conference on Communication, Control, and Computing*, Urbana-Champaign, IL, Sep. 2008, pp. 361–366.

[15] M. Franceschelli and A. Gasparri, "On agreement problems with gossip algorithms in absence of common reference frames," in *IEEE International Conference on Robotics and Automation*, Anchorage, AK, May 2010, pp. 4481–4486.

[16] R. Aragues, L. Carlone, C. Sagues, and G. Calafiore, "Distributed centroid estimation from noisy relative measurements," *Systems & Control Letters*, vol. 61, no. 7, pp. 773–779, Jul. 2012.

[17] A. Singer, "A remark on global positioning from local distances," *Proceedings of the National Academy of Sciences*, vol. 105, no. 28, pp. 9507–9511, Jul. 2008.

[18] Y. Diao, Z. Lin, M. Fu, and H. Zhang, "A new distributed localization method for sensor networks," in *Proceedings of the 9th Asian Control Conference*, Istanbul, Turkey, Jun. 2013, pp. 1–6.

[19] H. Coxeter, *Introduction to Geometry*. John Wiley & Sons, Inc., 1969.

[20] M. Sippl and H. Scheraga, "Cayley-menger coordinates," *Proceedings of the National Academy of Sciences*, vol. 83, no. 8, pp. 2283–2287, Apr. 1986.

[21] F. Thomas and L. Ros, "Revisiting trilateration for robot localization," *IEEE Transactions on Robotics*, vol. 21, no. 1, pp. 93–101, Feb. 2005.

[22] R. Connelly, "Generic global rigidity," *Discrete & Computational Geometry*, vol. 33, no. 4, pp. 549–563, Apr. 2005.

[23] S. J. Gortler, A. D. Healy, and D. P. Thurston, "Characterizing generic global rigidity," *American journal of mathematics*, vol. 132, no. 4, pp. 897–939, Aug. 2010.

[24] J. M. Hendrickx, B. D. O. Anderson, and V. D. Blondel, "Rigidity and persistence of directed graphs," in *Proceedings of the 44th IEEE Conference on Decision and Control*, Seville, Spain, Dec. 2005, pp. 2176–2181.

[25] D. K. Goldenberg, P. Bihler, M. Cao, J. Fang, B. D. O. Anderson, A. S. Morse, and Y. R. Yang, "Localization in sparse networks using sweeps," in *Proceedings of the 12th annual international conference on Mobile computing and networking*, Los Angeles, CA, Sep. 2006, pp. 110–121.

[26] L. F. Richardson, "The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam," *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, vol. 210, pp. 307–357, 1911.

[27] Y. Saad, *Iterative methods for sparse linear systems*. SIAM, 2003.

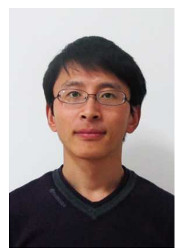
[28] P. Yang, R. A. Freeman, G. J. Gordon, K. M. Lynch, S. S. Srinivasa, and R. Sukthankar, "Decentralized estimation and control of graph connectivity for mobile sensor networks," *Automatica*, vol. 46, no. 2, pp. 390–396, Feb. 2010.

[29] M. Franceschelli, A. Gasparri, A. Giua, and C. Seatzu, "Decentralized Laplacian eigenvalues estimation for networked multi-agent systems," in *Proceedings of the 48th IEEE Conference on Decision and Control*, Shanghai, China, Dec. 2009, pp. 2717–2722.

[30] T. Sahai, A. Speranzon, and A. Banaszuk, "Hearing the clusters of a graph: A distributed algorithm," *Automatica*, vol. 48, no. 1, pp. 15–24, Jan. 2012.

[31] D. Marelli and M. Fu, "Distributed weighted least squares estimation with fast convergence in large-scale systems," in *Proceedings of the 52th IEEE Conference on Decision and Control*, Florence, Italy, 2013, pp. 5432–5437.

[32] Z. Yang, Y. Liu, and X. Li, "Beyond trilateration: On the localizability of wireless ad hoc networks," *IEEE/ACM Transactions on Networking*, vol. 18, no. 6, pp. 1806–1814, Dec. 2010.



Yingfei Diao is now a Research Fellow in the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore.

He received the Ph.D. degree from the School of Control Science and Engineering, Shandong University, China, in 2013, and Bachelor degree from the School of Information and Electrical Engineering, China University of Mining and Technology, China, in 2006. He has been a visiting student in the University of Newcastle, Australia, from March 2012 to February 2013.

His research interests include cooperative localization for sensor network, multi-agent systems, algebraic graph theory, distributed estimation and collective behavior of robotics.

Zhiyun Lin received his bachelor degree in Electrical Engineering from Yanshan University, China, in 1998, master degree in Electrical Engineering from Zhejiang University, China, in 2001, and PhD degree in Electrical and Computer Engineering from the University of Toronto, Canada, 2005.

He was a Postdoctoral Research Associate in the Department of Electrical and Computer Engineering, University of Toronto, Canada, from 2005 to 2007. He joined the College of Electrical Engineering, Zhejiang University, China, in 2007. Currently, he is a Professor of Systems Control in the same department. He is also affiliated with the State Key Laboratory of Industrial Control Technology at Zhejiang University. He held visiting professor positions at several universities including The Australian National University (Australia), University of Cagliari (Italy), and University of Newcastle (Australia).

His research interests focus on distributed control, estimation and optimization, coordinated and cooperative control of multi-agent systems, hybrid and switched control system theory, and locomotion control of biped robots. He is a senior member of IEEE. He is currently an associate editor for Hybrid systems: Nonlinear Analysis and International Journal of Wireless and Mobile Networking.

Minyue Fu received his Bachelor's Degree in Electrical Engineering from the University of Science and Technology of China, Hefei, China, in 1982, and M.S. and Ph.D. degrees in Electrical Engineering from the University of Wisconsin-Madison in 1983 and 1987, respectively.

From 1987 to 1989, he served as an Assistant Professor in the Department of Electrical and Computer Engineering, Wayne State University, Detroit, Michigan. He joined the Department of Electrical and Computer Engineering, the University of Newcastle, Australia, in 1989. Currently, he is a Chair Professor in Electrical Engineering. He was a Visiting Associate Professor at University of Iowa in 1995-1996, a Senior Fellow/Visiting Professor at Nanyang Technological University, Singapore, 2002, and Visiting Professor at Tokyo University in 2003. He has held a ChangJiang Visiting Professorship at Shandong University, a visiting Professorship at South China University of Technology, and a Qian-ren Professorship at Zhejiang University in China.

He was elected to a Fellow of IEEE in late 2003. His main research interests include control systems, signal processing and communications. His current research projects include networked control systems, smart electricity networks and super-precision positioning control systems. He has been an Associate Editor for the IEEE Transactions on Automatic Control, Automatica, IEEE Transactions on Signal Processing, and Journal of Optimization and Engineering.