DOI: 10.7641/CTA.2013.12115

传感器网络的可定位性条件

刁英斐[1], 付敏跃[2,3†], 张焕水[1]

(1. 山东大学 控制科学与工程学院, 山东 济南 250061; 2. 浙江大学 控制科学与工程学系, 浙江 杭州 310027;
3. 纽卡斯尔大学 电气工程与计算机科学学院, 澳大利亚 卡拉汉 2308)

摘要: 针对两个节点集合, 提出了一组可定位性条件. 可定位性条件包含两部分: 两个集合之间需要的连接边的数目, 以及如何布置这些连接边. 本文将每组节点和它们的内部连接边刻画成一个距离图. 两个节点集合之间的可定位性判定等同于两个融合图的全局刚性测试. 针对两个融合图的可定位性, 给出了一系列的充要条件.

关键词: 传感器网络; 定位算法; 图论

中图分类号: TP273　　　**文献标识码**: A

Localizability exploration of sensor network

DIAO Ying-fei[1], FU Min-yue[2,3†], ZHANG Huan-shui[1]

(1. School of Control Science and Engineering, Shandong University, Jinan Shandong 250061, China;
2. Department of Control Science and Engineering, Zhejiang University, Hangzhou Zhejiang 310027, China;
3. School of Electrical Engineering and Computer Science, University of Newcastle, Callaghan NSW 2308, Australia)

Abstract: We propose a group of localizable conditions for two sets of nodes. The localizable conditions need to answer two questions: how many edges and where these edges are placed between two sets of nodes. Here, each set of nodes and their internal connections are modeled as a distance graph. The localizability exploration between two sets of nodes is characterized by the global rigidity test of two merging graphs. A series of necessary and sufficient conditions on the localizability of two merging graphs is given.

Key words: sensor network; localization; graph theory

1 Introduction

Localization problem is a fundamental and important issue among the abundant expected applications of sensor network[1–2], which include but not limited to the area of wildlife tracking[3], ocean monitoring[4], intelligent factory[5–6], information encryption[7] and the newly appeared carbon sink[8].

Generally, there are two kinds of methods for obtaining the location information: distance-based or distance-free localization scheme. In this paper, we will discuss the first category. Note that the localization scheme here is a different definition from the range detection technique. The distance detection or ranging technique usually refers to the technique used for obtaining the distance measurement. The distance between the signal source and the target can be measured by detecting the flight time of radio or ultrasonic signals[9]. A localization scheme is a strategy to use the detected distance information to get location information.

Localization scheme can be divided into two cases: sequential scheme and concurrent scheme[10]. For a traditional sequential scheme, there are at least three anchor nodes in a 2D plane and every location unknown node is required to check if it has three direct distance measurements with anchor nodes. If so, its location will be computed by its distance measurements with the three location known nodes and it will be added into the set of anchor nodes. If not, it will be treated as unlocalizable and its location could not be computed. The scheme will check all the nodes one by one. For a concurrent scheme, every node will compute its location by using distance measurement within its neighborhood. Each node can compute its location synchronously and they will iterate to the correct location value finally.

For a randomly deployed sensor network, it is probable that not all the sensor nodes are localizable since the existence of so-called flex and flip ambiguity in the localization problem[11].

The existing work on localizability are mostly focused on the localizable conditions of either a whole network[12] or one single node[13]. The localizability in both cases are checked from the perspective of the connectivity and rigidity of the graph. The drawback for

Received 21 May 2012; revised 15 October 2012.
† Corresponding author. E-mail: minyue.fu@newcastle.edu.au.
This work is supported by the National Natural Science Foundation of China (Nos. 61134001, 61120106011, 61203029), and the Taishan Scholar Construction Engineering by Shandong Government.

analyzing from the view of a whole network is that they can only judge whether the whole network is localizable but cannot point out the localizable nodes from a given network. In practice, a randomly deployed sensor network is hard to be entirely localizable[13]. The drawback from the view of one single node is that it can test only one node at a time and the condition is not necessary and sufficient. Moreover, connectivity and rigidity test for both cases need global information.

