Computer Networks 103 (2016) 33–43

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Tree-based coverage hole detection and healing method in wireless sensor networks

Wei Li[a,*], Yuwei Wu[b]

[a] School of Aeronautics and Astronautics, Sichuan University, No. 24, South Section 1, Yihuan Road, 610065, Chengdu, China
[b] Huawei Technologies Co., Ltd, No. 1899, Xiyuan road, 611731, Chengdu, China

ARTICLE INFO

Article history:
Received 6 August 2015
Revised 5 April 2016
Accepted 6 April 2016
Available online 9 April 2016

Keywords:
Coverage hole detection
Hole healing
Tree
Sensor network

ABSTRACT

In wireless sensor networks, coverage is a fundamental issue affecting the quality of service. A coverage hole may appear anywhere in the area being monitored at any time because of many reasons. Thus, hole detection and healing have become major challenges towards achieving perfect coverage. This study provides a novel algorithm using trees and graph theory to detect and describe the existing holes in the region of interest. Simulation results show that the tree-based method can indicate the location, size, and shape of coverage holes accurately. Based on the results for hole detection, a tree-based healing method is also proposed. The method is divided into two phases, namely, hole dissection and optimal patch position determination. Results obtained from the experimental evaluation reveal that the proposed healing method can increase the coverage rate with only a few additional sensors compared to other related methods.

1. Introduction

Wireless sensor networks (WSNs) have excellent capabilities in monitoring physical environments. For this reason, WSNs have been extensively applied in both the civil and military fields, such as habitat monitoring, industrial diagnosis, ambient assisted living, and disaster recovery. In WSNs, sensor nodes are scattered across regions of interest (ROIs) to detect events and collect information. Covering the ROI adequately and capturing events and information efficiently are desirable. A complete coverage of the region guarantees the achievement of such requirements. However, coverage holes unavoidably appear in the ROI because of many reasons, such as random deployment, sensor destruction, and energy exhaustion of nodes [1]. The existence of coverage holes dramatically affects the performance of WSNs. For example, coverage holes aggravate the transmission burden of the boundary nodes of holes, which leads to hole diffusion because the energy of nodes on hole boundaries will be consumed rapidly [2]. Therefore, coverage hole detection and healing are major issues in WSNs.

The sensing range of any node in a WSN is commonly assumed to be a disk, with which the sensor collects the information on the surroundings [3, 4], such as temperature, humidity, concentration, and radiation. Due to their nature, the values of such data change in a continuous fashion. Therefore, their values on locations which are close to sensor nodes can be derived from nearby nodes, even though these locations are not directly sensed by any node. For instance, if we obtain the temperature of the nodes on the boundary of a hole, then the temperature of nearby points inside the coverage hole can also be derived [5]. In other words, the performance damage induced by small coverage holes can be ignored in WSNs. However, conventional methods for coverage hole detection and healing always concentrate on how to achieve 100% sensing coverage, thereby incurring unnecessary consumption of resources.

In this study, we mainly focus on the issue of coverage hole detection and healing in WSNs and ignore the effects of small coverage holes on the overall performance of a network. On one hand, coverage hole detection is a fundamental problem in evaluating the quality of services in WSNs. If the information about coverage holes, such as positions, sizes, and shapes, is inaccurately detected, then we cannot provide relevant methods to patch the uncovered regions. On the other hand, deploying only a few additional sensors to reduce the uncovered areas as much as possible is desirable. Therefore, determining the optimal patch positions for healing holes is another key issue in WSNs. The main contributions of this study include: (1) We propose a tree-based method to detect and describe the existing coverage holes in WSNs. The tree can accurately indicate the locations, sizes, and shapes of coverage holes. (2) On the basis of this tree description, we propose a method to dissect a large hole into several smaller holes. We determine the optimal patch position of each small sub-coverage hole to deploy additional sensors. Thus, multiple patch positions can be

* Corresponding author. Tel.: +8613880438385.
E-mail address: aegeanwei@gmail.com (W. Li).

http://dx.doi.org/10.1016/j.comnet.2016.04.005
1389-1286/

Table 1
Comparisons of coverage hole detection methods.

Proposed solution	Distributed /Centralized	Advantages	Drawbacks
Zhang [6]	Distributed	Positions of one-hop neighbors are needed	Coverage hole described with boundary nodes
Ma [7]	Distributed	Every hole inside the WSN can be found	Requires the absolute location Hole described with boundary nodes
Qiu [8]	Distributed	Detects holes without the requirement of accurate location	Does not provide the global view of holes Hole described with boundary nodes
Liu [9]	Centralized	Describes the sizes of holes accurately	High computational complexity
Bejerano [10]	Distributed	Provides accurate size estimations of holes.	High computational complexity May fail in sparse WSN
Kroller [11]	Distributed	No need about location information of sensor nodes	Require a high density of nodes
Saukh [12]	Distributed	Applies in both sparse and dense networks	Do not provide the accurate boundaries of coverage holes
Ghrist [13], Yan [14]	Distributed	Constructs complexes to detect holes	Miss several coverage holes

simultaneously determined at each round of iteration with low complexity.

The remainder of the paper is organized as follows: Section 2 summarizes the related work on the detection, description, and healing of coverage hole. Section 3 provides the preliminaries used in the study. Section 4 describes in detail the tree-based coverage hole detection method. In Section 5, we propose a hole healing method based on trees. Section 6 includes the simulations conducted to assess the performance of the proposed method. Finally, Section 7 concludes the paper.

2. Related work

Given that coverage is one of the important issues in WSNs, a large number of studies on detecting and healing coverage holes have been previously conducted.

The most common method to detect and describe coverage holes is by directly determining the boundary nodes of the holes. If boundary nodes are recognized, the coverage holes contained by the boundary nodes can be detected. Zhang *et al.*, proposed a localized method for detecting the boundary nodes of coverage holes through the Voronoi and neighbor-embracing polygons [6]. This method can be extended to monitor irregular ROIs, but to implement it, each node needs the position information of its one-hop neighbors. Ma *et al.*, provided a geometry-based distributed hole detection algorithm to determine the coverage holes in a post-deployment scenario and used the boundary nodes to describe the coverage holes [7]. In their method, each sensor node knows its absolute location information. Qiu *et al.*, proposed a Delaunay-based coordinate-free mechanism to detect coverage holes, and the boundary nodes were used to indicate the locations of holes [8]. However, this method cannot provide the global view of holes. The preceding works used the boundary nodes to describe the coverage holes. This is a common approach in hole detection. However, this kind of methods has a drawback: the coverage hole, which is surrounded by the boundary nodes, does not describe accurately the uncovered region. The estimated sizes of the coverage holes are always larger than the actual sizes of the corresponding coverage holes. Furthermore, in several cases, the coverage holes surrounded by boundary nodes may contain several sensor nodes inside themselves, which will be discussed in the subsequent section.

