Distributed Localization for 2-D Sensor Networks With Bearing-Only Measurements Under Switching Topologies

Zhiyun Lin, *Senior Member, IEEE*, Tingrui Han, Ronghao Zheng, and Minyue Fu, *Fellow, IEEE*

***Abstract*—This paper investigates the problem of bearing measurement based distributed localization for sensor networks that are undirected and switching. Each node holds a local coordinate system with no knowledge about the global coordinate system and measures the bearing angle information about its neighbors in its local coordinate system. A novel scheme for localization is developed using a complex Laplacian to overcome the challenges due to the absence of a global coordinate system and the presence of topology switching in communication. First, by using bearing-only measurements, an algorithm is proposed to establish linear equation constraints for the coordinates of sensor nodes in the global coordinate frame. The main idea is that each node uses its own bearing and its neighbors' bearing information to construct a similar configuration, though it is not able to recover the true configuration by using only bearing measurements. Second, a distributed iterative algorithm is proposed such that all the sensor nodes can cooperatively find the true coordinates of themselves. It is shown that the algorithm exponentially converges, provided that the communication network jointly satisfies certain connectivity properties. The simulation results validate our proposed algorithm.**

***Index Terms*—Bearing measurement, distributed localization, switching topology.**

I. Introduction

LOCALIZATION problems can be found in many applications such as tracking a target, collecting sensor data with location information, intelligent home systems and so on [1]–[3]. Due to the advantages of distributed localization in reducing the network resource consumption and fully utilizing the computation ability of each sensor node, numerous distributed localization schemes have been developed using different kinds of local measurement information in recent years. Three kinds of measurements are often adopted, namely, distance measurements [4]–[10], bearing measurements [11]–[17] and relative position measurements [18]–[22].

Bearing-only distributed localization has received growing interest due to its potential applications for vision-based systems. Distributed localization problems with bearing-only measurements can be described as follows. In a large sensor network, there are several sensor nodes called anchors, which can obtain their own absolute position information in a global coordinate system through some equipment such as GPS. Other sensor nodes are to be localized. They can acquire bearing-only measurements of neighboring nodes. The objective is to design a distributed position estimation algorithm, by which each sensor node can use locally available information to iteratively determine its own coordinate in the global coordinate system.

For the bearing-only localization problem, a fundamental problem is whether all the nodes in a sensor network can localize themselves using bearing-only measurements. From this perspective, it has been shown in [12], [15], [23] that if a 2-D network is infinitesimally bearing rigid with at least two anchor nodes, then the network is localizable. Recently, [17] extends this problem to d-dimensional spaces and shows that if every infinitesimal bearing motion of the network involves at least one anchor, then the network is localizable. However, all these works focus on sensor networks under fixed communication topologies. In practice, communications between sensor nodes may fail temporarily or even permanently, which means that a sensor node may not receive its neighbors' estimates temporarily or even permanently. So the communication topologies in the localization process may switch over time. Under such a setup, it is remaining open results on whether a sensor network is localizable in a distributed manner.

Moreover, for a large-scale network, all the sensor nodes may not agree on a common direction without extra equipments such as compass. That is, the sensor nodes may not have the knowledge about the orientation of the global coordinate system and thus they can only measure the bearing angles in their own local frames. However, in the literature such as [15] and [17], it is often assumed that all the sensor nodes know the orientation of the global coordinate system in order to make their distributed localization algorithms work. More specifically, consider a sensor network with the estimate of each node i's coordinate in

Manuscript received February 3, 2016; revised July 23, 2016; accepted August 19, 2016. Date of publication September 8, 2016; date of current version October 6, 2016. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Wee Peng Tay. This work was supported in part by the Zhejiang Provincial Natural Science Foundation of China under Grant LR13F030002, in part by the Zhejiang Open Foundation of the Most Important Subjects, the Fundamental Research Funds for the Central Universities, China, under Grant 110201-172210151, and in part by the Key Laboratory of System Control and Information Processing, Ministry of Education, Shanghai, China, under Grant Scip201504. *(Corresponding authors: Zhiyun Lin and Ronghao Zheng.)*

Z. Lin and T. Han are with the State Key Laboratory of Industrial Control Technology, College of Electrical Engineering, Zhejiang University, Hangzhou 310027, China (e-mail: linz@zju.edu.cn; hantingrui@zju.edu.cn).

R. Zheng is with the State Key Laboratory of Industrial Control Technology, College of Electrical Engineering, Zhejiang University, Hangzhou 310027, China, and also with the Zhejiang Province Marine Renewable Energy Electrical Equipment and System Technology Research Laboratory, Zhejiang University, Hangzhou 310027, China (e-mail: rzheng@zju.edu.cn).

M. Fu is with the School of Electrical Engineering and Computer Science, University of Newcastle, Callaghan, NSW 2308, Australia, and also with the School of Control Science and Engineering, Zhejiang University, Hangzhou 310027, China (e-mail: minyue.fu@newcastle.edu.au).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2016.2607144

the global coordinate system denoted by $\hat{p}_i$. The basic idea in [13]–[17] is that each node i updates its estimate to drive $v_{ij}^T(\hat{p}_j - \hat{p}_i)$ to zero, where v_{ij} is a vector derived from the measured bearing angles, which is perpendicular to $p_j - p_i$. With this method, v_{ij} must be obtained with respect to the global coordinate frame. Thus a common orientation is needed for every node. To address the issue caused by different frame orientations, algorithms for asymptotically aligning the orientations of the local frames with the one of the global frame are required. One way for orientation alignment is a sequential method, which begins with an anchor node and adjusts the orientation of other nodes one by one or group by group [13], [14]. But the drawback is that it may accumulate the orientation alignment errors along the sequence of propagation. An alternative way is to utilize a simultaneous orientation consensus law for distributed bearing-only localization with different frame orientations [16]. Nevertheless, an assumption is needed, i.e., initial angles between any two local frames should be less than π. This assumption may not be satisfied unless the nodes are equipped with some extra sensors. Moreover, if the anchors only know their own absolute locations and have no access to the global orientation, then all these methods no longer work.

In this paper, we aim to overcome these difficulties and provide a novel scheme for bearing measurement based distributed localization. That is to say, a sensor network under consideration does not share a common sense of north and moreover the communication topology between the nodes switches over time. To address the localization problem under such a scenario, our first effort is to establish linear equation constraints for the coordinates of sensor nodes in the global coordinate frame by using only local bearing measurements and local information exchange in local coordinate systems. Our main idea is to let each node use its own bearing and its neighbors' bearing information to construct a similar configuration, although it is not able to recover the true configuration by using bearing-only measurements. Then linear equation constraints can be established for the absolute coordinates by using the similarity property of similar configurations, which overcomes the challenge due to the absence of a global coordinate system. After obtaining the linear equation constraints, the remaining challenge lies in the aspect of finding an effective distributed algorithm under switching communication topology to ensure that all sensor nodes can cooperatively solve the linear equations to get the absolute coordinates of themselves. To overcome this difficulty, a Hermitian matrix based iteration algorithm is proposed. Moreover, the idea similar to persistent excitation is introduced, which plays a key role in establishing convergence under switching communication topologies. For the localization scheme proposed in this paper, a sufficient graphical condition is also provided, which requires each node to have at least two neighbors that are mutually neighbors. Though the condition is necessary if a node has exactly two neighbors, it may not be necessary if a node has more than two neighbors. However, from the practical viewpoint, if the sensor nodes have the ability to increase their communication range, it may become always possible to make the neighbors of every node also mutual neighbors.

To sum up, the main contributions of the paper are as follows. (i) A distributed localization algorithm is proposed under switching topologies. The algorithm requires no common orientation for local frames, but ensures globally exponential convergence. (ii) The distributed localization algorithm relies on the establishment of linear equation constraints to construct a complex Laplacian, which is novel. (iii) Graphical conditions are obtained for switching topologies such that under the proposed algorithm the estimates will converge to the true locations exponentially.

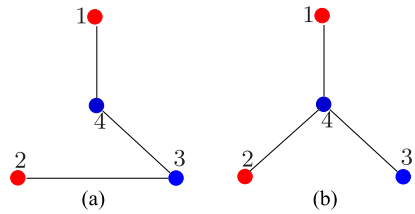


Fig. 1. (a) Node 3 is 2-reachable from $\{1, 2\}$. (b) Node 3 is not 2-reachable from $\{1, 2\}$.

The rest of the paper is organized as follows. Section II provides preliminaries to graph theory and formulates the localization problem. Section III develops a new scheme for establishing linear equation constraints about the absolute locations of the sensor nodes. To make it clearer for understanding the basic idea, fixed topologies are considered in Section IV. Then Section V presents a distributed algorithm and provides convergence analysis for switching topologies. Simulation examples are given in Section VI and conclusions are drawn in Section VII.

Notation: $\mathbb{C}$ represents the set of complex numbers. $\iota = \sqrt{-1}$ denotes the imaginary unit. $\mathbf{1}_n$ denotes the n-dimensional vector of ones and I_n denotes the identity matrix of order n. For a complex number c, $|c|$ and $\bar{c}$ represent the modulus and the conjugate respectively. For a complex vector or matrix A, A^T and A^H denote the transpose and conjugate transpose respectively. rank(A) represents the rank of A.

II. Preliminaries and Problem Statement

A. Graph Theory

An *undirected graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consists of a node set $\mathcal{V}$ of elements called *nodes* and an edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ of pairs of nodes called *edges*. For each node $i \in \mathcal{V}$, let $\mathcal{N}_i = \{j \in \mathcal{V} : (j, i) \in \mathcal{E}\}$ denote the set of its *neighbors*.

For a graph $\mathcal{G}$, a *complex Laplacian matrix* $L \in \mathbb{C}^{n\times n}$ is defined as follows:

$$L(i,j) = \begin{cases} -w_{ij} & \text{if } i \neq j \text{ and } j \in \mathcal{N}_i \\ 0 & \text{if } i \neq j \text{ and } j \notin \mathcal{N}_i \\ \sum\limits_{k\in\mathcal{N}_i} w_{ik} & \text{if } i = j \end{cases}$$

where $w_{ij} \neq 0$ is called the complex weight associated with edge (j, i).

For a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, a node v is said to be *2-reachable* from a set $\mathcal{R} \subset \mathcal{V}$ if there exists a path from a node in $\mathcal{R}$ to v after removing any other one node except v. Consider for example the graphs in Fig. 1. In Fig. 1(a), let $\mathcal{R} = \{1, 2\}$ and it is known that node 3 is 2-reachable from $\mathcal{R}$ as after removing any other one node we are still able to find a path from a node

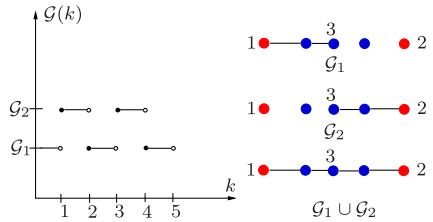


Fig. 2. A dynamic graph $\mathcal{G}(k)$, for which node 3 is jointly 2-reachable from $\{1, 2\}$.

in $\mathcal{R}$ to node 3. In Fig. 1(b), again let $\mathcal{R} = \{1, 2\}$, but node 3 is not 2-reachable from the set $\mathcal{R}$ as if we remove node 4, there is no path from any node in $\mathcal{R}$ to node 3.

A *dynamic graph* $\mathcal{G}(k) = (\mathcal{V}(k), \mathcal{E}(k))$ represents a graph whose node set and edge set change over time. For a time interval $[k_1, k_2]$ the *union graph* is defined as $\mathcal{G}([k_1, k_2]) = (\bigcup_{k\in[k_1,k_2]} \mathcal{V}(k), \bigcup_{k\in[k_1,k_2]} \mathcal{E}(k))$. A node v is called *jointly 2-reachable* from $\mathcal{R}$ if there exists $K > 0$ such that for every k, v is 2-reachable from $\mathcal{R}$ in the union graph $\mathcal{G}([k, k + K))$. An example is given in Fig. 2, for which node 3 is jointly 2-reachable from the set $\{1, 2\}$ since we can take $K = 2$ and for any k the union graph $\mathcal{G}([k, k + K)) = \mathcal{G}_1 \cup \mathcal{G}_2$, for which node 3 is 2-reachable from $\{1, 2\}$.