Neither for a whole network nor for one single node, our thread is to explore the localizability condition for a set of nodes. Assume that there is one set of location known nodes in the whole network. Previous researchers would analyze the connectivity and rigidity of every location unknown node one by one. In this paper, we will collect each single node with its several neighbors together to build a test set. The set of nodes and their connected edges form a graph, named a free graph, whereas the location known nodes form another graph named the anchor graph. Then, we explore the localizable conditions between the two graphs. The localizable conditions here are try to answer two questions: how many connections between these two graphs are required and where should they be drawn. Our preliminary work on this topic also appeared as a conference proceeding[14].

In this paper, we only consider two cases of the set of nodes to be localized, in which the number of nodes to be localized is no more than three. As pointed out in both [15] and [16], the gap between sufficient conditions of localizability and the commonly used necessary conditions can be significantly reduced after considering the jointly localizability of two or three nodes to be localized. In other way, through introduce an operation of Henneberger sequence, our sufficient conditions between two set of nodes can be extended to the cases of more nodes to be localized in a sequential way.

2 Problem statement and related work

In the localizability exploration, the sensor network is usually treated as a corresponding distance graph $G(V,E)$, where V and E stand for the vertex set and edge set of the graph, respectively. The localizability of a network can be modeled as the global rigidity of its distance graph.

A graph is rigid if it could not be continuously deformed. One graph may correspond to several kinds of network realization in practice. If every realization of the graph with the same distance constraints is identical, then the graph is globally rigid. A graph is globally rigid if and only if it is 3-connected and redundantly rigid. Here, 3-connected means the graph is still connected after removal of any two nodes. Redundantly rigid means the graph is still rigid after removal of any edge. If a globally rigid graph could not be globally rigid after removing any one edge from the graph, it is a minimally globally rigid graph. We describe these concepts in Fig.1.

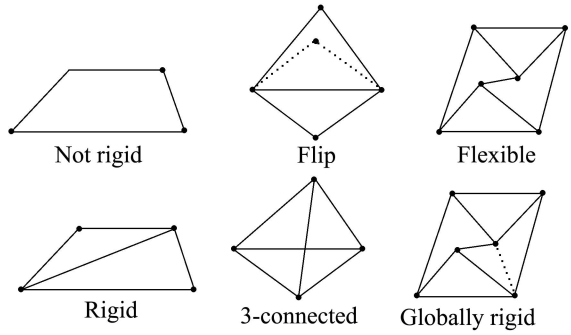


Fig. 1 Several concepts of graph rigidity

In this paper, we want to explore the localizable condition between two sets of nodes. This can be characterized as the global rigidity test of two merging graphs, one of which is location-all-known. It can be mathematically described as below. Given two graphs $G_1(V_1,E_1)$ containing N_1 nodes and $G_2(V_2,E_2)$ containing N_2 nodes, we want to draw fixed number of edges between these two graphs to merge them together and obtain a merged minimally globally rigid graph $G(V,E)$ containing all N_1+N_2 nodes and edges between them. Without loss of generality, we assume all N_1 nodes in G_1 are location known. Then, we turn to our question that, when merging G_2 onto G_1, how many edges are required and how to draw these edges between graphs G_1 and G_2. For simplicity of analysis, we assume that G_1 has only three nodes here and discuss the several cases of G_2 with different number of nodes and variable edges.

A natural question is why we choose to explore conditions for several specified cases of G_1 and G_2.

First, two merging graphs $G_1(V_1,E_1)$ and $G_2(V_2,E_2)$ have variables (V_1,E_1) and (V_2,E_2) and numerous possible combinations, if either of them if not globally rigid. It is hard to give a generalized condition for merging G_1 and G_2 and we choose to give a necessary and sufficient condition for two merging graphs with specified number of nodes and edges and also give a necessary condition for two merging graphs with a variable number of nodes and edges.