Many existing approaches utilize the intersections of the sensing disks of sensors to detect and heal coverage holes. In this type of approaches, the intersections of the sensing disks of the boundary nodes of holes are calculated. The unions of such intersection points are used to describe the holes. For example, Liu *et al.*, used the intersections of sensing disks to determine the locations and sizes of holes and employed a greedy method to patch the holes [9]. Yigal introduced the concept of the cycling segment sequence, which was used to construct a localized and efficient algorithm for coverage hole detection [10]. This algorithm is also based on the intersections of sensing borders. The coverage hole description using the intersections of sensing disks can provide accurate size estimations of holes. However, the coverage hole description using the intersections requires burdensome computation and estimation. This method may be impossible to implement locally in sensors and may also be ineffective in determining coverage holes in sparse deployments of sensor nodes. In sparse deployments, isolated sensors, whose coverage regions do not intersect with any other sensors, may appear. Thus, the isolated sensors cannot work with other nodes to determine the boundary of coverage holes.

The topological properties of WSNs are also adopted in several studies to detect and describe the coverage holes. Kroller *et al.*, presented an algorithm to search for several types of patterns, so-called flowers, which were further extended and merged in the augmenting phase of the algorithm to form a boundary of the network [11]. Saukh *et al.*, provided a solution for boundary recognition that approximates the boundary of the sensor network by determining the majority of the inner nodes through geometric constructions [12]. However, these types of methods require a high density of nodes. These methods may also fail to recognize the boundaries of holes accurately. Ghrist *et al.*, introduced the nerve complex and the Rips complex to indicate hole locations. They used homology to describe the detected coverage holes [13]. Yan *et al.*, proved that coverage holes can be detected through the use of the Cech and Rips complexes. Therefore, they proposed a homology-based distributed method to determine coverage holes [14]. Topology-based methods have an evident advantage: no accurate location information of sensor nodes is needed. However, these methods may miss several coverage holes. As shown in the literature [14], a homology-based algorithm can only detect non-triangular holes with the probability of 99% and cannot detect triangular holes.

Table 1 summarizes the main advantages and drawbacks of the aforementioned coverage hole detection methods. To avoid the drawbacks, we present a distributed method to detect all types of coverage holes and provide a tree-based description method to determine the locations and sizes of coverage holes accurately.

Several studies simply focused on the solutions of coverage hole healing. Wu *et al.*, proposed an iterative Delaunay triangulation-based method to eliminate existing coverage holes [15]. In their method, an optimal patch position was obtained in each iteration. A viable solution involves enhancing several sensors with mobile capability; then, the mobile sensors are relocated to heal the holes. For instance, Shen *et al.*, partitioned the area of a WSN into a number of grids and developed a method to move the sensors to the desired grids, thereby meeting the optimization requirements [16]. Wang *et al.*, provided a bidding protocol for mobile sensor deployment to achieve a balance between sensor coverage and sensor cost [4]. This protocol chooses the farthest Voronoi vertex from sensors as the target location of the additional sensor. Wang *et al.*, proposed a deterministic method to achieve a multilevel coverage

Table 2
Comparisons of coverage enhancing and hole healing methods.

Proposed solution	Distributed /Centralized	Advantages	Drawbacks
Wu [15]	Centralized	Suitable in the environments with obstacles	Only one patch position is obtained; Computational complexity is $O(n^2\log(n))$
Shen [16]	Centralized	Provides generic framework for sensor redeployment	ROI should be partitioned into several grids
Wang [4]	Distributed	Designs bidding protocols to balance cost and coverage	Requires a global computation
Wang [17]	Centralized	Uses probabilistic sensor model for the sensor placement	Applied in deterministic sensor deployment
Mahboubi [18]	Distributed	Increases coverage of a field with different coverage priorities	High message complexity
Liu [19]	Distributed	The connectivity of the WSN is guaranteed	Only applied in grid coverage
Senouci [20]	Distributed	Estimates and enhances coverage with little cost	A high node density is required.

of the ROI by solving the k-coverage sensor deployment problem [17]. Mahboubi *et al.*, utilized a multiplicatively-weighted Voronoi diagram to reduce the sizes of coverage holes by instructing the sensors to move in the correct direction [18]. This diagram highly relies on the communications among nodes. Liu presented an algorithm based on ant colony optimization with three classes of ant transitions to solve the grid coverage of WSN with minimum cost and guaranteed connectivity [19]. Senouci proposed a distributed virtual forces-based local healing approach in a mobile WSN [20]. In the method, in order to generate the virtual forces, a high node density is required.

The main advantages and drawbacks of the aforementioned healing methods are summarized in Table 2. Although there are a number of distributed healing methods have been proposed, the optimal patch positions of additional sensors cannot be always obtained, because the nodes make decisions based on the local information. Therefore, the distributed methods always utilize a great number of sensors to heal the holes. Centralized methods could find the optimal candidate locations of additional sensors. However, such methods always find only one patch positions at each round of computation, which leads to unnecessary computations. In this paper, we report a novel healing method based on trees. This novel method does not require partitioning of the ROI. The proposed healing method can also determine multiple optimal patch positions simultaneously at each round of iteration with low complexity.

3. Preliminaries

3.1. Assumptions

We assume that sensors are randomly deployed in a 2D ROI. We consider the coverage holes inside the sensor network rather than the big outer holes. The determination of boundary holes heavily relies on the absolute coordinate of sensor nodes. However, in this paper, we assume that each sensor node can only obtain the relative positions of its neighboring nodes. Also, we assume that there is no error in position determination of neighboring nodes. However, the errors of the neighboring nodes localization always exist in practice. The errors in relative position and angle determination lead to the following two unpleasant scenarios: (1) several covered regions are mislabeled as coverage holes, whereas (2) several uncovered regions are mistaken as sensed regions. In this study, we use the lower bound of the irregular sensing range to represent the sensing radius of nodes. The sensing range of nodes in practice is slightly greater than the value used in the sensing model. Therefore, if the error of neighboring node estimation is within a small range, then the missed uncovered regions are likely to be detected by the nodes in practice. Therefore, we provide a conservative estimation of coverage in consideration of mislabeled coverage holes. This method is preferred in many applications, particularly in security-critical applications.

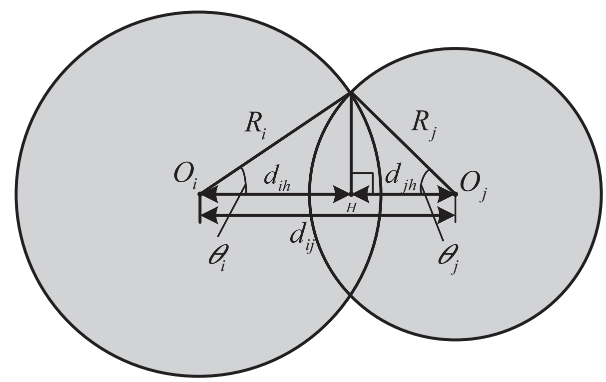

Fig. 1. Overlapped area of two disks.

3.2. Sensing model

A disk is commonly used to represent the sensing range of a sensor node. The sensing capability is uneven in different directions around a sensor node [21]; for this reason, several researchers use probabilistic models [22] and irregular polygons [23] to indicate the sensing range of nodes. Given that we can determine the lower bound of the irregular sensing range [24], any event within the bound is also detected with probability of 1.

3.3. Delaunay triangulation

We construct a Delaunay triangulation to discover the topological properties of a WSN. Delaunay triangulation is an important data structure in computational geometry, which satisfies the empty circle property. That is, for each side in Delaunay triangulation, we can determine a circle passing through the endpoints of the side without enclosing other points [15]. In this study, we adopt a distributed and localized coordinate-free method to construct the Delaunay triangulation [8], which relies on the relative positions and angles of neighboring nodes.