A graph $\mathcal{G}$ is said to be *connected* if there is a path between every pair of nodes. A dynamic graph $\mathcal{G}(k)$ is said to be *jointly connected* if there exists $K > 0$ such that for every k, the union graph $\mathcal{G}([k, k + K))$ is connected.

For a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and a set $\mathcal{R} \subset \mathcal{V}$, the *$\mathcal{R}$-induced graph in $\mathcal{G}$* is defined as $\Pi = (\mathcal{R}, \mathcal{E} \cap \mathcal{R} \times \mathcal{R})$.

A *configuration* of n nodes in $\mathbb{C}$ is defined by their coordinates in $\mathbb{C}$, denoted as $p = [p_1, \ldots, p_n]^T \in \mathbb{C}^n$, where each $p_i \in \mathbb{C}$ for $1 \le i \le n$. A *framework* is a graph $\mathcal{G}$ equipped with a configuration p, denoted as $\mathcal{F} = (\mathcal{G}, p)$. Two frameworks $(\mathcal{G}, p)$ and $(\mathcal{G}, q)$ are called *similar* if

$$p_i - p_j = \gamma e^{\iota\theta}(q_i - q_j),\ \forall i, j \in \mathcal{V}, \tag{1}$$

where $\gamma > 0$ is a scaling factor and θ represents a rotation, and we write $(\mathcal{G}, q) \sim (\mathcal{G}, p)$.

For a square matrix $E \in \mathbb{C}^{n\times n}$, the *associated graph* $\mathcal{G}(E)$ consists of n nodes labeled 1 through n where an edge leads from node j to node i $(i \neq j)$ if and only if the (i, j)th entry of E is nonzero.

B. Problem Formulation

We consider a sensor network consisting of a set of *anchor nodes*, whose absolute positions are already known, and sensor nodes, which are to be localized. Suppose each sensor node can acquire the bearing measurements of its neighboring nodes in its local coordinate system.

Each sensor node i holds a local coordinate system Σ_i and has no knowledge about the global coordinate system Σ_g. For a node i, let p_i denote its coordinate in Σ_g and let p_i^j denote its coordinate in Σ_j, where the superscript j for $j = 1, \ldots, n$ is used throughout the paper to represent the value in local coordinate system Σ_j. We use a complex number to represent

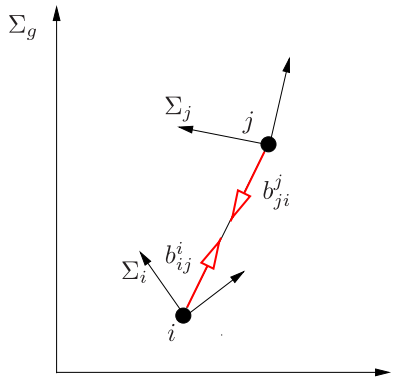


Fig. 3. An illustration for the bearing measurements.

p_i instead of a two-dimensional vector in this paper. Note that $p_i^i = 0$.

We say node j is a neighbor of node i if and only if i can obtain the bearing measurement about j, and j can communicate with i. That is to say, node i can access the normalized vector

$$b_{ij}^i = \frac{p_j^i - p_i^i}{|p_j^i - p_i^i|}$$

in node i's local coordinate system Σ_i.

Technically, we assume the following.

Assumption 2.1: The bearing measurements and the communications are bidirectional.

Fig. 3 shows an illustration for the bearing measurements, where node i can measure b_{ij}^i about node j, and node j can measure b_{ji}^j about node i.

For a network of n nodes with bearing-only measurements, at least two anchor nodes are needed for localization [14]. Thus, without loss of generality, we assume that there are two anchor nodes, whose coordinates in Σ_g are denoted by $p_1, p_2 \in \mathbb{C}$. The coordinates of other sensor nodes in Σ_g are denoted by $p_3, \ldots, p_n \in \mathbb{C}$. Let $p_a = [p_1, p_2]^T$, and $p_s = [p_3, \ldots, p_n]^T$. The following is a standing assumption.

Assumption 2.2: Assume that the sensor nodes do not overlap each other, i.e., $p_i \neq p_j,\ \forall i, j$.

Communications between sensor nodes may fail from time to time due to severe environmental factors. We use a dynamic graph $\bar{\mathcal{G}}(k) = (\mathcal{V}, \bar{\mathcal{E}}(k))$ to model the whole sensor network, where $\mathcal{V} = \mathcal{A} \cup \mathcal{S}$ with $\mathcal{A} = \{1, 2\}$ and $\mathcal{S} = \{3, \ldots, n\}$. Let $\bar{\mathcal{N}}_i(k)$ be the set of the neighbors of node i in $\bar{\mathcal{G}}(k)$. In other words, $(j, i) \in \bar{\mathcal{E}}(k)$ if and only if $j \in \bar{\mathcal{N}}_i(k)$. That is to say, when $(j, i) \in \bar{\mathcal{E}}(k)$, node i can access b_{ij}^i, and node j can access b_{ji}^j, and they can exchange information between each other via communication at time k.

Here we give some explanations about why the entire networks can be localized merely based on some local bearing measurements and how it is different from localization with bearing measurements in a global frame. Consider a simple example as in Fig. 4 where node 1 and 2 are anchors and node 3 is a sensor node. If the bearing measurements are acquired in the global coordinate system as in Fig. 4(a), then we are able to uniquely determine the position of node 3 since the two lines with specific bearing in the global frame and passing through

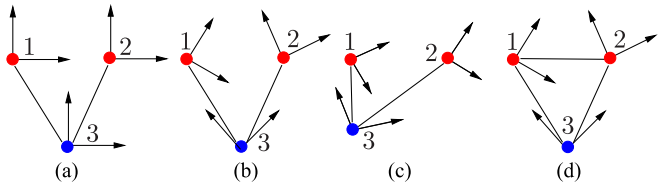


Fig. 4. (a) Bearing measurements in the global coordinate system. (b) A configuration with bearing measurements in local coordinate systems. (c) Different orientation leads to a different configuration with the same bearing constraints as in (b). (d) Connecting node 1 and 2 leads to the localizability with local bearing measurements.

node 1 and 2 must intersect at a point. However, this is not true when the orientations of the local frames are unknown. An example is given in Fig. 4(b)–4(c), for which different orientations lead to different configurations with both satisfying the bearing constraints in the local frames. That is to say, with only bearing measurements in the local frames, additional measurements are required for localizability. For instance, if we add an edge between node 1 and 2 as shown in Fig. 4(d), then node 3 can be localized since three angles in the triangle are determined.

The problem in this paper is described as follows. Given a dynamic graph $\bar{\mathcal{G}}(k)$ with the edge set $\bar{\mathcal{E}}(k)$ indicating the bearing measurements and communication topology in the network, develop a distributed localization algorithm and explore the graphical conditions for the whole network, which ensure globally exponential convergence.

III. Linear Equation Constraints

To solve the localization problem, our first step is to establish some equation constraints in terms of the coordinates of all sensor nodes. In this section, we aim to obtain such equation constraints using the complex linear representation concerning the coordinates in Σ_g for the sensor network. More specifically, for each sensor node i, consider i along with its neighbors $j \in \bar{\mathcal{N}}_i$. The goal is then to design complex weights $w_{ij} \in \mathbb{C}$, $j \in \bar{\mathcal{N}}_i$, with only bearing measurements in Σ_i and local information exchange such that

$$\sum_{j\in\bar{\mathcal{N}}_i} w_{ij}(p_j - p_i) = 0. \tag{2}$$

However, note that p_i and p_j $(j \in \bar{\mathcal{N}}_i)$ are the coordinates in Σ_g. Moreover, every node does not know the orientation of its own local frame with respect to the global frame Σ_g. So it is challenging to find complex weights w_{ij}'s using bearing-only measurements and local information exchange to establish the formula (2). To overcome the difficulty, we will consider to construct a similar framework, based on which a new approach will be developed to find proper complex weights. This can be done mainly due to the following reason. For a similar framework $(\mathcal{G}, q) \sim (\mathcal{G}, p)$, from (1) we know that

$$\sum_{j\in\bar{\mathcal{N}}_i} w_{ij}(q_j - q_i) = 0, \quad \forall i$$

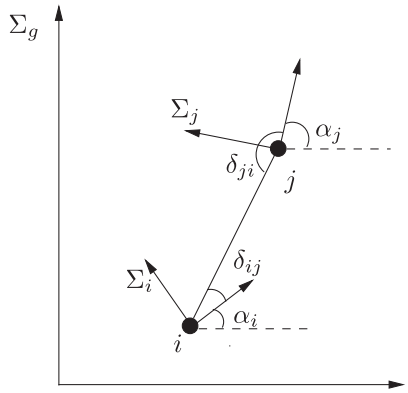


Fig. 5. An illustration for the relationship between different frames.

implies

$$\sum_{j\in\bar{\mathcal{N}}_i} w_{ij}(p_j - p_i) = 0, \quad \forall i.$$

In the following, we develop a *sensing-plus-communication scheme* to construct a similar framework and then compute complex weights, which are divided into three steps.

Step A1: Compute relative orientations of local frames.

Let δ_{ij} be the angle between b_{ij}^i and the real axis of Σ_i, and δ_{ji} be the angle between b_{ji}^j and the real axis of Σ_j. Denote α_i and α_j the angle between the local frame Σ_i, Σ_j and the global coordinate system Σ_g, respectively (Fig. 5).

Define the relative orientation as $\alpha_{ji} := \alpha_j - \alpha_i$. Then, for node i, the relative orientation α_{ji} can be calculated based on b_{ij}^i measured by onboard sensors and b_{ji}^j transmitted from its neighbor node j. That is,

$$e^{\iota\alpha_{ji}} = e^{\iota(\delta_{ij}-\delta_{ji}+\pi)} = \frac{b_{ij}^i}{b_{ji}^j}e^{\iota\pi}. \tag{3}$$

Step A2: Compute complex weights regarding two neighbors.

Consider node i with its two neighbors j and l. Let $\mathcal{G}_3$ represent the complete graph with node set $\{i, j, l\}$. Let $p' = [p_i, p_j, p_l]^T$. If node j and node l are mutual neighbors, and i, j and l are not collinear, then node i can construct a framework $(\mathcal{G}_3, q')$ with $q' = [q_i, q_j, q_l]^T$, which is similar to $(\mathcal{G}_3, p')$. The procedure of constructing a similar framework $(\mathcal{G}_3, q')$ is as follows.

Notice that node i can access b_{ij}^i and b_{il}^i via local sensing, and can obtain b_{jl}^j and b_{lj}^l through communication. Based on (3) in Step A1, node i can compute b_{jl}^i and b_{lj}^i according to the following formula

$$b_{jl}^i = e^{\iota\alpha_{ji}}b_{jl}^j, \text{ and } b_{lj}^i = e^{\iota\alpha_{li}}b_{lj}^l.$$

We then construct a similar framework $(\mathcal{G}_3, q')$ by setting $q_i = 0$ and $q_j = b_{ij}^i$, and solving q_l from the following formula

$$\begin{cases} \mathrm{Re}[\iota b_{il}^i(\overline{q_l - q_i})] = 0, \\ \mathrm{Re}[\iota b_{jl}^i(\overline{q_l - q_j})] = 0. \end{cases} \tag{4}$$

The equality (4) holds since we aim to obtain q_l satisfying

$$(q_l - q_i) \perp \iota b_{il}^i, \ (q_l - q_j) \perp \iota b_{jl}^i,$$

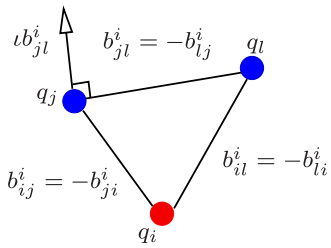


Fig. 6. An illustration for node i with two neighbors j and l.

where ιb^i_{jl} means rotating b^i_{jl} by 90 degrees counterclockwise. See Fig. 6 as an illustration.

After node i knows (q_i, q_j, q_l), it can simply choose w_{ij} and w_{il} to satisfy

$$w_{ij}(q_j - q_i) + w_{il}(q_l - q_i) = 0.$$

Due to (1), the same complex weights also make the following holds

$$w_{ij}(p_j - p_i) + w_{il}(p_l - p_i) = 0.$$

Step A3: Compute complex weights regarding more than two neighbors.