Second, we believe the condition for two merging graphs could be much tighter than the condition for one node merging with a graph[13] since the interconnection between nodes in the test graph might eliminate some degrees of freedom.

Third, for a localization scheme, the main object is to localize nodes as many as possible even if not all nodes of the network are localizable. In other words, we can tolerate an algorithm that cannot find all possible combinations. There are kinds of 'easily localizable' network topology, such as trilateration case in [11] and 'wheel' case in [15]. Both cases can find parts of the localizable nodes through the whole network even if the network is not whole localizable. The condition, such as the one of trilateration, is only a sufficient but not necessary condition. In other words, there are still some localizable nodes that are not included in that case.

We show an example of graph in Fig.1, which does not fit trilateration condition but is still localizable. As mentioned in [15], this kind of deployment is common for a practical sensor network, especially when nodes are deployed at the boundary of the sensor network. Its localizability can be analyzed through the localizable condition of one single node in [13]. We will explore the jointly localizability by combining two location unknown nodes together, which are indicated by solid circles in Fig.1. We will prove the connections shown as dotted line in Fig.2 is the unique way for merging this set onto the location known set by introducing minimal connections.

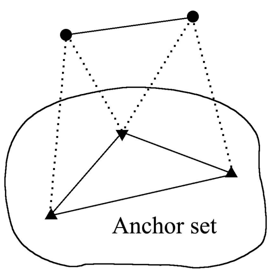


Fig. 2 A counter example of 3-connected condition

The algorithm in [17], based on the 3-connected and redundantly rigid condition, could be used for checking the global rigidity of the network. This condition is explored in view of the whole network. Using this condition on a given network graph G, we could determine whether the whole network is localizable but cannot find out which nodes of the network are not localizable. The work of [13] analyzed the localizability of each single node. They give a pair of mutual independent sufficient condition and necessary condition for one single node's localizability. So far as we know, this is the closest pair of localizable conditions.

Similar to our set based thread, there are also some other works focused on the localizable conditions for the merging of two globally rigid subnetwork[18–19]. Both works are concerned with localizable conditions between two globally rigid graphs. In this paper, we also give conditions for the global rigidity of the merged graph, but only one of the graph is required to be globally rigid and the other graph's structure is not constrained, which is named as a free graph here. The graph without promise of global rigidity is also contained in the range of free graph. This is also our main difference from the existing work such as [18] and [19].

3 Conditions for localizability of two merging graphs

In this paper, we propose localizable conditions for several specialized cases of G_1 and G_2. Without loss of generality, we assume G_1 has three location known nodes inside and G_2 is the free graph.

3.1 A necessary condition for localizability of two merging graphs

In a 2D plane, if a graph is fixed on the uniquely position, it has zero degree of freedom. But if a graph is globally rigid, it has still 3 degrees of freedom since the lack of three anchors[12]. These 3 degrees of freedom correspond to the rotation, translation and reflection of the graph in 2D plane[20]. Global rigidity is only a necessary condition for uniquely localizable.

Actually, each free node's movement in a 2D plane has 2 degrees of freedom and thus there are $2n$ degrees of freedom for n nodes. One pair-wise connection between two nodes could eliminate 1 degree of freedom. First, we need to figure out how many connections are required at least to guarantee the global rigidity of a graph. Then, how many connections are required in several cases of G_1 and G_2 with specified number of vertex and edges.

We first give a lemma about the general case.

Lemma 1 If a graph G with n nodes is globally rigid, it has at least $2n-2$ edges in the graph.