3.4. Area union of two intersecting disks

The area union of two intersected disks equals the area sum of two disks minus the area of the intersected part. As shown in Fig. 1, disk O_i intersected with disk O_j. R_i and R_j are the radii of the two disks respectively. d_{ij} is the distance between the centers of the two disks. The area union S_{ij} of the two disks can be computed as follows:

$$d_{ih} = \frac{d_{ij}^2 + R_i^2 - R_j^2}{2d_{ij}}, \quad d_{jh} = \frac{d_{ij}^2 + R_j^2 - R_i^2}{2d_{ij}}, \tag{1}$$

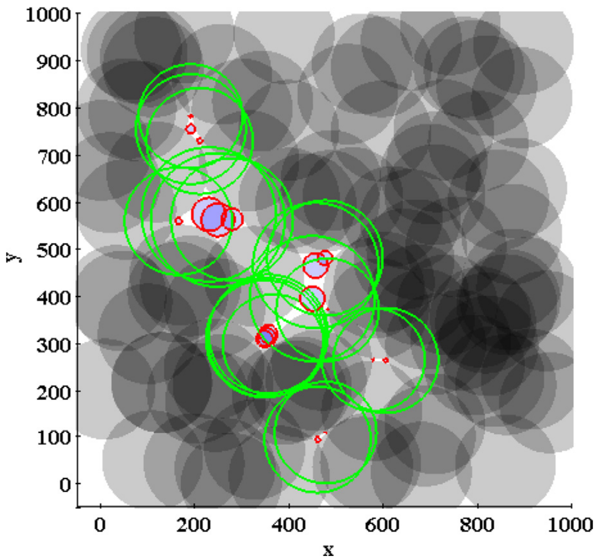

Fig. 2. Results of hole detection through empty circles.

$$\theta_i = \arccos \frac{d_{ih}}{R_i}, \quad \theta_j = \arccos \frac{d_{jh}}{R_j}, \tag{2}$$

$$S_{ij} = \pi R_i^2 + \pi R_j^2 - 2 \cdot (R_i^2 \theta_i - R_i d_{ih} \sin\theta_i + R_j^2 \theta_j - R_j d_{jh} \sin\theta_j). \tag{3}$$

4. Tree-based coverage hole detection

In this section, we propose a novel tree-based method to detect and localize coverage holes in a WSN. The proposed method consists of four phases, as follows: (1) Detection of coverage holes: every node detects whether coverage holes exist around it based on a hole detection algorithm. (3) Merging of coverage holes: a merging method of holes is provided to present the global view of a coverage hole by indicating the location, and shape of an isolated coverage hole. (2) Size estimation of local coverage holes: we use the inscribed empty circles (IECs) defined in Section 4.3 to estimate the size of every local coverage hole. (4) Tree description: an isolated coverage hole is described as a tree.

4.1. Detection of coverage holes

In this phase, we introduce empty circles to detect the uncovered regions in the region of interest. The empty circles are defined as the circum-circles of the Delaunay triangles. Based on the properties of Delaunay triangulation, no other nodes exist inside an empty circle. Thus, we can definitely determine an empty circle with a radius greater than R_s if an uncovered region exists. We compare the radius R_e of each empty circle with the sensing radius R_s of the sensors. If $R_e > R_s$, uncovered regions must exist in the empty circle. Therefore, the problem of hole detection switches to the problem of determining the empty circles with radius greater than R_s. If we can determine all the empty circles with $R_e > R_s$, then the coverage hole in the ROI is exhaustively identified.

We use an example to illustrate the process of coverage hole detection, which is shown in Fig. 2. All the empty circles with radii greater than R_s are plotted, which are shown as green circles in Fig. 2. If an uncovered region exists, at least one empty circle with $R_e > R_s$ exists. In Fig. 2, all the empty circles with $R_e > R_s$ are detected, whereas the uncovered regions in the ROI are assuredly contained in the empty circles.

4.2. Merging of coverage holes

The empty circles can show the locations of coverage holes locally. However, the isolated empty circles cannot provide the global view of coverage hole shapes. In this stage, we propose a method to merge the isolated empty circles, which are classified as belonging to the same coverage hole. Any two isolated empty circles that are generated from a pair of neighboring Delaunay triangles have a common side. If the length of the common side is greater than $2R_s$, then the two isolated empty circles are classified as the same coverage hole. When the length of the common side is less than $2R_s$ and the centers of the isolated empty circles are located at the same part of the common side, the local coverage holes indicated by the two neighboring empty circles can be merged into one. Otherwise, the two local coverage holes are classified as different holes. We provide the pseudocode of the hole merging process, as shown in Fig. 3.

4.3. Size estimation of a local coverage hole

Although the empty circles can reveal the uncovered regions in the ROI, they are not the same as the coverage holes. Several areas in the empty circles with $R_e > R_s$ are sensed by the sensor nodes, such as the gray areas in the green empty circles in Fig. 2. Therefore, the sizes of empty circles are larger than the corresponding coverage holes. In this study, we introduce the IECs to estimate the sizes of coverage holes. An IEC is concentric with its corresponding empty circle, and its radius $R_{ie} = R_e - R_s$, where $R_e > R_s$. Thus, the IEC contains some of the uncovered region in a coverage hole. In Fig. 2, the red circles represent the IECs. We use the union of IECs, which are classified as a coverage hole, to indicate the size of the hole. The detailed process of size estimation is shown as a pseudocode in Fig. 4. We use an example to illustrate the process of coverage hole merging and size estimation, as shown in Fig. 5

Fig. 5 shows a part of a typical coverage hole. The gray disks indicate the sensing ranges of sensors, whereas the blank circles represent the IECs. The dotted lines are the Delaunay triangles

```
If the ith and jth Delaunay triangles have a common side S_ij
  l_ij the length of the common side S_ij;
  If l_ij > 2R_s
    The empty circles generated from the ith and jth Delaunay triangles are classified as the
    same coverage hole;
  Else if l_ij > 2R_s and the centers of the two empty circles are located at the same side of S_ij
      The two empty circles are classified as the same coverage hole;
  End If
End If
```

Fig. 3. Procedure of coverage hole merging.

```
A coverage hole contains n IECs;
The size of coverage hole S is initiated as 0;
For i = 1 to n − 1
  For j = i + 1 to n
    The distance d_ij between the centers of the ith IEC and the jth IEC is computed;
    R_iei is the radius of the ith IEC;
    R_iej is the radius of the jth IEC;
    If d_ij < R_iei + R_iej
      The area union S_ij of the ith IEC and the jth IEC is computed with the method shown in Section 3.4.
    Else
      The area union S_ij of the two IECs is πR_iei^2 + πR_iej^2
    End If
    S = S + S_ij
  End For
End For
```

Fig. 4. Procedure of hole size estimation.