For the case that node i has more than two neighbors, we choose a combination of two neighbors from $\bar{\mathcal{N}}_i$, say j and l. If j and l are mutual neighbors, and i, j and l are not collinear, then according to Step A2, we are able to find complex weights $w^{(r)}_{ij}$ and $w^{(r)}_{il}$ to satisfy

$$w^{(r)}_{ij}(p_j - p_i) + w^{(r)}_{il}(p_l - p_i) = 0$$

where r enumerates the possible combinations of any two neighbors from $\bar{\mathcal{N}}_i$. Then we take a random complex coefficient γ_r for each r to combine all these representations, that is,

$$w'_{ij} = \sum_r \gamma_r w^{(r)}_{ij} \text{ for } j \in \bar{\mathcal{N}}_i.$$

At last, node i normalizes the complex weights. That is,

$$w'_{ii} = -\sum_{j\in\bar{\mathcal{N}}_i} w'_{ij},\ w_{ij} = w'_{ij} / \sum_{k=1}^{n} |w'_{ik}| \text{ for any } j.$$

Thus, the complex weights $w_{ij},\ j \in \bar{\mathcal{N}}_i$, are found to satisfy

$$\sum_{j\in\bar{\mathcal{N}}_i} w_{ij}(p_j - p_i) = 0. \tag{5}$$

We summarize the above results to give an algorithm of computing complex weights w_{ij} for node i. The pseudo code is given in Algorithm 1.

Remark 3.1: Each node executes Algorithm 1 in a distributed way. That is, each node only requires its own bearing measurements about its neighbors and some bearing measurement information from its neighbors via communication. It is worth to emphasize that Algorithm 1 works for all networks. ■

Remark 3.2: According to the normalization step we know that $\sum_{j=1}^{n} |w_{ij}| = 1$, which ensures the infinity norm of the resulted matrix to be one. ■

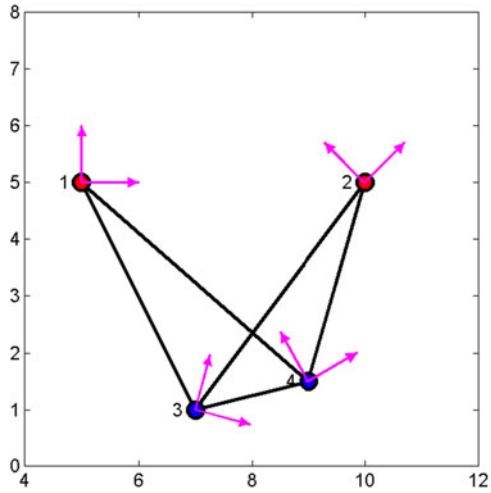


Fig. 7. A simple example with graph $\bar{\mathcal{G}}$.

Algorithm 1: Compute Complex Weights w_{ij} for Node i.

Input: The measurement b^i_{ij} for $j \in \bar{\mathcal{N}}_i$; The measurement b^j_{jl} for $j, l \in \bar{\mathcal{N}}_i$ and $(l, j) \in \bar{\mathcal{E}}$.

Output: $w_{ij}, j \in \bar{\mathcal{N}}_i$.

Set $m = 0$;
Set $w^{(m)}_{ij} = 0$ for $j \in \bar{\mathcal{N}}_i$;
while there is still a combination of two neighbors in $\bar{\mathcal{N}}_i$ that have not been selected before **do**
  Choose a combination of two neighbors from $\bar{\mathcal{N}}_i$, say j and l;
  if node j and l are mutual neighbors, and i, j and l are not collinear **then**
    (1) Update $m := m + 1$;
    (2) Compute the weights $w^{(m)}_{ij}, w^{(m)}_{il}$ of i with respect to j and l based on Steps A1-A2;
    (3) Set $w^{(m)}_{is} = 0$ for $s \in \bar{\mathcal{N}}_i - \{j, l\}$;
  end if
end while
Choose random complex coefficients $\gamma_1, \ldots, \gamma_m$;
Compute $w'_{ij} = \sum_{r=1}^{m} \gamma_r w^{(r)}_{ij}$ for $j \in \bar{\mathcal{N}}_i$;
Compute $w'_{ii} = -\sum_{j\in\bar{\mathcal{N}}_i} w'_{ij}$;
return $w_{ij} = w'_{ij} / \sum_{k=1}^{n} |w'_{ik}|$ for any $j \in \bar{\mathcal{N}}_i$.

In the following, we use a simple example in Fig. 7 to demonstrate how Algorithm 1 calculates the complex weights. In the example, the nodes in red are anchors while the nodes in blue are sensor nodes. Each node has its own local coordinate system as shown in the figure. The absolute coordinates in this case are

$$p_1 = 5 + 5\iota,\ p_2 = 10 + 5\iota,\ p_3 = 7 + \iota,\ p_4 = 9 + 1.5\iota.$$

We take node 3 for example. Notice that nodes $1, 4 \in \bar{\mathcal{N}}_3$ and they are mutual neighbors. Then node 3 can compute w^1_{31} and w^1_{34} based on Steps A1-A2. To be specific, node 3 measures

$$b^3_{31} = -0.6635 + 0.7482\iota,\ b^3_{34} = 0.8743 + 0.4854\iota$$

and obtain the following bearing information via communication

$$b^1_{13} = 0.4472 - 0.8944\iota,\ b^1_{14} = 0.7526 - 0.6585\iota,$$
$$b^4_{41} = -0.3225 + 0.9466\iota,\ b^4_{43} = -0.9614 + 0.2750\iota.$$

The next step is to convert the received bearing information to the ones in node 3's local frame. For example, b^1_{13} is converted into b^3_{13} according to the following formula:

$$b^3_{13} = \frac{b^3_{31}}{b^1_{13}} e^{\iota\pi} b^1_{13} = 0.6635 - 0.7482\iota.$$

Similar, all the others can be obtained in the same way. That is,

$$b^3_{14} = 0.8974 - 0.4413\iota,$$
$$b^3_{41} = -0.8974 + 0.4413\iota,$$
$$b^3_{43} = -0.8743 - 0.4854\iota.$$

Then we construct a similar configuration by setting $q_3 = 0$ and $q_1 = b^3_{31} = -0.6635 + 0.7482\iota$ and by solving q_4 from the following formula

$$\begin{cases} \mathrm{Re}[\iota b^3_{34}(\overline{q_4 - q_3})] = 0 \\ \mathrm{Re}[\iota b^3_{14}(\overline{q_4 - q_1})] = 0 \end{cases}$$

By solving the above equations, we get $q_4 = 0.4030 + 0.2237\iota$. Next, node 3 choose w^1_{31} and w^1_{34} to satisfy

$$w^1_{31}(q_1 - q_3) + w^1_{34}(q_4 - q_3) = 0.$$

That is,

$$w^1_{31} = -0.1000 - 0.2000\iota,\ w^1_{34} = -0.4706 + 0.1176\iota.$$

Moreover, since $2, 4 \in \bar{\mathcal{N}}_3$ and they are mutual neighbors, node 3 can also obtain

$$w^2_{32} = 0.1200 - 0.1600\iota,\ w^2_{34} = -0.4706 + 0.1176\iota.$$

Choose $\gamma_1 = 0.5$ and $\gamma_2 = 0.5$. Then

$$w'_{31} = \gamma_1 w^1_{31} = -0.0500 - 0.1000\iota,$$
$$w'_{32} = \gamma_2 w^2_{32} = 0.0600 - 0.0800\iota,$$
$$w'_{34} = \gamma_1 w^1_{34} + \gamma_2 w^2_{34} = -0.4706 + 0.1176\iota.$$

and

$$w'_{33} = -w'_{31} - w'_{32} - w'_{34} = 0.4606 + 0.0624i.$$

At last, node 3 normalizes the complex weights as

$$w_{31} = -0.0430 - 0.0861\iota,$$
$$w_{32} = 0.0517 - 0.0689\iota,$$
$$w_{34} = -0.4051 + 0.1013\iota.$$

Remark 3.3: Notice that for almost all randomly chosen complex coefficients $\gamma_1, \ldots, \gamma_m$, $w^{(r)}_{ij} \neq 0$ for some r implies that $w_{ij} \neq 0$. ■

From Algorithm 1 we know that if nodes i and j ($j \in \bar{\mathcal{N}}_i$) do not have a common neighbor, then w_{ij} will be zero. We then consider a new graph $\mathcal{G}(k) = (\mathcal{V}, \mathcal{E}(k))$, where $\mathcal{E}(k)$ is a subset of $\bar{\mathcal{E}}(k)$, such that $(j, i) \in \mathcal{E}(k)$ if and only if $w_{ij} \neq 0$. Indeed, $(j, i) \in \mathcal{E}(k)$ if and only if node i and node j have at least one common neighbor in $\bar{\mathcal{G}}(k)$. Let $\mathcal{N}_i(k)$ be the set of the neighbors of node i in $\mathcal{G}(k)$.

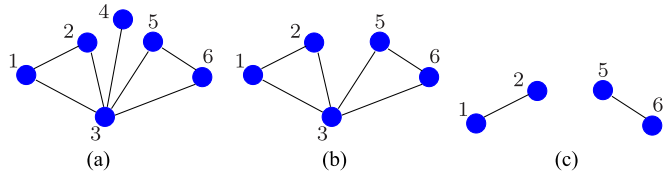


Fig. 8. (a) Graph $\bar{\mathcal{G}}$. (b) Graph $\mathcal{G}$. (c) Graph Π_3.

We rewrite (5) into the matrix form as

$$L(k)p = 0,$$

where $L(k)$ is the complex Laplacian with the complex weights chosen by Algorithm 1. Then we obtain the following complex linear representation regarding the coordinates in Σ_g of all the sensor nodes

$$L(k)p = \left[\begin{array}{c|c} 0 & 0 \\ \hline L_a(k) & L_s(k) \end{array}\right] \begin{bmatrix} p_a \\ p_s \end{bmatrix} = 0, \tag{6}$$

which leads to

$$L_a(k)p_a + L_s(k)p_s = 0. \tag{7}$$

Notice that there is an edge (j, i), $i \neq j$, in $\mathcal{G}(k)$ at k if and only if the (i, j)th entry of $L(k)$, i.e., $-w_{ij}(k)$, is nonzero at time instant k.

Moreover, we introduce the *node-i-neighbor-induced graph* $\Pi_i(k)$ for each sensor node i, which is the $\mathcal{N}_i(k)$-induced graph in $\mathcal{G}(k)$, i.e., $\Pi_i(k) = (\mathcal{N}_i(k), \mathcal{E}(k) \cap \mathcal{N}_i(k) \times \mathcal{N}_i(k))$.

Take Fig. 8 as an example. A graph $\bar{\mathcal{G}}$ is shown in Fig. 8(a) with $\bar{\mathcal{N}}_3 = \{1, 2, 4, 5, 6\}$. By Algorithm 1, graph $\mathcal{G}$ is constructed in Fig. 8(b) with $\mathcal{N}_3 = \{1, 2, 5, 6\}$. As a result, Π_3 is presented in Fig. 8(c).

However, assume that no sensor can act as a centralized node to obtain all the information in (7). Then a distributed algorithm is needed for each sensor node to solve (7) in parallel, which will be developed in next section. The convergence of the distributed algorithm depends on $\mathcal{G}(k)$ and $\Pi_i(k)$.

IV. Distributed Algorithm for Fixed Topologies

To make it clearer for understanding the basic idea, in this section we treat fixed topologies firstly.

A. Localizability Condition

Considering fixed topologies, the equality (7) becomes

$$L_a p_a + L_s p_s = 0, \tag{8}$$

which indicates that p_s can be solved uniquely if and only if L_s is nonsingular, i.e.,

$$p_s = -L_s^{-1} L_a p_a.$$

Then it results in two problems. One is that under what graphical conditions, L_s is nonsingular. The other is how to solve (8) in a distributed manner.

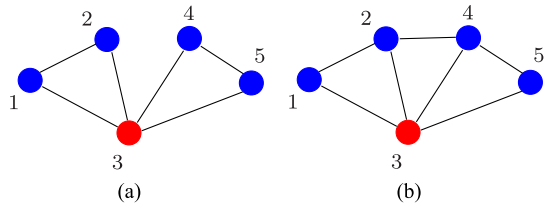


Fig. 9. An example to show that $L(3,:)$ may not span an (n_3-1)-dimensional linear subspace.