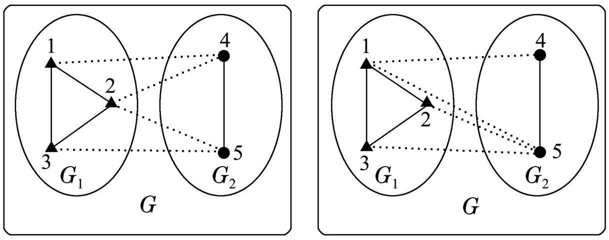
Proof For a globally rigid graph G containing n nodes, although relative position between each pair of node is fixed, it still has 2 degrees of freedom in a 2D plane as a whole. Since each node in a 2D plane has 2 degrees of freedom, there should be $2n$ degrees of freedom for n nodes if all connections in G are eliminated. In other words, all the edges in a globally rigid graph G eliminate at least $2n-2$ degrees of freedom. There should be at least $2n-2$ edges to guarantee the elimination of the $2n-2$ degrees of freedom.

3.2 Two nodes cases

As shown in Fig.3, graph G_1 contains three anchor nodes and graph G_2 contains two nodes. There is an interconnected node in graph G_2 between edges from node 4 and node 5. The example shown in Fig.3 is almost the same with the counter example shown in Fig.2. Difference is in the treatment that we package node 4 and node 5 into a set. Though either node 4 or node 5 does not fit the trilateration condition, the merged graph G is still globally rigid. This is caused by the fact that, compared with the global rigidity condition for each single node, the condition for two merging graphs will be weaken through introducing the interconnections inside one of the graphs.

We first answer the question that how many edges between two graphs are required at least. Through Lemma 1, there should be at least 4 connections to merge two graph G_1 and G_2 together to be a globally rigid graph.

Then, we should explore how to draw these 4 connections between two graphs to make the merging graph globally rigid. Since each node in both graph is identical, if the link number is limited to 4, it is easy to exhaust that there are only two possible cases for two graphs as shown in Fig.3.



(a) (b)

Fig. 3 Two nodes case

Here, we introduce a previous conclusion about globally rigidity as a lemma:

Lemma 2[21] A distance graph is globally rigid if and only if the graph is 3-connected and redundantly rigid.

Then, we give our condition about the above case:

Theorem 1 Given two graphs, one is globally rigid G_1, containing only three anchors inside, and the other one G_2 is free, having two connected nodes inside. The merged graph of G_1 and G_2 is globally rigid if and only if there is one node in G_1 connected to two nodes in G_2, while the other two nodes in G_1 connect to two different nodes in G_2.

Proof (Sufficiency): There are two nodes in G_2 and each of them have two connecting edges added. They have at least one different connected node in G_1. To show the 3-connectedness of the merged graph, we note that if both nodes in G_2 are removed, the remaining graph, which is G_1, is connected. If one node in G_2 and one node in G_1 are removed, the remaining node in G_2 is still connected to G_1. To see redundant rigidity, if one connecting edge is removed, the node in G_2 not connected to this edge is rigid and once this node is rigid, the other node is also rigid because of its connection to the first node and to G_1.

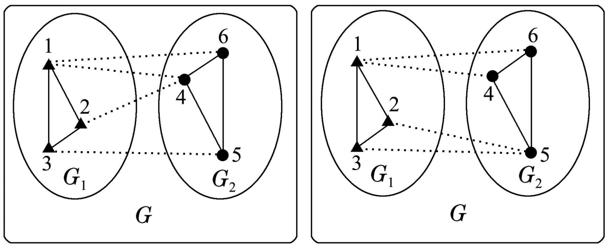
(Necessity): There are four edges needed to draw on three nodes at each side, and every node in corresponding graph is identical. There are only two possible connected cases if the number of links is fixed at 4, one is shown in Fig.3(a) and the other in Fig.3(b). From Lemma 2, we could determine the merged graph shown in Fig.3(b) is not globally rigid since the graph would not be connected if node 1 and node 5 were removed. Then, only one possible connected case is globally rigid.