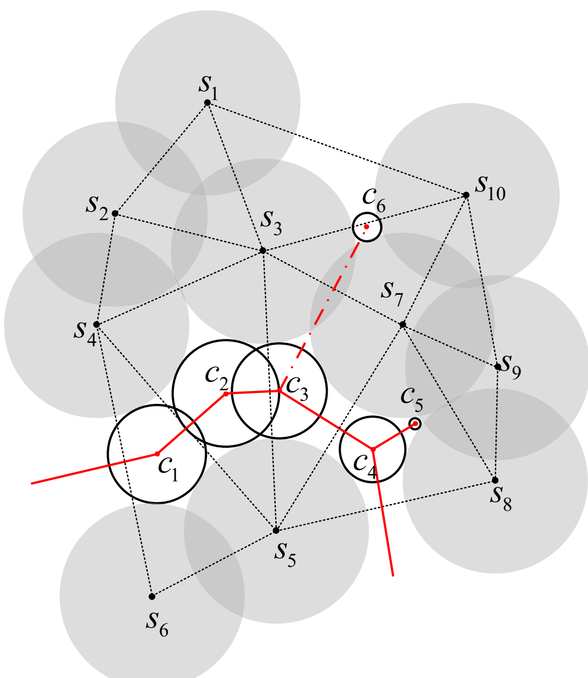

Fig. 5. Isolated IECs merging into the same coverage hole.

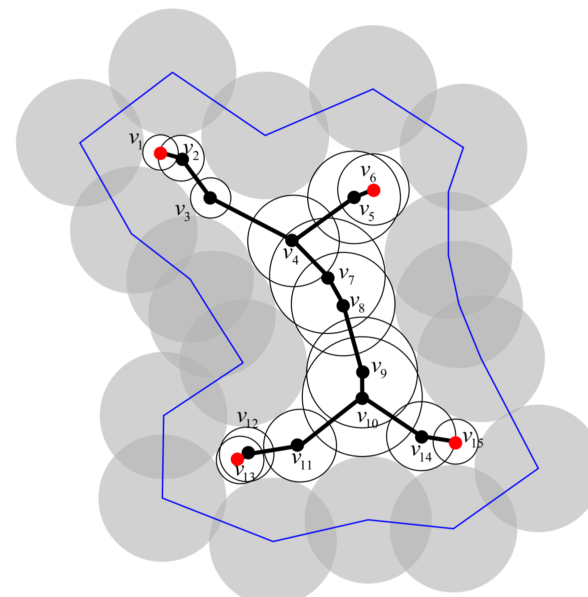

Fig. 6. A coverage hole described with a tree.

formed by the sensors. We use the symbol "Δ" to denote a triangle. For example, the triangle formed by s_1, s_2 and s_3 is denoted as $\Delta s_1 s_2 s_3$. Given that $\Delta s_4 s_5 s_6$ and $\Delta s_4 s_5 s_3$ are neighbors and the length of the common side $s_4 s_5$ is greater than $2R_s$, IECs c_1 and c_2 are classified as the same coverage hole. In the same manner, we can infer that IECs c_3 and c_4 are also contained in the same hole. For IECs c_4 and c_5, the common side of $\Delta s_5 s_7 s_8$ and $\Delta s_7 s_8 s_9$ is less than $2R_s$. However, the centers of the two IECs are located at the left part of the common side. Thus, c_5 is also a part of the coverage hole. However, c_3 and c_6 are classified as two different coverage holes because the centers of IECs c_3 and c_6 are located in two different parts of the common side of the neighboring Delaunay triangles $\Delta s_3 s_5 s_7$ and $\Delta s_{10} s_3 s_7$, which are less than $2R_s$. Therefore, two isolated coverage holes, namely, $\{c_1, c_2, c_3, c_4, c_5\}$ and $\{c_6\}$, are obtained by using the proposed method.

4.4. Tree description

After coverage hole merging, we can obtain all the isolated holes, which are indicated by the groups of IECs. For each isolated coverage hole, we use line segments to connect the centers of each pair of IECs, which are inside the hole. In most cases where no sensors are isolated from the others, a group of line segments that are contained in a hole is similar to a tree, as shown in Fig. 6. If we can recognize a separated tree, the corresponding coverage hole that contains the tree can be exclusively determined. We can use trees to describe the corresponding coverage holes. Thus, the proposed coverage hole detection method is called tree-based hole detection method (THD).

5. Coverage hole healing based on trees

In this section, we propose a tree-dissect based coverage hole healing method (THH), which is based on the results on coverage hole detection of the previous section. We initially dissect a large coverage hole into sub-holes and determine multiple locations that are the positions far away from sensor nodes, in the large hole. Then, we deploy the additional sensor nodes on the positions of the locations in each sub-hole. The majority of the existing methods can obtain only one optimal position to be repaired in a hole [9, 25]. By contrast, our proposed healing method can find several patch positions simultaneously at each round of iteration, and deploy additional sensors on such positions simultaneously to repair coverage holes.

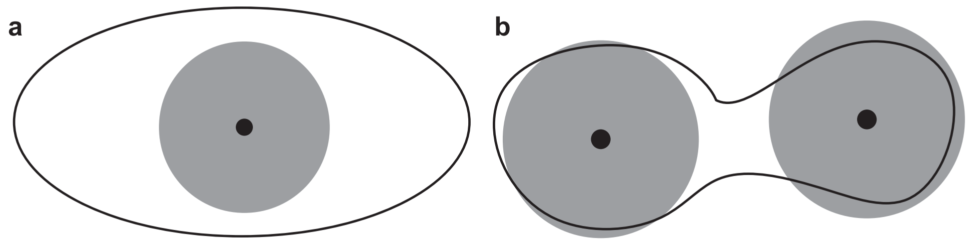

Fig. 7. (a) A convex hole; (b) a non-convex hole.

5.1. Coverage hole dissection

In general, the additional sensors should be dispatched at the centers of the coverage holes to overlap with the sensed area as little as possible. However, the shapes of the coverage holes are always irregular. In several extreme cases, the geometric center of a coverage hole may be located in the sensed regions. Thus, deploying the sensors at the centers of holes cannot always achieve satisfying results. For example, if the shape of the coverage hole is convex, then the sensor should be deployed at the center of the hole, as shown in Fig. 7(a). However, if the shape of the coverage hole is non-convex, then the sensors should be deployed at the centers of two convex parts, as shown in Fig. 7(b).

Therefore, the aim of coverage hole healing suggests that dissecting a non-convex hole into several small convex holes is desirable. Hole dissection intends to obtain multiple locations that should be repaired simultaneously. First, we provide a definition that will be used in the description of the dissection method.

A **boundary IEC** is defined as follows: In a tree, the IECs that are connected with only one other IEC are defined as **boundary IECs**. For example, a coverage hole is represented by a tree, which is illustrated in Fig. 6. In Fig. 6, the black bold lines indicate a tree that can be used to describe a coverage hole. The vertices are the centers of the IECs. The red vertices are the centers of **boundary IECs**.

Then, we can propose a method to dissect a large coverage hole into several small holes. The steps of coverage hole dissection are as follows:

Step 1: For a complete tree, a list is made to store the IECs according to their sizes.
Step 2: The largest IEC is removed from the list and is set as the origin to construct a new sub-tree.
Step 3: The IECs that are adjacent to the origin are removed from the list and added to the sub-tree and are set as the **boundary IECs**.
Step 4: If the radius of an IEC still in the list is smaller than that of its corresponding adjacent **boundary IEC**, then the IEC is removed from the list and added to the sub-tree and is set as the new **boundary IEC** of the sub-tree.
Step 5: Step 4 is repeated until no other IEC is added to the tree. The construction of a sub-tree is completed.
Step 6: Steps 2 to 5 are repeated until the list becomes empty.