We first present a sufficient graphical condition such that L_s is nonsingular.

Theorem 4.1: For almost all randomly chosen complex coefficients $\gamma_1,\ldots,\gamma_m$ in Algorithm 1, the sensor network is localizable by solving (8) if the following two conditions hold

1) every sensor node is 2-reachable from $\mathcal{A}$ in $\mathcal{G}$,
2) the node-i-neighbor-induced graph Π_i is connected for every sensor node i.

The proof of Theorem 4.1 needs several lemmas. We denote by $\mathcal{L}(\mathcal{G})$ the set of all Laplacian matrices with nonzero weights on the edges in $\mathcal{G}$. Let $\mathcal{R}$ be a subset of $\mathcal{V}$ and let $L_{\mathcal{R}}$ be the submatrix of $L\in\mathcal{L}(\mathcal{G})$ with the rows and columns corresponding to nodes in $\mathcal{R}$ crossed out. The following lemma provides the relationship between the determinant of $L_{\mathcal{R}}$ and the connectivity of $\mathcal{G}$.

Lemma 4.1: Consider $\xi\in\mathbb{C}^n$ satisfying that $\xi_i\neq\xi_j,\ \forall i,j$. The following two are equivalent.

1) Let $\mathcal{R}=\{1,2\}$ and every node in $\mathcal{V}-\mathcal{R}$ is 2-reachable from $\mathcal{R}$.
2) For almost all[1] $L\in\{L\in\mathcal{L}(\mathcal{G}):L\xi=0\}$, the determinant of $L_{\mathcal{R}}$ is distinct from zero.

The proof of Lemma 4.1 is given in the Appendix.

Remark 4.1: We now give more explanations about the "almost all" property in Lemma 4.1. Denote $n_i=|\mathcal{N}_i|$ and denote the i-th row of L by $L(i,:)$. We can know that there are n_i+1 nonzero entries in $L(i,:)$. Also with the fact that $L(i,:)\mathbf{1}=0$ and $L(i,:)\xi=0$, it is known that $L(i,:)$ can span an (n_i-1)-dimensional linear subspace. This means that if $L(i,:)$ is randomly taken from the (n_i-1)-dimensional linear subspace for all i, then the "almost all" property in Lemma 4.1 implies that for such an L, the determinant of $L_{\mathcal{R}}$ must not be zero in probability one. However, if $L(i,:)$ can only be taken from a subspace of dimension less than n_i-1, then the "almost all" property in Lemma 4.1 may not guarantee that such an L has non-zero $\det(L_{\mathcal{R}})$. An example is given in Fig. 9. For the case in Fig. 9(a), by Algorithm 1 we can obtain

$$L(3,:)=\begin{bmatrix}\gamma_1 w_{31}^{(1)} & \gamma_1 w_{32}^{(1)} & * & \gamma_2 w_{34}^{(2)} & \gamma_2 w_{35}^{(2)}\end{bmatrix},$$

which lies in a 2-dimensional linear subspace. For the case in Fig. 9(b), similarly, we get

$$L(3,:)=\begin{bmatrix}\gamma_1 w_{31}^{(1)} & \gamma_1 w_{32}^{(1)}+\gamma_2 w_{32}^{(2)} & * & \gamma_2 w_{34}^{(2)}+\gamma_3 w_{34}^{(3)} & \gamma_3 w_{35}^{(3)}\end{bmatrix},$$

which lies in a 3-dimensional linear subspace. So whether $L(i,:)$'s constructed by Algorithm 1 span (n_i-1)-dimensional linear subspace depends on the connectivity of Π_i. ■

The following lemma describes the relationship between the node-i-neighbor-induced graph Π_i and the dimension of the linear subspace formed by $L(i,:)$ with all possible weights chosen by Algorithm 1.

Lemma 4.2: With the weights computed by Algorithm 1, $L(i,:)$ can span an (n_i-1)-dimensional linear subspace if Π_i is connected.

The proof of Lemma 4.2 is given in the Appendix.

We are now ready to present the proof for Theorem 4.1.

Proof of Theorem 4.1: Suppose that in $\mathcal{G}$, every sensor node is 2-reachable from $\mathcal{A}$, and Π_i is connected for all i. It follows from Lemma 4.1, Remark 4.1 and Lemma 4.2 that

$$\det(L_s)\neq 0.$$

Hence, p_s can be solved uniquely from (8). ■

Remark 4.2: The graphical condition in Theorem 4.1 also implies infinitesimal bearing rigidity. This is because infinitesimal bearing rigidity is necessary and sufficient with global bearing measurements [17]. As shown in Section II, compared with global bearing measurements, more edges should be added to the graph such that the network can be localizable with only local bearing measurements. ■

B. Distributed Algorithm

In this subsection, we will propose a distributed algorithm for each sensor node to solve (8).

Let w_{ij}'s be the complex weights designed in Section III and let $\bar{w}_{ij}$ be the conjugate of the complex number w_{ij}. We propose the following iterative algorithm for each sensor node $i=3,\ldots,n$,

$$\hat{p}_i(k+1)=\hat{p}_i(k)-\epsilon\sum_{j\in\mathcal{N}_i}\bar{w}_{ij}\theta_i(k)+\epsilon\sum_{j\in\mathcal{N}_i}\bar{w}_{ji}\theta_j(k) \quad (9)$$

where $\hat{p}_i$ is the estimate of p_i for node i and

$$\theta_i(k)=-\sum_{j\in\mathcal{N}_i}w_{ij}(\hat{p}_j(k)-\hat{p}_i(k)).$$

In the above iterative update, ϵ is a positive scalar that satisfies

$$0<\epsilon<2/\lambda_{\max}(L_s^H L_s),$$

where $\lambda_{\max}(L_s^H L_s)$ represents the maximum eigenvalue of $L_s^H L_s$.

The following theorem shows that $\hat{p}_i$ will converge to p_i.

Theorem 4.2: Suppose that L_s is nonsingular. Under distributed algorithm (9), every $\hat{p}_i(k)$ is exponentially convergent to p_i.

Proof: Denote by $\hat{p}$ the aggregated vectors of all $\hat{p}_i$'s. We can write (9) in the matrix form as

$$\hat{p}_s(k+1)=\left(I-\epsilon L_s^H L_s\right)\hat{p}_s(k)-\epsilon L_s^H L_a p_a. \quad (10)$$

[1]Here "for almost all" parameter values is to be understood as "for all parameter values except for those in some proper algebraic variety in the parameter space". The proper algebraic variety for which a property is not true is the zero set of some nontrivial polynomial with real coefficients in the parameters. A proper algebraic variety has Lebesgue measure zero [24]. Here "for almost all L" means "for almost all weights used to construct L".

Algorithm 2: Distributed Implementation for the Iteration (9).

1: Each node i computes complex weights w_{ij} for $j \in \bar{\mathcal{N}}_i$;
2: Each node i transmits w_{ij} to its neighbors $j \in \mathcal{N}_i$;
3: Each node i receives w_{ji} from its neighbors $j \in \mathcal{N}_i$ and then computes $l_i := \sum_{j\in\mathcal{N}_i} |w_{ji}|$;
4: Each node i computes $\|L_s\|_1 = \max_i l_i$ by the maximum consensus algorithm;
5: Each node i chooses the parameter $\epsilon = 1/\|L_s\|_1$;
6: Each node i transmits its own estimate $\hat{p}_i$ to its neighbors $j \in \mathcal{N}_i$;
7: Each node i receives $\hat{p}_j$ from its neighbors $j \in \mathcal{N}_i$ and then computes $\theta_i = -\sum_{j\in\mathcal{N}_i} w_{ij}(\hat{p}_j - \hat{p}_i)$;
8: Each node i transmits $\bar{w}_{ij}\theta_i$ to its neighbors $j \in \mathcal{N}_i$;
9: Each node i receives $\bar{w}_{ji}\theta_j$ from its neighbors $j \in \mathcal{N}_i$ and then updates $\hat{p}_i = \hat{p}_i - \epsilon \sum_{j\in\mathcal{N}_i(k)} \bar{w}_{ij}\theta_i + \epsilon \sum_{j\in\mathcal{N}_i} \bar{w}_{ji}\theta_j$;
10: Go back to 6.

Do the coordinate transformation $\tilde{p}_s(k) = \hat{p}_s(k) - p_s$ and apply the fact that $\epsilon L_s^H L_a p_a = -\epsilon L_s^H L_s p_s$. We derive that

$$\tilde{p}_s(k+1) = \left(I - \epsilon L_s^H L_s\right)\tilde{p}_s(k). \tag{11}$$

Since L_s is nonsingular, $L_s^H L_s$ is positive definite and the eigenvalues of $L_s^H L_s$ are all real. It follows from $0 < \epsilon < 2/\lambda_{\max}(L_s^H L_s)$ that all the eigenvalues of $I - \epsilon L_s^H L_s$ lie in the open unit disk. Thus, every $\hat{p}_i(k)$ is exponentially convergent to p_i. ■

In what follows, we give the distributed implementation of the iteration (9). Notice that $\lambda_{\max}(L_s^H L_s) \le \|L_s\|_1 \|L_s\|_\infty$. By Remark 3.2 we know that $\|L_s\|_\infty \le 1$. Thus, it remains to choose ϵ satisfying $\epsilon < 2/\|L_s\|_1$. Since each sensor node i is able to obtain the entries in the i-th column of L_s from its neighbors through communication, $\|L_s\|_1$ can be computed for each node in finite time by the maximum consensus algorithm [25]. Then the iteration (9) can be implemented in a distributed manner as described in Algorithm 2.

C. Sensitivity Analysis

In this subsection, we will present sensitivity analysis for the proposed localization scheme.

Considering the noises in the bearing measurements, the localization algorithm (10) becomes

$$\hat{p}_s(k+1) = \left(I - \epsilon \hat{L}_s^H \hat{L}_s\right)\hat{p}_s(k) - \epsilon \hat{L}_s^H \hat{L}_a p_a \tag{12}$$

where $\hat{L}_a$ and $\hat{L}_s$ are the corresponding matrices obtained according to Algorithm 1. Denote Δ_a and Δ_s the error matrices due to the measurement noises and write $\hat{L}_a = L_a + \Delta_a$ and $\hat{L}_s = L_s + \Delta_s$.

It is true that if $\hat{L}_s$ is nonsingular, then the final estimate of (12) is given by

$$\hat{p}_s^* = -\hat{L}_s^{-1}\hat{L}_a p_a.$$

We then give an upper bound for $\|\Delta_s\|$ to ensure the nonsingularity of $\hat{L}_s$.

Theorem 4.3: Suppose that L_s is nonsingular. The matrix $\hat{L}_s$ is nonsingular if Δ_s satisfies

$$\|\Delta_s\| < \sqrt{\lambda_{\min}(L_s^H L_s)},$$

where $\lambda_{\min}(L_s^H L_s)$ represents the smallest eigenvalue of $L_s^H L_s$.

Proof: Notice that $\|\Delta_s\| < \sqrt{\lambda_{\min}(L_s^H L_s)} = 1/\|L_s^{-1}\|$, which further implies $\|L_s^{-1}\Delta_s\| \le \|L_s^{-1}\|\|\Delta_s\| < 1$. Thus the matrix $I + L_s^{-1}\Delta_s$ is nonsingular. Hence, $\hat{L}_s = L_s + \Delta_s = L_s(I + L_s^{-1}\Delta_s)$ is nonsingular. ■

V. Distributed Algorithm for Switching Topologies

In this section we consider the distributed localization problem under switching topologies.

A. Localizability Condition

Recall the linear equation constraints (7), i.e.,

$$L_a(k)p_a + L_s(k)p_s = 0.$$

Although $L_s(k)$ may change over time and may become singular at some time, it is still possible for each node to solve (7) if the sum of $L_s(k)$ across a time interval is nonsingular. To be specific, from

$$\sum_{k=1}^{m} L_a(k)p_a + \sum_{k=1}^{m} L_s(k)p_s = 0$$

we can attain

$$p_s = -\left(\sum_{k=1}^{m} L_s(k)\right)^{-1} \sum_{k=1}^{m} L_a(k)p_a.$$

However, this property should be persistently satisfied over the whole time horizon. From this perspective we present the following sufficient graphical condition, under which the sensor network is localizable.