Remark 1 For the case with more than 3 nodes in G_1, the merged graph G is globally rigid if there are 4 anchor nodes having connections with the free graph, any three of the anchor nodes is not collinear, and one of the link from the intersected node in G_1 is replaced by a new edge from the 4th node. This remark could be proved through adding a new node into the graph in Theorem 1 to build a newly minimally globally rigid graph.

Remark 2 For the case with more than 4 links between two graphs, it could be recognized as adding redundant links after building the minimally globally rigid graph. The condition we given here is a tight bound for the setup of a merged globally rigid graph. Without the requirement of minimal links, the sufficient condition still works.

3.3 Three-node cases

As shown in Fig.4, both graph G_1 and graph G_2 contain 3 nodes. The difference is that there are still 3 degrees of freedom for G_2. In other words, the coordination of nodes in G_1 is fixed but nodes in G_2 could move in 2D plane ignoring links between two graphs. Here, the nodes in G_2 are pairwise connected and form a triangle.



(a) (b)

Fig. 4 Three-node case

First, we analyzed possible connected way between two graphs as shown in Fig.4. For a graph G containing 6 nodes, according to Lemma 1, there should be at least 10 edges to make sure the global rigidity. Besides, 6 edges forming two triangles, there still should be 4 more edges between two graphs. An extra constraint about these 4 edges is that they should be directly connected with 3 anchors in graph G_1.

Since there are 4 edges directly connected with 3 anchor nodes, one of three anchors in G_1 should have two direct connections with G_2. Without loss of generality, we assume that node 1 is the one that has two direct connections with two nodes in graph G_2, nodes 4 and 6 as shown in Fig.4. Then, there are only two kinds of choice for the left two nodes in graph G_1. They are connected to either two different nodes in G_2, as shown in Fig.4(a), or the same node left in G_2, as shown in Fig.4(b).

Now, we prove the previous one is the right choice to make the formed graph globally rigid. We describe this result as a theorem as follows:

Theorem 2 Given a graph G_1 containing 3 anchor nodes, the free graph G_2 could be added on G_1 to construct a minimally globally rigid graph G if and only if the two conditions below satisfied simultaneously.

1) There are four edges between G_1 and G_2.

2) One node in G_1 has direct connections with two different anchor nodes in G_1, while the other two nodes in G_1 has single connection with two different anchor nodes in G_1.

Proof (Sufficiency): Without loss of generality, we can construct a merging graph and connecting edges as shown in Fig.4(a), according to the requirement of the theorem. As required in Lemma 1, there needs at least four edges between two graphs G_1 and G_2. Thus, it is necessary that each node in G_2 has a connecting edge and one node has an extra connecting edge. There are either 3 or 4 connecting nodes in G_1. There are two ways to draw these connecting edges between two graphs. Each of them is shown in Fig.4(a) and Fig.4(b), respectively. Since our proposed condition also require two nodes having single connecting edge not to interconnect with the same node in G_1. Fig.4(b) is except and only Fig.4(a) fits the requirement. In this case, the 3-connected condition can be visually inspected. To see the redundant rigidity, we note that if the fourth connecting edge is removed, the three nodes in G_2 are rigid because they have 6 edges constraining them. If a different edge (connecting or internal) is removed, the node in G_2 with two connecting edges is rigid, which in turn implies that the remaining nodes in G_2 are rigid.

(Necessity): As the analysis before, there are only two kinds of connections if the number of edges between two graphs is limited to four. By contradictory, suppose that there exist a globally rigid graph, in which connecting edges are the same as shown in Fig. 4(b). According to Lemma 2, the globally rigid graph G should be 3-connected and redundantly rigid. If we remove node 1 and node 5, the merged graph formed by all 6 nodes will not be connected and therefore, the merged graph is not 3-connected. The graph shown in Fig. 4 (b) is not globally rigid and the merged graph shown as Fig. 4 (a) is the only choice.