We also use an example shown in Fig. 8 to illustrate our proposed method. First, we assign the largest IEC v_{10} as the origin of a new sub-tree. Then, we add the neighboring IECs v_9, v_{11}, and v_{14} to the sub-tree. All of the three IECs are set as **boundary IEC**s. Given that the radius of IEC v_{11} is greater than that of its neighboring IEC v_{12}, we add IEC v_{12} to the sub-tree. In the same manner, the IECs v_{13}, v_{15}, and v_8, are also in the same sub-tree. Although IEC v_8 is adjacent to IEC v_7, the radius of IEC v_8 is less than that of IEC v_7. Since v_7 is the largest IEC remaining from the original list, IEC v_7 is set as an origin of another sub-tree. The preceding process is repeated. Then, we can dissect the coverage hole into three sub-trees, namely, $\{v_{10}, v_9, v_{11}, v_{14}, v_{12}, v_{15}, v_8, v_{13}\}$, $\{v_7, v_4, v_5, v_6, v_3\}$, and $\{v_1, v_2\}$.

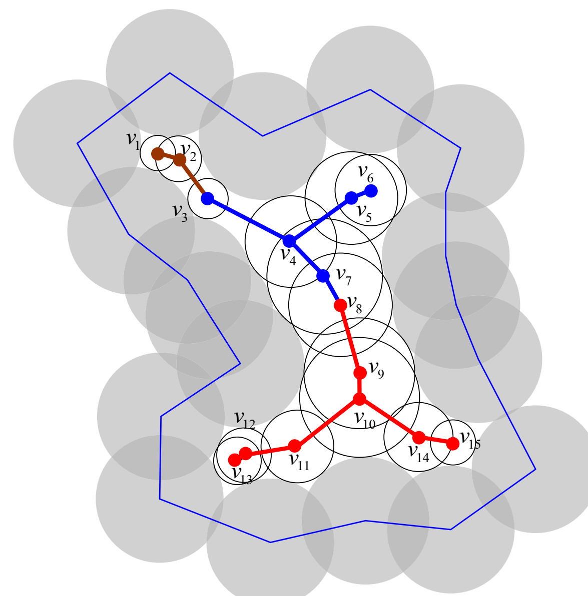

Fig. 8. Results of hole dissection.

5.2. Optimal patch position determination

As mentioned previously, IECs in a tree represent the size of the corresponding hole. Deploying additional sensors at the center of the large IECs and avoiding overlap with the ranges of other existing sensors are desirable. Our proposed coverage hole dissection method ensures that any two largest IECs in two different subtrees are not adjacent to each other. Thus, the center of the largest IEC in each sub-tree can be regarded as the position that should be patched. For the preceding example, we dissect the coverage hole into three sub-trees. IECs v_9, v_7, and v_2 are the largest IECs in the corresponding sub-trees. Therefore, the additional sensors should be preferentially deployed at the centers of these three IECs. We use three different colors in filling the corresponding IECs to show the results clearly, as shown in Fig. 9. The optimal patch positions of the additional sensors are not adjacent to each another, thereby increasing the efficiency of coverage hole healing. After the additional sensors deployment, new hole trees are obtained. Then, we repeat the above the process to deploy additional sensors on the centers of the updated largest IECs. In several sub-trees, the radii of the largest IEC are still small. If we indiscriminately deploy additional sensors on such positions, resource consumption for healing small coverage holes in the global view undoubtedly increases. Therefore, we set a threshold to avoid such cases: When the radius of an IEC is larger than the threshold, the IEC can be selected to be the potential patch position; otherwise, the IEC cannot be

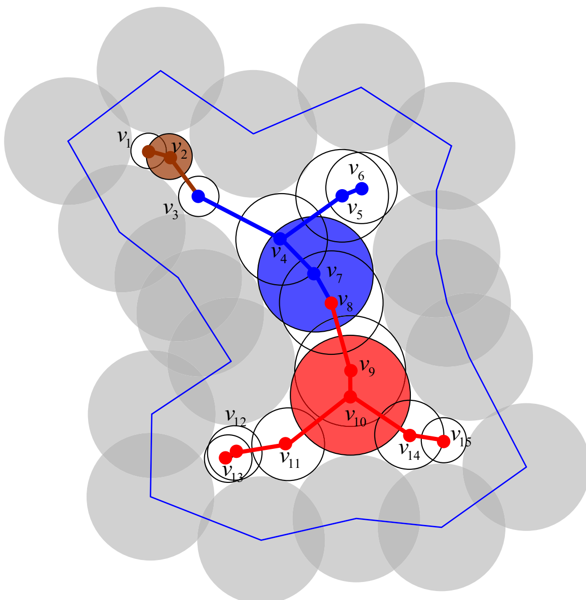

Fig. 9. Results of optimal patch position determination.

selected. The threshold can be set as a constant or a variable that corresponds to the purpose of the application.

5.3. Complexity analysis

The computational complexity of Delaunay triangulation construction is $O(bn)$ [8], where n is the number of sensors in the network and b is the number of nearby sensors in each node. In the coverage hole detection phase, the sensing radius of the nodes is compared with the radius of the empty circle, which originated from the Delaunay triangles. Thus, the computational complexity of this phase is $O(n)$. In the hole merging phase, we should determine whether any two neighboring IECs have a common side and compute the length of the side. Since there are totally $O(b)$ common sides, the worst computational complexity of such process is $O(bn)$. In the size estimation phase, we should determine whether any two IECs in a coverage hole have overlapped regions. Since the IEC only needs to execute size estimation with its neighboring IECs, the worst complexity of size estimation is $O(bn)$. The complexity of the tree description is $O(n)$. Therefore, the total worst computational complexity of coverage hole detection is $O(bn)$, if we assume that $b << n$.

The healing of coverage holes has two stages, namely, (1) coverage hole dissection and (2) optimal patch position determination. The computational complexities of both coverage hole dissection and optimal patch position determination are $O(n)$. Thus, the total worst computational complexity of coverage hole healing is $O(n)$.

6. Performance evaluation

In this section, we execute simulations in MATLAB to assess the performance of the tree-based coverage hole detection and healing method. We use the simulation setting previously described in [14]: A total of 80 sensors are randomly deployed in a 100 m × 100 m rectangle ROI, and the sensing range of the sensor nodes is 10 m. Similar to the performance evaluation method in [7], the sensing range also varies, which will be described in detail in the subsequent section.