Theorem 5.1: For almost all randomly chosen complex coefficients $\gamma_1, \ldots, \gamma_m$ in Algorithm 1, the sensor network is localizable by solving (7) if the following two conditions hold

1) every sensor node is jointly 2-reachable from $\mathcal{A}$ in $\mathcal{G}(k)$,
2) the node-i-neighbor-induced graph $\Pi_i(k)$ is jointly connected for every sensor node i.

The proof of Theorem 5.1 is given in the appendix.

B. Distributed Algorithm

To solve (7) in a distributed way, we consider the following iterative algorithm for each sensor node $i = 3, \ldots, n$,

$$\hat{p}_i(k+1) = \hat{p}_i(k) - \epsilon(k) \sum_{j\in\mathcal{N}_i(k)} \bar{w}_{ij}(k)\theta_i(k) + \epsilon(k) \sum_{j\in\mathcal{N}_i(k)} \bar{w}_{ji}(k)\theta_j(k) \tag{13}$$

Algorithm 3: Distributed Implementation for the Iteration (13).

1: Each node i computes complex weights $w_{ij}^{(c)}$ for $j \in \bar{\mathcal{N}}_i^{(c)}$;
2: Each node i transmits $r_{ij} := \max_c |w_{ij}^{(c)}|$ to its neighbors $j \in \mathcal{N}_i$;
3: Each node i receives r_{ji} from its neighbors $j \in \mathcal{N}_i$ and then computes $l_i := \sum_{j\in\mathcal{N}_i} r_{ji}$;
4: Each node i computes $\max_i l_i$ by the maximum consensus algorithm;
5: Each node i chooses the parameter $\epsilon = 1/\max_i l_i$;
6: Each node i selects complex weights $w_{ij}(k)$ at each k by the look-up table method;
7: Each node i transmits its own estimate $\hat{p}_i$ to its neighbors $j \in \mathcal{N}_i(k)$;
8: Each node i receives $\hat{p}_j$ from its neighbors $j \in \mathcal{N}_i(k)$ and then computes $\theta_i = -\sum_{j\in\mathcal{N}_i(k)} w_{ij}(k)(\hat{p}_j - \hat{p}_i)$;
9: Each node i transmits $\bar{w}_{ij}(k)\theta_i$ to its neighbors $j \in \mathcal{N}_i(k)$;
10: Each node i receives $\bar{w}_{ji}(k)\theta_j$ from its neighbors $j \in \mathcal{N}_i(k)$ and then updates $\hat{p}_i = \hat{p}_i - \epsilon \sum_{j\in\mathcal{N}_i(k)} \bar{w}_{ij}(k)\theta_i + \epsilon \sum_{j\in\mathcal{N}_i(k)} \bar{w}_{ji}(k)\theta_j$;
11: Go back to 6.

where

$$\theta_i(k) = -\sum_{j\in\mathcal{N}_i(k)} w_{ij}(k)(\hat{p}_j(k) - \hat{p}_i(k)).$$

In the above iterative update, $\epsilon(k)$ is a positive scalar that satisfies

$$0 < \epsilon(k) \le 1/\lambda_{\max}(L_s^H(k)L_s(k)).$$

Next we present the convergence analysis for the proposed algorithm (13).

Theorem 5.2: For almost all randomly chosen complex coefficients $\gamma_1, \ldots, \gamma_m$ in Algorithm 1, under the distributed algorithm (13), every $\hat{p}_i(k)$ is exponentially convergent to p_i if the graphical conditions in Theorem 5.1 hold.

The proof of Theorem 5.2 is given in the Appendix.

Next we consider the distributed implementation of the iteration (13). Note that switching topologies are used to model possible link failures. Thus, each node i is able to know all its possible sub-neighbor-set $\bar{\mathcal{N}}_i(k) \subset \bar{\mathcal{N}}_i$, based on which each node can compute the complex weights for each combination of its neighbors, denoted by $\bar{\mathcal{N}}_i^{(c)}$ where c enumerates all the possible sub-neighbor-set. For each possible sub-neighbor-set, node i can calculate $w_{ij}^{(c)}$ for $j \in \bar{\mathcal{N}}_i^{(c)}$. Then each node takes $\epsilon(k)$ the same value ϵ that satisfies $0 < \epsilon \le 1/\lambda_{\max}(L_s^H(k)L_s(k))$ for all k. To be specific, the implementation of (13) in a distributed way is described in Algorithm 3.

Remark 5.1: Here we give a brief discussion for sensitivity analysis of Algorithm (13). Notice that

$$\hat{p}_s(k+1) = \left(I - \epsilon\hat{L}_s^H(k)\hat{L}_s(k)\right)\hat{p}_s(k) - \epsilon\hat{L}_s^H(k)\hat{L}_a(k)p_a$$

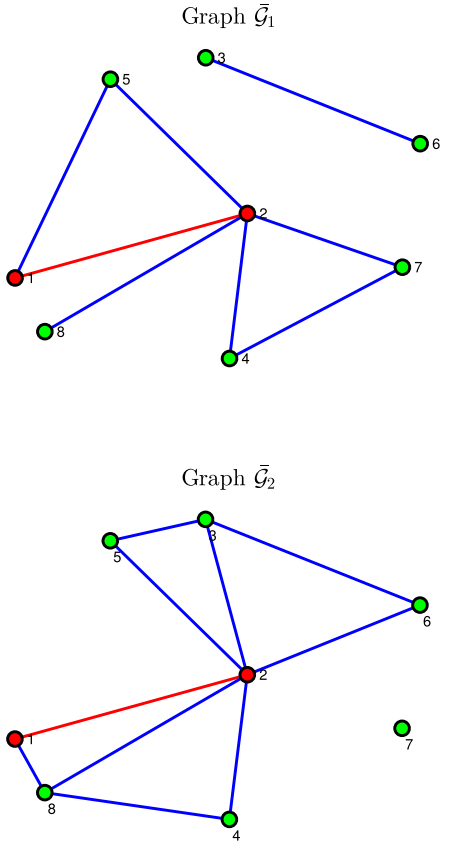


Fig. 10. A periodic switching graph $\bar{\mathcal{G}}(k)$ that switches between $\bar{\mathcal{G}}_1$ and $\bar{\mathcal{G}}_2$.

where $\hat{L}_a(k) = L_a(k) + \Delta_a(k)$ and $\hat{L}_s(k) = L_s(k) + \Delta_s(k)$. When measurement noises exist, the resulting switching system may have multiple equilibria. Thus, the error system can be written as

$$\tilde{p}_s(k+1) = \left(I - \epsilon\hat{L}_s^H(k)\hat{L}_s(k)\right)\tilde{p}_s(k) + \epsilon\hat{L}_s^H(k)\hat{L}_s(k)\tilde{p}_s^*(k),$$

where $\tilde{p}_s = \hat{p}_s - p_s$ and $\tilde{p}_s^* = p_s^* - p_s$. Moreover, $p_s^*(k)$ satisfies $\hat{L}_a(k)p_a + \hat{L}_s(k)p_s^*(k) = 0$. Note that $\epsilon\hat{L}_s^H(k)\hat{L}_s(k)\tilde{p}_s^*(k)$ is bounded and the system

$$\tilde{p}_s(k+1) = \left(I - \epsilon\hat{L}_s^H(k)\hat{L}_s(k)\right)\tilde{p}_s(k),$$

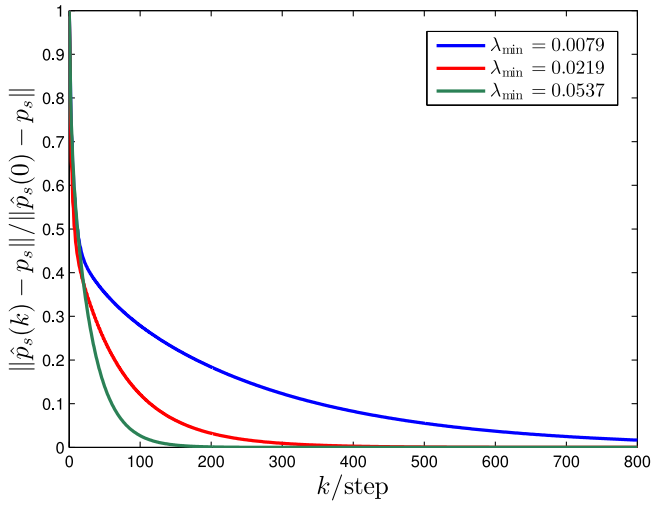


Fig. 11. Comparison of convergence speed with $\lambda_{\min}(\epsilon L_{s1}^H L_{s1} + \epsilon L_{s2}^H L_{s2}) = 0.0079, 0.0219$ and 0.0537, which shows that larger $\lambda_{\min}(\epsilon L_{s1}^H L_{s1} + \epsilon L_{s2}^H L_{s2})$ indicates faster convergence.

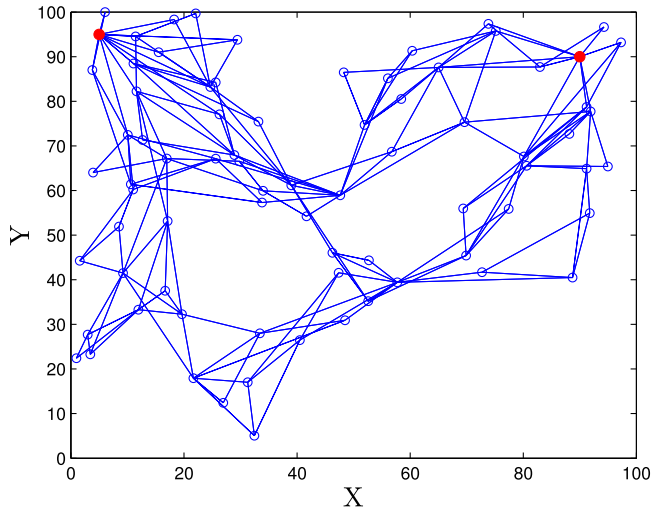


Fig. 12. A configuration of sensor network of 2 anchor nodes and 75 sensor nodes, where the anchor nodes are represented by red circles while the others are sensor nodes. The lines between pairs of nodes indicate that the bearing measurements are available for these pairs.

is exponentially stable. Then $\tilde{p}_s$ will converge to a region that contains all $\tilde{p}_s^*(k)$. That is to say, the estimates will converge to a region nearby the true locations. ■

VI. Simulation

In this section, we provide two simulation results to validate our results.

A. Performance Influenced by Complex Weights

In this subsection, we use a toy example to demonstrate the effect of different complex weights on the performance of the algorithm. Consider a network consisting of 8 nodes. The anchor nodes are labeled by 1 and 2. The remaining nodes are

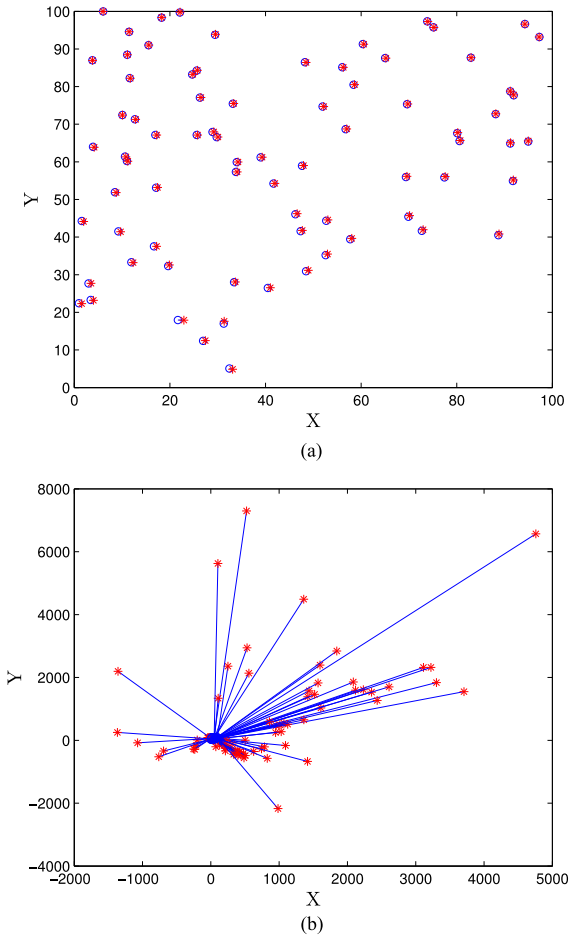


Fig. 13. (a) The estimates obtained from the algorithm (9) perfectly matches the true positions, where the red stars represent the estimates while the blue circles are the true locations. (b) The estimates obtained by the algorithm in [17] do not match the true positions as the algorithm may diverge when the local frames do not have a common orientation as the global one.

sensor nodes to be localized. In the simulation, suppose that the absolute coordinates for the nodes are

$$p = [4 + 35\iota,\ 43 + 47\iota,\ 36 + 76\iota,\ 40 + 20\iota,\ 20 + 72\iota,\ 72 + 60\iota,\ 69 + 37\iota,\ 9 + 25\iota]^T,$$

and consider the graph $\bar{\mathcal{G}}(k)$ that switches between $\bar{\mathcal{G}}_1$ and $\bar{\mathcal{G}}_2$ as shown in Fig. 10.