4 Extension to more general cases

4.1 Henneberg sequence and vertex splitting

We have shown the localizable conditions when the number of nodes in the free graph is no more than three. When the number of nodes in the free graph increases, we can use a series of Henneberg sequence to extend our localizable conditions to be suitable for more general cases.

BERG and JORDAN[22] have given a constructive condition on how to build a globally rigid graph beginning with a so-called K_4 graph. They pointed out that every globally rigid graph can be obtained from a K_4 graph through Henneberg operations and edge insertions. Jordan and Szabadka[23] show that a vertex splitting process can also construct a globally rigid graph.

A basic concept is the K_4 graph as shown in Fig.5.

In Fig.5, the left part indicates an realization of a original K_4 graph. One important property of K_4 is its global rigidity. It is easy to check that it is also minimally globally rigid. The other two graphs indicate two operations of Henneberg sequences, named 0-extension and 1-extension. The 0-extension, shown in the middle of Fig.5, adds a new vertex u and edges uv and uw to two vertex v and w. The 1-extension, shown on the right, adds a new vertex u and edges uv, uw and ut. Besides, the edge vw is also deleted. Actually, deleting vt or wt is identical. As defined in [23], a series of 0-extension and 1-extension are named as a Henneberg sequence.

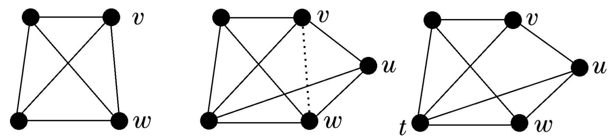


Fig. 5 The left one is original K_4 and the other shows the process of Henneberg extension

There is an important result in [21] and [1], which is expressed as a lemma here.

Lemma 3 If a graph G is 3-connected and redundantly rigid, then G can be obtained from K_4 by a sequence of Henneberg operations and edge insertions.

There is also an conclusion in [23] introducing a newly appeared vertex splitting operation, which can preserve global rigidity of the graph. Fig.6 indicate the operation of vertex splitting. For a globally rigid graph $G = (V, E)$, vertex splitting operation choose any one vertex, u, and an edge, uv, and then divide u's neighbors into two parts, V_1 and V_2. Without loss of generality, choose the square vertex in Fig.6 as V_1 and the circle vertex as V_2. Then, split u into two vertex u_1 and u_2. Add edges between u_1 and u_2. Connect u_1, u_2 with V_1, V_2. The connecting edges between V_1 and V_2 do not change. Also connect u_1, u_2 with v. Then, we can obtain a new graph G'.

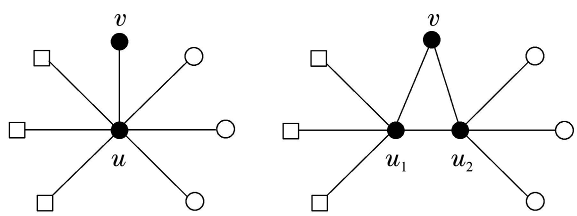


Fig. 6 Vertex splitting operation (square and circle vertex indicate V_1 and V_2, respectively)

The result of Theorem 4.5 in [23] can be expressed as a lemma below.

Lemma 4 If a graph G is 3-connected and redundantly rigid, and then G', which is obtained from G by a nontrivial vertex splitting operation, is also 3-connected and redundantly rigid.

4.2 Remark on sequential localization scheme

The traditional sequential localization scheme is based on 3-connected condition. Every node's coordination is computed by its three neighbors if those neighbors' coordination was known. The nodes fit 3-connected condition is computed one by one. This process can be finished in polynomial steps. As we pointed before, only small part of the network is localizable according to this way. The case shown in Fig.2 is a counter example.

We choose a sequential way, and however, we consider a set of nodes rather than one single node at each step. This will improve the efficiency of the sequential way and reduce the chance of missing localizable nodes in the network.