6.1. Results of coverage hole detection

We compare the results of the proposed tree-based method with those of the boundary node detection of [7]. The coverage holes in this method are described by the nodes on the boundaries of holes. The simulation results are shown in Fig. 10. Fig. 10(a) shows the results of the boundary node detection method. All of the boundary nodes are denoted as red points. We connect boundary nodes with blue lines in counterclockwise order to provide the global view of holes. The regions bounded in the blue polygons are the estimated coverage holes. Fig. 10(b) shows the results of our proposed tree-based method. Each red curve comprised of ling segments indicates an isolated coverage hole. The blue disks indicate the sizes of the corresponding estimated coverage holes. Both methods can evidently detect all the holes in the regions. However, the boundary node detection method has severe flaws. For instance, two sensor nodes, denoted by red circles in Fig. 10(a), are included in the estimated coverage holes. This observation contradicts the definition of coverage holes. In our proposed method, the number of coverage holes can be easily estimated, and each tree is located in the uncovered region of the corresponding hole. From this perspective, the performance of our coverage hole detection method is better than that of the conventional boundary node detection method. A tree is also a minimally connected graph. Thus, if we use the trees to describe the properties of coverage holes, we

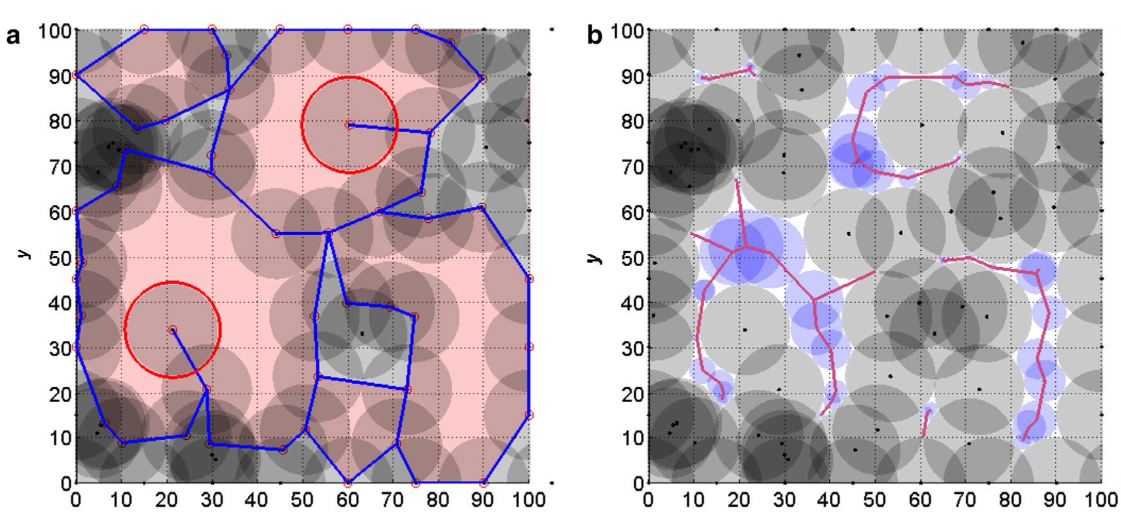

Fig. 10. (a) Coverage hole detection via boundary nodes; (b) hole detection via trees.

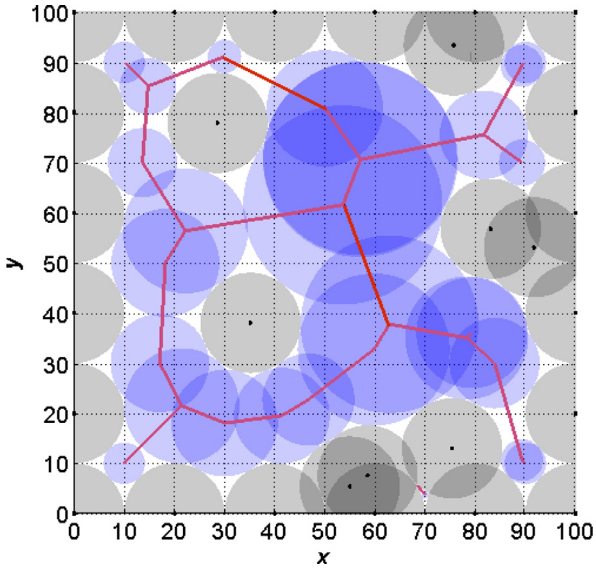

Fig. 11. Result of the proposed method in sparse deployments of nodes.

can save several resources of storing and transmitting. For an coverage hole, if the boundary node detection method needs n nodes and n edges to describe the hole, our proposed method only needs $n-2$ nodes and $n-3$ edges to do the same job.

From another perspective, the tree-based method can accurately identify the real extend of the uncovered regions of holes compared with the boundary node detection method. In Fig. 10(a), the regions bounded by the boundary nodes are regarded as the sizes of the coverage holes, which are shown as red polygons. In the tree-based method, the sizes of the coverage holes are the unions of isolated IECs, which are shown as blue circles in Fig. 10(b). We can estimate the real sizes of coverage holes through the Monte Carlo method. For the 100 m × 100 m ROI, 1,000,000 points are evenly distributed. If a point is located in the sensing range of any sensor, then the point is regarded as a covered point; otherwise, the point is regarded as an uncovered point. We can use the ratio of the uncovered points to the total number of points to indicate the sizes of the coverage holes. In this case, the real size of coverage holes is approximately 16.5% of the ROI. In the tree-based method, the estimated size of coverage holes is approximately 13.2% of the ROI. The sizes of holes described by boundary nodes occupy 73.3% of the ROI because the boundary nodes are not located on the real boundaries of uncovered regions. The estimated coverage holes bounded by the boundary nodes contain a large number of covered regions. The tree-based method uses the IECs to determine the sizes of coverage holes located in the uncovered regions. Thus, our proposed method exhibits better performance in size estimation of holes than the conventional boundary node detection method.

In order to show the effectiveness of our proposed method, we also present results for the sparse deployments. As shown in Fig. 11, it can be observed that the location and shape of holes are clearly detected. Since two sensors are isolated from other nodes in the network, there are loops in the graph. We can still use the our proposed THD and THH methods to detect and heal the holes. Therefore, in this paper, we regard this kind of hole tree as a special case of trees. The sizes of holes described by boundary nodes occupy 93.3% of the ROI, and in our method, the estimated hole size is 56.6%. The real size of coverage hole is 61.2% of the ROI. Therefore, our proposed method could estimate the size of holes more accurately, when the sensors are sparsely deployed inside WSN. We also use different radii of the sensing ranges to verify the effectiveness of our proposed method. We let the radius of the sensing range change from 1 m to 10 m. We run 30 rounds of simulations with random deployments of sensor nodes for each radius to obtain the average. The results of hole size estimation are shown in Fig. 12. The estimated uncovered regions decrease with the increase of the radius. However, the hole sizes obtained by the tree-based method is closer to the real sizes of the holes than that of the boundary node detection method.

We also assess the performance of our tree-based hole detection method on the input topologies used in [26]. In these topologies, a large number of sensor nodes are deployed in the ROI. The irregular shapes of coverage holes are artificially designed. Fig. 13 shows the results of the tree-based method. The red lines are the trees, whereas the blue disks are the IECs that indicate the sizes of the coverage holes. The shapes and locations of the intricate coverage holes are also detected.