By taking $K = 2$, it is known that $\mathcal{G}([k, k+K)) = \mathcal{G}_1 \cup \mathcal{G}_2$, and it can be checked that every sensor node is jointly 2-reachable from $\mathcal{A}$ in $\mathcal{G}_1 \cup \mathcal{G}_2$. It also can be checked that $\Pi_i(k)$ is jointly connected with $K = 2$.

Denote L_{s1} and L_{s2} the computed matrices corresponding to $\mathcal{G}_1$ and $\mathcal{G}_2$, respectively. Notice that both L_{s1} and L_{s2} are singular, which means that for each subsystem the estimate will not converge to the true absolute coordinate. However, if two subsystems switch between each other, then under the proposed algorithm the estimate eventually converges since $L_{s1} + L_{s2}$ is nonsingular. Next we carry out a simulation to demonstrate the effect of the eigenvalues of $\epsilon L_{s1}^H L_{s1} + \epsilon L_{s2}^H L_{s2}$ on the convergence speed. A simulation result is shown in

Fig. 11, for which three parameters with $\lambda_{\min}(\epsilon L_{s1}^H L_{s1} + \epsilon L_{s2}^H L_{s2}) = 0.0079, 0.0219$ and 0.0537 are considered. The estimation error ratio $\|\hat{p}_s(k) - p_s\|/\|\hat{p}_s(0) - p_s\|$ is plotted for each case. From the simulation result we can see that larger $\lambda_{\min}(\epsilon L_{s1}^H L_{s1} + \epsilon L_{s2}^H L_{s2})$ indicates faster convergence.

B. Comparison with Localization Algorithms Needing the Global Frame

In this subsection, we compare the proposed algorithm (9) with the approach in [17].

Consider a large network with 2 anchor nodes and 75 sensor nodes. Each node holds its local coordinate system. Consider a graph $\bar{\mathcal{G}}$, for which an edge in $\bar{\mathcal{E}}$ indicates that the corresponding two nodes can obtain the bearing measurements and communication with each other. The configuration is plotted in Fig. 12, in which the anchor nodes are represented by red cycles while the others are sensor nodes. The lines between pairs of nodes in the figure indicate that the bearing measurements are available for these pairs. Each sensor node computes the complex weights w_{ij}'s by Algorithm 1 and a new graph $\mathcal{G}$ is resulted, in which every sensor node is 2-reachable from the anchor set, and Π_i is connected.

A simulation result with the estimates of sensor locations obtained is shown in Fig. 13. The red stars represent the estimates while the blue circles are the true locations. Fig. 13(a) shows that the estimates perfectly match the true positions of the sensor nodes by algorithm (9). However, the algorithm in [17] diverges for this simulated scenario as we can see from Fig. 13(b) since a key assumption in [17] is that the orientations of all the local coordinate systems should be consistent with the orientation of the global coordinate system.

VII. Conclusion

This paper studies the distributed localization problem for a large sensor network in the plane with bearing-only measurements. In this paper, we utilize complex numbers to represent node i's estimate $\hat{p}_i$ and the coordinate p_i in the global frame instead of two-dimensional vectors. The main idea is that each node i drives $\sum_{j\in\mathcal{N}_i} w_{ij}(\hat{p}_j - \hat{p}_i)$ to zero, where the complex weights w_{ij}'s can be computed in node i's local coordinate system by constructing a similar framework. In this way, the global frame is no longer required. Furthermore, by using complex weights we attain the complex linear representation concerning the absolute locations for the sensor network. However, for the complex Laplacian L, in general w_{ij} is not equal to $\bar{w}_{ji}$, i.e., L is not Hermitian. Also the value w_{ij} will change over time due to switching topologies. To overcome the difficulty in designing the distributed algorithm and analyzing the convergence, we introduce an auxiliary state and an idea similar to persistent excitation is developed to show the convergence of the algorithm under switching topologies. Moreover, graphical conditions ensuring globally exponential convergence is obtained.

Many interesting problems deserve further investigation. One problem is to design localization distributed algorithm for sensing networks that only have unidirectional sensing and communication capability. Under such a setup, two issues need to be taken into consideration. First, for each node, how to construct similar frameworks to compute complex weights. Second, to acquire a Hermitian matrix, auxiliary states θ_j from i's out-neighbors j need to be transmitted back to i. This paper considers undirected topologies, i.e., neighbors are also out-neighbors. However, for unidirectional case, θ_j cannot be transmitted directly from j to i. So extra edges may need for information exchange and the graphical conditions ensuring global convergence will be different. Another problem is to consider noisy measurements. Belief propagation based algorithms can take into account uncertainty of the measurements and estimate the posterior probability density function of the positions of all unknown nodes [26], [27]. It may become feasible by combining the method in this paper with BP based algorithms such that the localization problem can be solved in the probability framework.

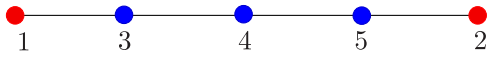


Fig. 14. A path graph with with its terminal nodes labeled as 1 and 2.

Appendix

We introduce a lemma before proving Lemma 4.1.

Lemma A.1 ([28]): Consider a framework $(\mathcal{G}, \xi)$, where $\mathcal{G}$ is a path graph of n nodes with its terminal nodes labeled as 1 and 2 (Fig. 14). If $\xi_i \neq \xi_j$ for $i \neq j$, then there exists a complex Laplacian matrix $L \in \mathcal{L}(\mathcal{G})$

$$\left[\begin{array}{c|c} A_{2\times 2} & B_{2\times(n-2)} \\ \hline C_{(n-2)\times 2} & D_{(n-2)\times(n-2)} \end{array}\right]$$

such that D is of rank $n-2$.

Proof of Lemma 4.1: $(1) \Longrightarrow (2)$ For a graph $\mathcal{G}$, denote the subset by $\mathcal{R}_0 := \mathcal{R} = \{1, 2\}$ and every other node is 2-reachable from $\mathcal{R}_0$. Choose a node i not in $\mathcal{R}_0$ and then we can find two disjoint paths from 1 to i and from 2 to i. Denote the set of nodes in these two paths excluding the nodes in $\mathcal{R}_0$ by $\mathcal{R}_1$ and let $n_1 = |\mathcal{R}_1|$. Relabel the nodes in $\mathcal{R}_1$ from 3 to $n_1 + 2$. Next we choose another node $j \neq i$ not in $\mathcal{R}_0 \cup \mathcal{R}_1$. Also, there must exist two disjoint paths from two different nodes in $\mathcal{R}_0 \cup \mathcal{R}_1$ to node j, for which only the two terminal nodes are in $\mathcal{R}_0 \cup \mathcal{R}_1$. Denote the set of nodes in these two paths excluding the nodes in $\mathcal{R}_0 \cup \mathcal{R}_1$ by $\mathcal{R}_2$ and let $n_2 = |\mathcal{R}_2|$. Then relabel these nodes from $n_1 + 3$ to $n_1 + n_2 + 2$. Repeat the procedure until all the nodes are included. It is certain that

$$\sum_i n_i + 2 = n.$$

Take the graph $\mathcal{G}'$ with only edges included in the above procedure. And $\mathcal{G}'$ is a subgraph of $\mathcal{G}$ with the same node set. Notice that if a node i in $\mathcal{R}_{m_1}$ is also a terminal node of some paths in $\mathcal{R}_{m_2}$ for some $m_2 > m_1$, i already has two neighbors in $\bigcup_{k=0,\ldots,m_1} \mathcal{R}_k$. So we can select 0 for the complex weight w_{ij} where $i \in \mathcal{R}_{m_1}$ and $j \in \mathcal{R}_{m_2}$ with $m_2 > m_1$. Thus, for the Laplacian L' corresponding to $\mathcal{G}'$, it is obtained

$$L'_{\mathcal{R}_0} = \begin{bmatrix} L_1 & 0 & 0 & 0 \\ * & L_2 & 0 & 0 \\ * & * & L_3 & 0 \\ \cdots & \cdots & \cdots & \cdots \end{bmatrix}$$

where L_i is the corresponding block to the the subgraph induced by $\mathcal{R}_i$ in $\mathcal{G}'$. By the procedure for construction, it is known that nodes in $\mathcal{R}_i$ together with the two terminal nodes form a path graph. Thus, according to Lemma A.1 we attain that $\text{rank}(L_i) = n_i$ and

$$\text{rank}(L'_{\mathcal{R}_0}) = \sum_{i=1,\ldots,} \text{rank}(L_i) = \sum_{i=1,\ldots,} n_i = n-2.$$

Note that L' can be considered as a Laplacian matrix of graph $\mathcal{G}$ for a special choice of weights with some weights being 0. Hence, by utilizing the fact that either a polynomial is zero or it is not zero almost everywhere, it follows that for almost all $L \in \{L \in \mathcal{L}(\mathcal{G}) : L\xi = 0\}$, $L_{\mathcal{R}}$ is distinct from zero.

$(2) \Longleftarrow (1)$ We prove it in a contrapositive way. Suppose that there exists a node $i \notin \mathcal{R}$ such that after deleting any other node, without loss of generality say $\{1\}$, i is not reachable from $\mathcal{R}$. Denote $\mathcal{U}$ the set of nodes not in $\mathcal{R}$. So all the nodes in $\mathcal{U}$ are not reachable from $\mathcal{R}$ after removing $\{1\}$. Denote $\bar{\mathcal{U}} = \mathcal{V} - \mathcal{U} - \{1\}$. It follows that there is no edge from any node in $\bar{\mathcal{U}}$ to any node in $\mathcal{U}$. Thus by relabeling the nodes in $\mathcal{U}$ and $\bar{\mathcal{U}}$ in a consecutive way respectively, the matrix L can be transformed to the following form by a permutation matrix P, that is,

$$PLP^T = L' = \left[\begin{array}{c|c|c} L_{11} & L_{12} & L_{13} \\ \hline L_{21} & L_{22} & 0 \\ \hline L_{31} & L_{32} & L_{33} \end{array}\right],$$

where the rows and columns in L_{11} correspond to nodes $\{1\}$, the rows and columns in L_{22} correspond to the nodes in $\mathcal{U}$, and the rows and columns in L_{33} correspond to the nodes in $\bar{\mathcal{U}}$. Thus,

$$\begin{bmatrix} L_{21} & L_{22} & 0 \end{bmatrix} \mathbf{1} = 0, \ \begin{bmatrix} L_{21} & L_{22} & 0 \end{bmatrix} P\xi = 0.$$

Therefore, $\begin{bmatrix} L_{21} & L_{22} & 0 \end{bmatrix}$ is not of full row rank, which means $\det(L_{\mathcal{R}}) = 0$ for any $L \in \{L \in \mathcal{L}(\mathcal{G}) : L\xi = 0, L\zeta = 0, L\eta = 0\}$. ■

Proof of Lemma 4.2: According to Algorithm 1 and the definition of graph Π_i, an edge $(j,l) \in \Pi_i$ indicates that the information from node j and l are used to compute $w_{ij}^{(r)}$ and $w_{il}^{(r)}$.