When the number of nodes in the free graph larger than three, the extended localizable condition still based on a set of nodes, and therefore, the nodes can also be localized sequentially.

5 Performance evaluation

In this section, we give a 100-round Monte-Carlo simulation to evaluate the improvement on detecting localizable nodes of our proposed conditions compared with the commonly-used trilateration method. In each round, we create a randomly-deployed network and use the trilateration and our proposed conditions to detect localizable nodes respectively. The network is deployed in a 100×100 unit area and the average degree of connectivity in all these rounds are 8.27. The number of these methods are recorded and shown in Fig.7. To show the result in a more compact way, we average the result of every 5 rounds, and thus, there are 20 data in the horizontal axis. As shown in Fig.7, the solid line with solid circle mark indicate the detected localizable ratio by our proposed conditions. The dashed line with square mark indicate the ratio of localizable nodes detected by the trilateration method. We can note that our proposed conditions can significantly improve the localizability of the network. According to our simulation, the average localizable ratio of our proposed conditions and trilateration method are 63.63% and 45.88%. The improvement of our proposed conditions in localizability is about 38.69%.

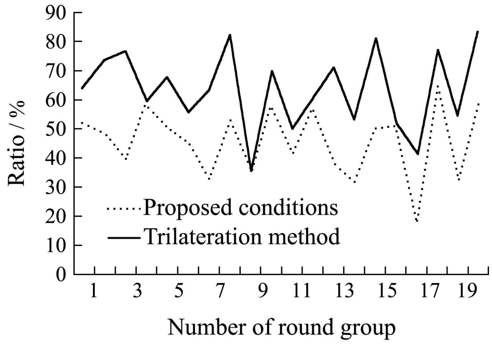


Fig. 7 Comparison of localizable ratio

6 Conclusions

In this paper, we discuss the localizability of two merging graphs of nodes. The mathematical tool is graph rigidity theory. Different from other works, the analysis object is neither the whole network nor one single node, but a set of nodes. We give localizable conditions for several specified cases of the graph. Compared with the commonly-used trilateration method, ours could reduce the risk of missing localizable nodes. The proposed conditions can also be extended to more general cases sequentially when the volume of the set increases. The analysis of these conditions between graphs can guide the localization algorithm in future work. For example, if there are some nodes lying outside the communication range of anchor graph, we could amplify the communication power of specified nodes according to the conditions given in this paper.

References:

[1] WYMEERSCH H, LIEN J, WIN M. Cooperative localization in wireless networks [J]. *Proceedings of the IEEE*, 2009, 97(2): 427 – 450.

[2] PATWARI N, ASH J, KYPEROUNTAS S, et al. Locating the nodes: cooperative localization in wireless sensor networks [J]. *IEEE Signal Processing Magazine*, 2005, 22(4): 54 – 69.

[3] JUANG P, OKI H, WANG Y, et al. Energy efficient computing for wildlife tracking: d esign tradeoffs and early experiences with Zebra net [J]. *International Conference on Architectural Support for Programming Languages and Operating Systems*. USA: ACM, 2002: 96 – 107.

[4] YANG Z, LI M, LIU Y H. Sea depth measurement with restricted floating sensors [C] //*The 28th IEEE International Real-Time Systems Symposium*. Piscataway: IEEE, 2007: 469 – 478.

[5] VOSSIEK M, WIEBKING L, GULDEN P, et al. Wireless local positioning [J]. *IEEE Microwave Magazine*, 2003, 4(4): 77 – 86.

[6] ELLINGER F, EICKHOFF R, GIERLICH R, et al. Local positioning for wireless sensor networks [C] //*IEEE Global Communications Conference*. New York: IEEE, 2007: 1 – 6.

[7] GAO J, SION R, LEDERER S. Collaborative location certification for sensor networks [J]. *ACM Transactions on Sensor Networks*, 2010, 6(4): 1 – 26.