6.2. Results of coverage hole healing

In this section, we assess the performance of our proposed hole healing method, which is based on the results of the tree-based detection and description method. For the hole detection results shown in Fig. 10, the covered area is approximately 83.5% of the ROI. In utilizing our healing method, 11 additional sensors should be deployed in the ROI, and the coverage rate of the ROI increases to 95.5%. The result of healing is shown in Fig. 14. Small coverage holes still exist in the ROI because the threshold is set as 0.5 m, as discussed in Section 5. The farthest distance between any uncovered point and its nearest sensor is less than or equal to 10.5 m in the simulation. As mentioned previously, the values of the continuous parameters that should be measured can be derived easily at the uncovered point 10.5 m away from the sensor when the values

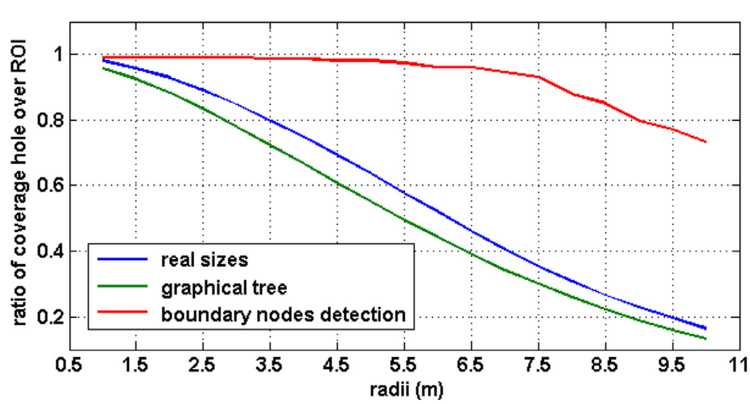

Fig. 12. Coverage rate in different radii of sensing ranges.

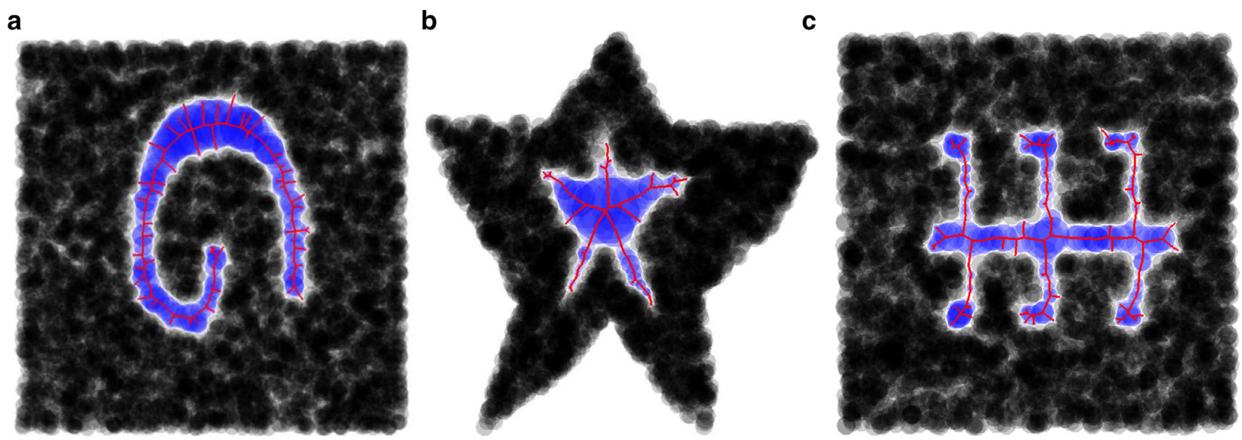

Fig. 13. Performance of the proposed method in irregularly shaped holes: (a) spiral shape, (b) star shape, and (c) cubical shape.

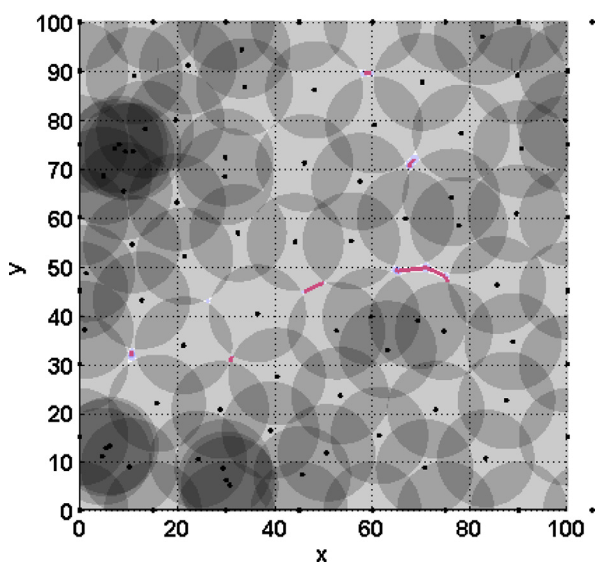

Fig. 14. Results of coverage hole healing.

at the point 10 m away from the sensor are known. We can also use the interpolation method to estimate the value. The criteria of small hole identification may be different among WSNs because of the differences in the applications and purposes of surveillance. Therefore, the threshold can be adjusted based on the purpose of the applications.

We take the Delaunay triangulation-based [15] and the Voronoi-based [4] coverage hole healing methods as references to evaluate the performance of our hole healing method. Fig. 15 shows the coverage improvements as the number of sensors increases in the three methods. The coverage of the ROI increases to 99% when 16 additional sensors are deployed in the proposed method. With the same number of additional nodes, the Delaunay triangulation-based method increases the coverage to 97.63%, and the Voronoi-based method increases to 97.43%. Therefore, our proposed method is a resource-saving method compared with the other two methods. We also observe that the performances of the Voronoi-based method and the Delaunay triangulation-based method are almost the same, and at the beginning of sensor increase, coverage improvements in the two methods are better than that in the proposed method. The reason is that Delaunay triangulation-based methods always determine the center with the largest radius as the best patch position, and the Voronoi-based method always chooses the farthest Voronoi vertex as the location of the additional sensor. However, overlaps between the sensing ranges of the additional sensors emerge with the increase in the number of sensors in the two methods. In our healing method, the patch positions of the additional sensors are kept apart from each other as much as possible through the dissection of holes.

Furthermore, since our healing method could find 7 patch locations of additional sensors in the first round of computation and 9 patch locations in the second round of computation, it can achieve a 97% coverage through only two rounds of computations. Compared with our proposed method, only one patch position is obtained in each round of computation in the other two methods.

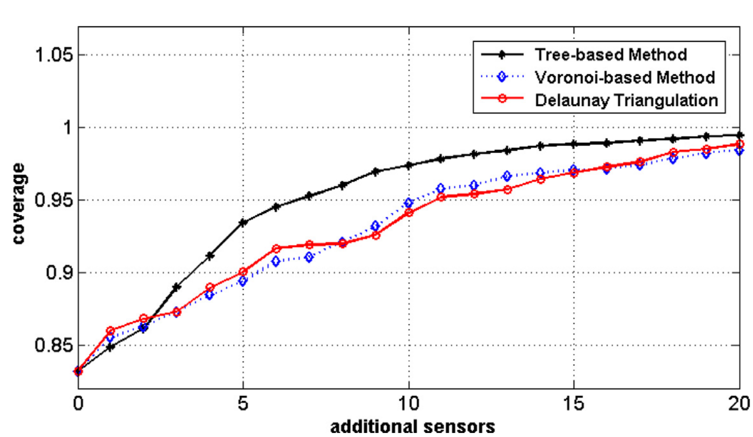

Fig. 15. Relationship between additional sensors and coverage rate.

In other words, the other two methods need 16 rounds of computations. Therefore, our proposed method exhibits a satisfactory performance in healing coverage holes.