Now we start with any edge in Π_i, say (i_1, i_2). Then we compute $w_{ii_1}^{(1)}$ and $w_{ii_2}^{(1)}$. Continually, we can find another node, say i_3, which is connected to $\{i_1, i_2\}$ since Π_i is connected. That is, edge $(i_3, i_a) \in \Pi_i$ for either $i_a = i_1$ or $i_a = i_2$. Thus, nonzero $w_{ii_3}^{(2)}$ and $w_{ii_a}^{(2)}$ are obtained. Again, there must exist a node, say i_4, which is connected to $\{i_1, i_2, i_3\}$. That is, edge $(i_3, i_a) \in \Pi_i$ for either $i_a = i_1$, $i_a = i_2$, or $i_a = i_3$. So $w_{ii_4}^{(3)}$ are $w_{ii_a}^{(3)}$ are calculated. Continuing in this way, since i has n_i neighbors we can get $n_i - 1$ linearly independent vectors. Thus, with such construction of $L(i,:)$ by Algorithm 1, if Π_i is connected, then $L(i,:)$ can span the $(n_i - 1)$-dimensional linear subspace. ■

Proof of Theorem 5.1: Suppose the graph $\mathcal{G}(k)$ switches at $k_0 = 0, k_1, k_2, \ldots$. We are always able to find a τ_M large enough such that

$$k_{i+1} - k_i \leq \tau_M \text{ for all } i = 0, 1, 2, \ldots$$

If after some time there is no more switching, we can partition $[k_i, k_{i+1})$ artificially.

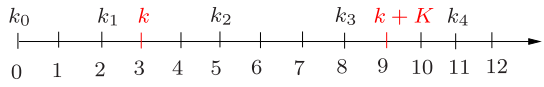


Fig. 15. An example for switching instants.

Suppose now every sensor node is jointly 2-reachable from $\mathcal{A}$ and $\Pi_i(k)$ is jointly connected. Then by the definitions there exits $K > 0$ such that for all k in the union graph $\mathcal{G}([k, k+K))$ every sensor is 2-reachable from $\mathcal{A}$, and the union graph $\Pi_i([k, k+K))$ is connected.

Let

$$\delta = K + 1.$$

For any k, without loss of generality, let

$$k \in [k_m, k_{m+1} - 1], \text{ and } k + \delta - 1 \in [k_h, k_{h+1} - 1].$$

Take Fig. 15 as an example. In the example, $k = 3$ and $K = 6$. As a result, $k \in [k_1, k_2 - 1]$, and $k + \delta - 1 \in [k_3, k_4 - 1]$.

Define

$$B(k) := L(k_m) + \cdots + L(k_h).$$

For almost all randomly chosen complex coefficients $\gamma_1, \ldots, \gamma_m$,

$$L(k_l)(i,j) \neq 0, \ l = m, \ldots, h$$

implies

$$B(k)(i,j) \neq 0.$$

It means that

$$\mathcal{G}(B(k)) = \mathcal{G}(L(k_m)) \cup \cdots \cup \mathcal{G}(L(k_h)).$$

Indeed,

$$\mathcal{G}([k, k+\delta-1]) = \mathcal{G}(L(k_m)) \cup \cdots \cup \mathcal{G}(L(k_h)) = \mathcal{G}(B(k)).$$

Moreover, note that

$$B(k)p = 0 \text{ and } B(k)\mathbf{1}_N = 0.$$

So $B(k)$ is also a Laplacian matrix corresponding to $\mathcal{G}([k, k+\delta-1])$ which has the following structure

$$\left[\begin{array}{c|c} 0_{2\times 2} & 0_{2\times(n-2)} \\ \hline * & B_s(k) \end{array}\right]$$

where

$$B_s(k) = L_s(k_m) + \cdots + L_s(k_h).$$

From

$$(k + \delta - 1) - k = K,$$

we know that in $\mathcal{G}([k, k+\delta-1])$, every sensor is 2-reachable from $\mathcal{A}$, and $\Pi_i([k, k+\delta-1])$ is connected for all i. It follows from Lemma 4.1, Remark 4.1 and Lemma 4.2 that

$$\det(B_s(k)) \neq 0,$$

which indicates that there exists a time interval δ such that the sum of $L_s(k)$ across δ is nonsingular for any k. Hence, the conclusion follows. ■

To prove Theorem 5.2, we develop two technical lemmas.

Lemma A.2: Consider the system

$$x(k+1) = (I - \epsilon(k)A^H(k)A(k))x(k), \tag{14}$$

where $x(k) \in \mathbb{C}^n$ and $A(k) \in \mathbb{C}^{n\times n}$ is taken from a finite set $\{A_1, A_2, \ldots, A_N\}$. If the following two conditions hold

1) for some positive δ and α, and all k,

$$\sum_{i=0}^{\delta-1} A^H(k+i)A(k+i) \geq \alpha I, \tag{15}$$

2) for each $A(k)$ and its corresponding positive $\epsilon(k)$,

$$I - \epsilon(k)A^H(k)A(k) \geq 0,$$

then the origin $x = 0$ of the system (14) is exponentially stable.

Proof: Consider a Lyapunov function $L(x(k)) = x^H(k)x(k)$. Then we calculate the one step difference of $L(x(k))$ along the trajectories of (14). We have

$$\begin{aligned}
&\Delta_1 L(x(k)) \\
&:= x^H(k+1)x(k+1) - x^H(k)x(k) \\
&= x^H(k)[(I - \epsilon(k)A^H(k)A(k))^2 - I]x(k) \\
&= x^H(k)(-2\epsilon(k)A^H(k)A(k) \\
&\quad + \epsilon(k)^2 A^H(k)A(k)A^H(k)A(k))x(k).
\end{aligned}$$

By condition (2), it follows that

$$\epsilon(k)^2 A^H(k)A(k)A^H(k)A(k) \leq \epsilon(k)A^H(k)A(k),$$

which is equivalent to

$$\epsilon(k)A^H(k)A(k)(I - \epsilon(k)A^H(k)A(k)) \geq 0.$$

Thus, we can obtain that

$$\begin{aligned}
&\Delta_1 L(x(k)) \\
&\leq x^H(k)(-2\epsilon(k)A^H(k)A(k) + \epsilon(k)A^H(k)A(k))x(k) \\
&= -\epsilon(k)x^H(k)A^H(k)A(k)x(k).
\end{aligned}$$

Similarly we have

$$\begin{aligned}
&\Delta_1 L(x(k+1)) \\
&:= x^H(k+2)x(k+2) - x^H(k+1)x(k+1) \\
&\leq -\epsilon(k+1)x^H(k+1)A^H(k+1)A(k+1)x(k+1),
\end{aligned}$$

and moreover

$$\begin{aligned}
&\Delta_\delta L(x(k)) \\
&:= x^H(k+\delta)x(k+\delta) - x^H(k)x(k) \\
&= \sum_{i=0}^{\delta-1} \Delta_1 L(x, k+i) \\
&\leq -\sigma \sum_{i=0}^{\delta-1} \|A(k+i)x(k+i)\|^2,
\end{aligned}$$

where σ is lower bound of $\epsilon(k)$ as stated in condition (2). So it can be concluded that $\Delta_\delta L(x(k)) \leq 0$.

Next we consider two cases: (a) For some $k' \in \{k, k+1, \ldots, k+\delta\}$, $x(k') = 0$; (b) $x(k') \neq 0$ for all $k' \in \{k, k+1, \ldots, k+\delta\}$.

For case (a), the state x will remain in zero.

For case (b), we will show that $\Delta_\delta L(x(k)) < 0$. Suppose on the contrary that $\Delta_\delta L(x(k)) = 0$. That is,

$$A(k+i)x(k+i) = 0,\ i = 0, 1, \ldots, \delta - 1.$$

With (14) we get

$$x(k) = x(k+1) = \cdots = x(k+\delta).$$

Thus,

$$\begin{aligned}
&\Delta_\delta L(x(k)) \\
&\leq -\sigma \sum_{i=0}^{\delta-1} x^H(k+i)A^H(k+i)A(k+i)x(k+i), \\
&= -\sigma x^H(k) \left(\sum_{i=0}^{\delta-1} A^H(k+i)A(k+i) \right) x(k),
\end{aligned}$$

Furthermore, considering condition (1), we know

$$\Delta_\delta L(x(k)) \leq -\sigma\alpha \|x(k)\|^2.$$

Recall that $\Delta_\delta L(x(k)) = 0$. So we can get that $x(k) = 0$, a contradiction. Thus we obtain $\Delta_\delta L(x(k)) < 0$ for $x(k) \neq 0$.

On the other hand, we can rewrite $\Delta_\delta(L(x(k))$ as follows.

$$\begin{aligned}
&\Delta_\delta L(x(k)) \\
&= -x^H(k)\Delta\left(A(k), \ldots, A(k+\delta-1)\right)x(k),
\end{aligned}$$

where $\Delta\left(A(k), \ldots, A(k+\delta-1)\right)$ is the matrix polynomial of $A(k), \ldots, A(k+\delta-1)$. Recall that $\Delta_\delta L(x(k)) < 0$ for $x(k) \neq 0$, and that $A(k)$'s and $\epsilon(k)$'s are taken from finite sets. There must exist a positive γ such that

$$\Delta\left(A(k), \ldots, A(k+\delta-1)\right) \geq \gamma I, \quad \forall k$$

which leads to

$$\Delta_\delta L(x(k)) \leq -\gamma \|x(k)\|^2.$$

That is, along all trajectories of (14), the Lyapunov function satisfies

$$\frac{\Delta_\delta L(x(k))}{L(x(k))} \leq -\gamma < 0.$$

Therefore, $L(x(k))$ exponentially converges to zero. Since $L(x(k))$ is quadratic, it follows that the origin $x = 0$ of (14) is exponentially stable. ■

Lemma A.3: Consider m matrices $A_1, A_2, \ldots, A_m \in \mathbb{C}^{n\times n}$. If $\text{rank}(A_1 + A_2 + \cdots + A_m) = n$, then $A_1^H A_1 + A_2^H A_2 + \cdots + A_m^H A_m$ is Hermitian and positive definite.

Proof: Let

$$B = \begin{bmatrix} A_1^H & A_2^H & \cdots & A_m^H \end{bmatrix}^H.$$

It is certain that $B^H B = A_1^H A_1 + A_2^H A_2 + \cdots + A_m^H A_m$. Note that $B^H B$ is a Hermitian matrix due to

$$(B^H B)^H = B^H B.$$

From the fact that

$$A_1 + A_2 + \cdots + A_m = \begin{bmatrix} I & I & \cdots & I \end{bmatrix} \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_m \end{bmatrix}$$

we have

$$n = \text{rank}(A_1 + A_2 + \cdots + A_m) \leq \text{rank}(B) \leq n$$

and this implies that

$$\text{rank}(B) = n.$$

In addition, suppose that there exists a vector $x \in \mathbb{C}^n$ such that

$$x^H B^H B x = 0.$$

Then

$$\|Bx\|^2 = 0 \Rightarrow Bx = 0.$$

Due to $\text{rank}(B) = n$ we have $x = 0$, which means that for any $x \neq 0$ it holds that

$$x^H B^H B x > 0.$$

Then the conclusion follows. ■

Proof of Theorem 5.2: We can write the distributed algorithm (13) in the matrix form as

$$\hat{p}(k+1) = \left(I - \epsilon(k) \begin{bmatrix} 0 & 0 \\ L_s^H(k) L_a(k) & L_s^H(k) L_s(k) \end{bmatrix} \right) \hat{p}(k). \tag{16}$$

Note that for the anchors, $\hat{p}_a(k) = p_a$. Then we apply the coordinate transformation $\tilde{p}_s(k) = \hat{p}_s(k) - p_s$ and derive

$$\tilde{p}_s(k+1) = \left(I - \epsilon(k) L_s^H(k) L_s(k) \right) \tilde{p}_s(k). \tag{17}$$

Notice that $w_{ij}(k)$'s are finite due to finite nodes and edges. So $L_s(k)$'s are finite. According to the procedure for designing $\epsilon(k)$, it holds that $\epsilon(k)$'s are finite, and satisfy

$$I - \epsilon(k) L_s^H(k) L_s(k) \geq 0.$$

So to show that $\hat{p}_i(k)$ is exponentially convergent to p_i, by Lemma A.2, we need to show that there exist some positive δ and α such that for all k,

$$\sum_{i=0}^{\delta-1} L_s^H(k+i) L_s(k+i) \geq \alpha I, \ \forall k. \tag{18}$$

Suppose the graph $\mathcal{G}(k)$ switches at $k_0 = 0, k_1, k_2, \ldots$. We are always able to find a τ_M large enough such that

$$k_{i+1} - k_i \leq \tau_M \text{ for all } i = 0, 1, 2, \ldots$$

If after some time there is no more switching, we can partition $[k_i, k_{i+1})$ artificially.