[8] LI M, HE Y, LIU Y H, et al. Canopy closure estimates with GreenOrbs: sustainable sensing in the forest [C] //*ACM Conference on Embedded Networked Sensor Systems*. USA: ACM, 2009: 4 – 6.

[9] DIAO Y F, FU M Y, ZHANG H H. An overview of range detection techniques for wireless sensor networks [C] //*World Congress on Intelligent Control and Automation*. New York: IEEE, 2010: 1150 – 1155.

[10] PRIYANTHA N B. *The cricket indoor location system* [EB/OL]. http://nms.csail.mit.edu/papers/bodhi-thesis.pdf.

[11] ANDERSON B, BELHUMEUR P, EREN T, et al. Graphical properties of easily localizable sensor networks [J]. *Wireless Network*, 2009, 15(2): 177 – 191.

[12] EREN T, GOLDENBERG D, WHITELEY W, et al. Rigidity, computation, and randomization in network localization [C] //*IEEE International Conference on Computer Communications*. New York: IEEE, 2004: 2673 – 2684.

[13] YANG Z, LIU Y H. Understanding node localizability of wireless ad-hoc networks [C] //*IEEE International Conference on Computer Communications*. New York: IEEE, 2010: 1 – 9.

[14] DIAO Y F, FU M Y, ZHANG H S. Localizability exploration of sensor network [C] //*Chinese Automation Congress*. Beijing: Chinese Association of Automation, 2011: 1 – 6.

[15] YANG Z, LIU Y H, LI X Y. Beyond trilateration: on the localizability of wireless ad hoc networks [J]. *IEEE/ACM Transactions on Networking*, 2010, 18(6): 1806 – 1814.

[16] DIAO Y F, FU M Y, ZHANG H S. Comparison of node localization methods for sensor networks [C] //*The 12th International Conference on Control, Automation, Robotics and Vision*. New York: IEEE, 2012: 1806 – 1814.

[17] JACOBS D, HENDRICKSON B. An algorithm for two-dimensional rigidity percolation: the pebble game [J]. *Journal of Computational Physics*, 1997, 137(2): 346 – 365.

[18] YU C B, FIDAN B, ANDERSON B D O. Principles to control autonomous formation merging [C] //*American Control Conference*. New York: IEEE, 2006: 762 – 768.

[19] WANG X P, LUO J, LI S S, et al. Component based localization in sparse wireless ad hoc and sensor networks [C] //*IEEE International Conference on Network Protocols*. New York: IEEE, 2008: 288 – 297.

[20] MOORE D, LEONARD J, RUS D, et al. Robust distributed network localization with noisy range measurements[C] //*ACM Conference on Embedded Networked Sensor Systems*. New York, NY: ACM, 2004: 50 – 61.

[21] JACKSON B, JORDAN T. Connected rigidity matroids and unique realizations of graphs [J]. *Journal of Combinatorial Theory Series B*, 2005, 94(1): 1 – 29.

[22] BERG A, JORDAN T. A proof of Connelly's conjecture on 3-connected generic cycles [J]. *Journal Combinatorial Theory Series B*, 2003, 88(1): 77 – 97.

[23] JORDAN T, SZABADKA Z. Operations preserving the global rigidity of graphs and frameworks in the plane [J]. *Computational Geometry*, 2009, 42(6): 511 – 521.

作者简介:

刁英斐 (1985−), 男, 博士研究生, 目前研究方向为传感器网络中的定位问题与分布式算法, E-mail: yfdiao@gmail.com;

付敏跃 (1959−), 男, 讲席教授, 国际电子电气工程师协会会士, 目前研究方向为控制系统、信号处理和通信系统, E-mail: minyue.fu@newcastle.edu.au;

张焕水 (1963−), 男, 长江学者特聘教授, 山东省政府泰山学者, 目前研究方向为协作信号与信息处理、无线传感器网络中的目标跟踪和随机时滞系统, E-mail: hszhang@sdu.edu.cn.