7. Conclusions

In this study, the problem of coverage hole detection and healing is considered. We introduce the concept of trees to detect, locate, and describe coverage holes. Based on the tree, we can determine the location and shape of the corresponding coverage hole and estimate its size. The tree-based method provides the global view of holes clearly and estimates the size of holes more accurately compared with the conventional boundary node detection method. We also proposed a coverage hole healing method based on the results of hole detection. The healing method is divided into two phases, namely, hole dissection and optimal patch position determination. These two phases ensure that the overlap between the sensing ranges of the additional sensors becomes as little as possible. Compared with the Delaunay triangulation-based and the Voronoi based methods, the proposed healing method can increase the coverage rate efficiently with the same number of additional sensors. Therefore, the proposed scheme serves as a useful tool for detecting and healing coverage holes. Future studies should address the coverage problem in directional sensor networks, such as camera networks.

Acknowledgement

This work was supported by “National Natural Science Foundation of China, No. 61403065”, “Project of Sichuan Science and Technology Bureau, No. 2015JY0084”, and “Doctoral Fund of Ministry of Education of China, No. 20120185120034”.

References

[1] I. Khan, H. Mokhtar, H. Merabti, An overview of holes in wireless sensor network, in: Proceedings of the 11th Annual Postgraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting, 2010 Liverpool, UK.
[2] F. Li, B. Zhang, J. Zheng, Geographic hole-bypassing forwarding protocol for wireless sensor networks, IET Commun. 5 (6) (2011) 737–744.
[3] T. Lee, C. Qiao, M. Demirbas, J. Xu, ABC: a simple geographic forwarding scheme capable of bypassing routing holes in sensor networks, Ad Hoc Netw. 8 (2010) 361–377.
[4] G. Wang, G. Cao, P. Berman, T.F. La Porta, Bidding protocols for deploying mobile sensors, IEEE Trans. Mobile Comput. 6 (5) (2007) 563–576.
[5] W. Li, A novel graphic coverage hole description in wireless sensor networks, IEEE Commun. Lett. 18 (12) (2014) 2205–2208.
[6] C. Zhang, Y. Zhang, Y. Fang, Localized algorithms for coverage boundary detection in wireless sensor networks, Wireless Netw. 15 (1) (2009) 3–20.
[7] H. Ma, P.K. Sahoo, Y.W. Chen, Computational geometrical based distributed coverage hole detection protocol for the wireless sensor networks, J. Netw. Comput. Appl. 34 (2011) 1743–1756.
[8] C. Qiu, H. Shen, A delaunay-based coordinate-free mechanism for full coverage in wireless sensor networks, IEEE Trans. Parallel Distributed Syst. 25 (4) (2014) 828–839.
[9] J. Liu, Y. Feng, H. Xu, et al., Greedy approximation algorithm of patching hole in wireless sensor networks, in: 2nd IEEE International Conference on Consumer Electronics, Communications and Networks, 2012, pp. 2604–2608.
[10] Y. Bejerano, Coverage verification without location information, IEEE Trans. Mobile Comput. 11 (4) (2012) 631–643.
[11] A. Kroller, S.P. Fekete, D. Pfisterer, S. Fischer, Deterministic boundary recognition and topology extraction for large sensor networks, in: Proceedings of the 17th ACM–SIAM Symposium on Discrete Algorithm, 2006, pp. 1000–1009.
[12] O. Saukh, R. Sauter, M. Gauger, P.J. Marron, On boundary recognition without location information in wireless sensor networks, ACM Trans. Sensor Netw. 6 (3) (2010) 1–35.
[13] R. Ghrist, A. Muhammad, Coverage and hole-detection in sensor networks via homology, in: Proceedings of the 4th International Conference on Information Processing in Sensor Networks, 2005, pp. 254–260.
[14] F. Yan, A. Vergne, P. Martins, L. Decreusefond, “Homology based distributed coverage hole detection in wireless sensor networks,” IEEE/ACM Trans. Netw. doi:10.1109/TNET.2014.2338355.
[15] C.H. Wu, K.C. Lee, Y.C. Chung, A Delaunay triangulation based method for wireless sensor network deployment, Comput. Commun. 30 (2007) 2744–2752.
[16] Z. Shen, Y. Chang, H. Jiang, Y. Wang, Z. Yan, A generic framework for optimal mobile sensor redeployment, IEEE Trans. Vehicular Technol. 59 (8) (2010) 4043–4057.
[17] Y.C. Wang, Y.C. Tseng, Distributed deployment schemes for mobile wireless sensor networks to ensure multilevel coverage, IEEE Trans. Parallel Distributed Syst. 19 (9) (2008) 1280–1294.
[18] H. Mahboubi, J. Habibi, A.G. Aghdam, K. Sayrafian-Pour, Distributed deployment strategies for improved coverage in a network of mobile sensors with priortitized sensing field, IEEE Trans. Ind. Inf. 9 (1) (2013) 451–461.
[19] X. Liu, Sensor deployment of wireless sensor networks based on ant colony optimization with three classes of ant transitions, IEEE Commun. Lett. 16 (10) (2012) 1604–1607.
[20] M.R. Senouci, A. Mellouk, K. Assnoune, Localized movement-assisted sensor deployment algorithm for hole detection and healing, IEEE Trans. Parallel Distributed Syst. 25 (5) (2014) 1267–1277.
[21] R. Tan, G. Xing, B. Liu, J. Wang, X. Jia, Exploiting data fusion to improve coverage of wireless sensor networks, IEEE/ACM Trans. Netw. 20 (2) (2012) 450–462.
[22] V. Akbarzadeh, C. Gagne, M. Parizeau, M. Argany, M.A. Mostafavi, Probabilistic sensing model for sensor placement optimization based on line-of-sight coverage, IEEE Trans. Instrum. Meas. 62 (2) (2013) 293–303.
[23] A. Boukerche, X. Fei, A coverage-preserving scheme for wireless sensor network with irregular sensing range, Ad Hoc Netw. 5 (2007) 1303–1316.
[24] B. Liu, O. Dousee, P. Nain, D. Towsley, Dynamic coverage of mobile sensor networks, IEEE Transactions on Parallel and Distributed Syst. 24 (2) (2013) 301–311.
[25] B. Wang, H.B. Lim, D. Ma., A survey of movement strategies for improving network coverage in wireless sensor networks, Comput. Commun. 32 (2009) 1427–1436.
[26] Y. Wang, J. Gao, J.S.B. Mitchell, Boundary recognition in sensor networks by topological methods, in: Proceedings of the 12th Annual International Conference on Nobile Computing and Networking, ACM, New York, NY, USA, 2006, pp. 122–133.

Wei Li received the BS and PHD degrees from School of Mechatronical Engineering, Beijing Institute Technology, China, in 2003 and 2008 respectively. He joined University of Electronic Science and Technology of China in 2008. From 2010 to 2011, he was a postdoctoral member at Center of Industrial Electronics, Polytechnic University of Madrid, Spain. In 2011, he became an associate professor in University of Electronic Science and Technology of China. Currently, He is an associate professor at School of Astronautics and Astronautics in Sichuan University, China. His research interests include quality of service of wireless sensor networks, target tracking in camera sensor networks. He has published more than 20 peer-reviewed papers.

Yuwei Wu received the BS degree in electronic science and technology from Sichuan University in 2012, and the MS degree in electronic and telecommunication engineering from University of Electronic Science and Technology of China. Currently, he works in Huawei Technologies Co., Ltd, China. His research interests include coverage evaluation and distributed algorithm design in wireless sensor network.