Suppose now every sensor node is jointly 2-reachable from $\mathcal{A}$ and Π_i is jointly connected. Then by the definitions there exits $K > 0$ such that for all k in the union graph $\mathcal{G}([k, k+K))$ every sensor is 2-reachable from $\mathcal{A}$, and the union graph $\Pi_i([k, k+K))$ is connected.

Let

$$\delta = K + 1.$$

In what follows, we will show that for all k

$$W(k) := \sum_{i=0}^{\delta-1} L_s^H(k+i) L_s(k+i) \geq \alpha I$$

holds for some positive α.

For any k, without loss of generality, let

$$k \in [k_m, k_{m+1} - 1], \text{ and } k + \delta - 1 \in [k_h, k_{h+1} - 1].$$

Define

$$E(k) := L^H(k_m) L(k_m) + \cdots + L^H(k_h) L(k_h),$$

and

$$B(k) := L(k_m) + \cdots + L(k_h).$$

According to the proof of Theorem 5.1, we get

$$\det\left(B_s(k)\right) \neq 0. \tag{19}$$

Furthermore, $E(k)$ has the following structure

$$\left[\begin{array}{c|c} * & * \\ \hline * & E_s(k) \end{array}\right]$$

where

$$E_s(k) = L_s^H(k_m) L_s(k_m) + \cdots + L_s^H(k_h) L_s(k_h).$$

From (19) and Lemma A.3, it is true that $E_s(k)$ is Hermitian and positive definite. From the definition of $W(k)$, we have

$$\begin{aligned} W(k) = {} & L_s^H(k_m) L_s(k_m)(k_{m+1} - k) \\ & + L_s^H(k_{m+1}) L_s(k_{m+1})(k_{m+2} - k_{m+1}) + \cdots \\ & + L_s^H(k_{h-1}) L_s(k_{h-1})(k_h - k_{h-1}) \\ & + L_s^H(k_h) L_s(k_h)(k + \delta - k_h). \end{aligned}$$

Note that the lengths of all time interval appeared in the formula $W(k)$ are positive and $E_s(k)$ is Hermitian and positive definite. So $W(k)$ is also Hermitian and positive definite. Next we will show that the smallest eigenvalue of $W(k)$ is uniformly lower bounded with respect to k. With the fact that the number of switches during $[k, k+\delta-1]$ is finite, and that the length between any two consecutive switching are bounded, we then know that all $W(k)$'s form a finite set. This means, the smallest eigenvalue of $W(k)$ is uniformly lower bounded. In other words, there exists $\alpha > 0$ such that for all k,

$$W(k) = \sum_{i=0}^{\delta-1} L_s^H(k+i) L_s(k+i) \geq \alpha I.$$

Thus, by using Lemma A.2, the conclusion follows. ■

REFERENCES

[1] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The cricket location-support system," in *Proc. 6th Annu. Int. Conf. Mobile Comput. Netw.*, 2000, pp. 32–43.

[2] G. Mao, B. Fidan, and B. D. O. Anderson, "Wireless sensor network localization techniques," *Comput. Netw.*, vol. 51, no. 10, pp. 2529–2553, 2007.

[3] R. M. Vaghefi and R. M. Buehrer, "Cooperative joint synchronization and localization in wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 63, no. 14, pp. 3615–3627, 2015.

[4] L. Doherty, K. S. J. Pister, and L. El Ghaoui, "Convex position estimation in wireless sensor networks," in *Proc. 20th Annu. Joint Conf. IEEE Comput. Commun. Soc.*, 2001, pp. 1655–1663.

[5] T. Eren *et al.*, "Rigidity, computation, and randomization in network localization," in *Proc. 23rd Annu. Joint Conf. IEEE Comput. Commun. Soc.*, 2004, pp. 2673–2684.

[6] M. Cao, B. D. O. Anderson, and A. S. Morse, "Sensor network localization with imprecise distances," *Syst. Control Lett.*, vol. 55, no. 11, pp. 887–893, 2006.

[7] B. D. O. Anderson *et al.*, "Graphical properties of easily localizable sensor networks," *Wireless Netw.*, vol. 15, no. 2, pp. 177–191, 2009.

[8] U. Khan, S. Kar, B. Sinopoli, and J. M. Moura, "Distributed sensor localization in euclidean spaces: Dynamic environments," in *Proc. 2008 46th Annu. Allerton Conf. Commun., Control, Comput.*, 2008, pp. 361–366.

[9] U. Khan, S. Kar, and J. M. F. Moura, "Distributed sensor localization in random environments using minimal number of anchor nodes," *IEEE Trans. Signal Process.*, vol. 57, no. 5, pp. 2000–2016, May 2009.

[10] Y. Diao, Z. Lin, and M. Fu, "A barycentric coordinate based distributed localization algorithm for sensor networks," *IEEE Trans. Signal Process.*, vol. 62, no. 18, pp. 4760–4771, Sep. 2014.

[11] D. Niculescu and B. Nath, "Ad Hoc positioning system (APS) using AOA," in *Proc. 22nd Annu. Joint Conf. IEEE Comput. Commun.*, 2003, vol. 3, pp. 1734–1743.

[12] I. Shames, A. N. Bishop, and B. D. O. Anderson, "Analysis of noisy bearing-only network localization," *IEEE Trans. Autom. Control*, vol. 58, no. 1, pp. 247–252, Jan. 2013.

[13] T. Eren, W. Whiteley, and P. N. Belhumeur, "Using angle of arrival (bearing) information in network localization," in *Proc. 45th IEEE Conf. Decision Control*, 2006, pp. 4676–4681.

[14] T. Eren, "Using angle of arrival (bearing) information for localization in robot networks," *Turk. J. Elect. Eng.*, vol. 15, no. 2, pp. 169–186, 2007.

[15] G. Zhu and J. Hu, "A distributed continuous-time algorithm for network localization using angle-of-arrival information," *Automatica*, vol. 50, no. 1, pp. 53–63, 2014.

[16] J. Zhong, Z. Lin, Z. Chen, and W. Xu, "Cooperative localization using angle-of-arrival information," in *Proc. 11th IEEE Int. Conf. Control Autom.*, 2014, pp. 19–24.

[17] S. Zhao and D. Zelazo, "Localizability and distributed protocols for bearing-based network localization in arbitrary dimensions," *Automatica*, vol. 69, pp. 334–341, 2016.

[18] P. Barooah and J. P. Hespanha, "Estimation on graphs from relative measurements," *IEEE Control Syst. Mag.*, vol. 27, no. 4, pp. 57–74, Aug. 2007.

[19] P. Barooah and J. P. Hespanha, "Estimation from relative measurements: Electrical analogy and large graphs," *IEEE Trans. Signal Process.*, vol. 56, no. 6, pp. 2181–2193, Jun. 2008.

[20] C. Ravazzi, P. Frasca, R. Tempo, and H. Ishii, "Almost sure convergence of a randomized algorithm for relative localization in sensor networks," in *Proc. 52nd IEEE Conf. Decision Control*, 2013, pp. 4778–4783.

[21] G. Antonelli, F. Arrichiello, F. Caccavale, and A. Marino, "A decentralized controller-observer scheme for multi-agent weighted centroid tracking," *IEEE Trans. Autom. Control*, vol. 58, no. 5, pp. 1310–1316, May 2013.

[22] Z. Lin, M. Fu, and Y. Diao, "Distributed self localization for relative position sensing networks in 2D space," *IEEE Trans. Signal Process.*, vol. 63, no. 14, pp. 3751–3761, Jul. 2015.

[23] G. Piovan, I. Shames, B. Fidan, F. Bullo, and B. D. O. Anderson, "On frame and orientation localization for relative sensing networks," *Automatica*, vol. 49, no. 1, pp. 206–213, 2013.

[24] J. M. Dion, C. Commault, and J. van der Woude, "Generic properties and control of linear structured systems: A survey," *Automatica*, vol. 39, no. 7, pp. 1125–1144, 2003.

[25] R. Olfati-Saber and R. M. Murray, "Consensus protocols for networks of dynamic agents," in *Proc. Amer. Control Conf.*, 2003, pp. 951–956.

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

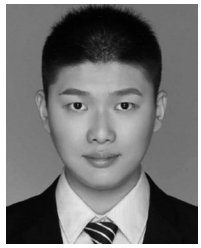
[27] F. Meyer, H. Wymeersch, M. Fröhle, and F. Hlawatsch, "Distributed estimation with information-seeking control in agent networks," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 11, pp. 2439–2456, Nov. 2015.

[28] Z. Lin, L. Wang, Z. Han, and M. Fu, "Distributed formation control of multi-agent systems using complex Laplacian," *IEEE Trans. Autom. Control*, vol. 59, no. 7, pp. 1765–1777, Jul. 2014.

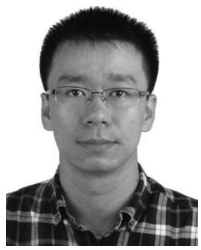
Zhiyun Lin (SM'10) received the Bachelor's degree in electrical engineering from Yanshan University, Qinhuangdao, China, in 1998, the Master's degree in electrical engineering from Zhejiang University, Hangzhou, China, in 2001, and the Ph.D. degree in electrical and computer engineering from the University of Toronto, Toronto, ON, Canada, 2005.

He was a Postdoctoral Research Associate in the Department of Electrical and Computer Engineering, University of Toronto, from 2005 to 2007. He joined the College of Electrical Engineering, Zhejiang University, in 2007. He is currently a Professor of systems control in the same department. He is also affiliated with the State Key Laboratory of Industrial Control Technology, Zhejiang University. He held Visiting Professor positions at several universities, including The Australian National University, Canberra, ACT, Australia, University of Cagliari, Cagliari, Italy, University of Newcastle, Callaghan, NSW, Australia, University of Technology Sydney, Ultimo, NSW, Australia), and Yale University, New Haven, CT, USA.

His research interests include 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 currently an Associate Editor for *Nonlinear Analysis: Hybrid Systems* and the *International Journal of Wireless and Mobile Networking*.



Tingrui Han received the B.S. degree in automation from Zhejiang University, Hangzhou, China, in 2012. He is currently working toward the Ph.D. degree in control theory and control engineering in the College of Electrical Engineering, Zhejiang University. His research interests include multiagent systems, networked control, and distributed algorithms.



Ronghao Zheng received the Bachelor's degree in electrical engineering and the Master's degree in control theory and control engineering, both from Zhejiang University, Hangzhou, China, in 2007 and 2010, respectively, and the Ph.D. degree in mechanical and biomedical engineering from the City University of Hong Kong, Hong Kong, in 2014.

He is currently in the College of Electrical Engineering, Zhejiang University. His research interests include distributed algorithms and control, especially the coordination of networked mobile robot teams with applications in automated systems and security.

Minyue Fu (F'03) received the Bachelor's degree in electrical engineering from the University of Science and Technology of China, Hefei, China, in 1982, and the M.S. and Ph.D. degrees in electrical engineering from the University of Wisconsin-Madison, Madison, WI, USA, in 1983 and 1987, respectively.

From 1987 to 1989, he was an Assistant Professor in the Department of Electrical and Computer Engineering, Wayne State University, Detroit, MI, USA. In 1989, he joined the Department of Electrical and Computer Engineering, the University of Newcastle, Callaghan, NSW, Australia, where he is currently the Chair Professor in electrical engineering. He was a Visiting Associate Professor at the University of Iowa from 1995 to 1996, a Senior Fellow/Visiting Professor at Nanyang Technological University, Singapore, in 2002, and a 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, China.

His main research interests include control systems, signal processing, and communications. His current research projects include networked control systems, smart electricity networks, and superprecision 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*.