Homology-Based Distributed Coverage Hole Detection in Wireless Sensor Networks

Feng Yan, Anäis Vergne, Philippe Martins, Laurent Decreusefond

HAL Id: hal-00783403
https://hal.archives-ouvertes.fr/hal-00783403v3

Submitted on 8 Feb 2016

Homology-based Distributed Coverage Hole Detection in Wireless Sensor Networks

Feng Yan, *Member, IEEE,* Anaïs Vergne, Philippe Martins, *Senior Member, IEEE,* Laurent Decreusefond

Abstract—Homology theory provides new and powerful solutions to address the coverage problems in wireless sensor networks (WSNs). They are based on algebraic objects, such as Čech complex and Rips complex. Čech complex gives accurate information about coverage quality but requires a precise knowledge of the relative locations of nodes. This assumption is rather strong and hard to implement in practical deployments. Rips complex provides an approximation of Čech complex. It is easier to build and does not require any knowledge of nodes location. This simplicity is at the expense of accuracy. Rips complex can not always detect all coverage holes. It is then necessary to evaluate its accuracy. This work proposes to use the proportion of the area of undiscovered coverage holes as performance criteria. Investigations show that it depends on the ratio between communication and sensing radii of a sensor. Closed-form expressions for lower and upper bounds of the accuracy are also derived. For those coverage holes which can be discovered by Rips complex, a homology-based distributed algorithm is proposed to detect them. Simulation results are consistent with the proposed analytical lower bound, with a maximum difference of 0.5%. Upper bound performance depends on the ratio of communication and sensing radii. Simulations also show that the algorithm can localize about 99% coverage holes in about 99% cases.

Index Terms—Wireless sensor networks, coverage hole, homology.

I. INTRODUCTION

WIRELESS sensor networks (WSNs) have attracted a great deal of research attention due to their wide potential applications such as battlefield surveillance, environmental monitoring and intrusion detection. Many of these applications require a reliable detection of specified events. Such requirement can be guaranteed only if the target field monitored by a WSN contains no coverage holes, that is to say regions of the domain not monitored by any sensor. Coverage holes can be formed for many reasons, such as random deployment, energy depletion or destruction of sensors. Consequently, it is essential to detect and localize coverage holes in order to ensure the full operability of a WSN.

There is already an extensive literature about the coverage problems in WSNs. Several approaches are based on computational geometry with tools such as Voronoi diagram

F. Yan was with the Network and Computer Science Department, TELECOM ParisTech, Paris, France. He is currently with the Networks, Security and Multimedia Department, TELECOM Bretagne, Rennes, France (e-mail: feng.yan@telecom-bretagne.eu).

A. Vergne is with the Geometrica team, Inria Saclay - Ile de France, Palaiseau, France (e-mail: anais.vergne@inria.fr)

P. Martins and L. Decreusefond are with the Network and Computer Science Department, TELECOM ParisTech, Paris, France (e-mail: martins@telecom-paristech.fr, decreuse@telecom-paristech.fr.)

and Delaunay triangulations, to discover coverage holes [1]–[3]. These methods require precise information about sensor locations. This substantially limits their applicability since acquiring accurate location information is either expensive or impractical in many settings. Some other approaches attempt to discover coverage holes by using only relative distances between neighbouring sensors [4]–[6]. Similarly, obtaining precise range between neighbouring nodes is costly.

More recently, homology is utilized in [7]–[9] to address the coverage problems in WSNs. Ghrist and his collaborators introduced a combinatorial object, Čech complex, which fully characterizes coverage properties of a WSN (existence and locations of holes). Unfortunately, this object is very difficult to construct even if the precise location information of sensors is provided. Thus, they introduced a more easily computable complex, Rips complex. This complex is constructed with the sole knowledge of the connectivity graph of the network and gives an approximate coverage by simple algebraic calculations. As regards implementation in real WSN, these homology based methods are necessarily centralized, which makes them impractical in large scale sensor networks. Some algorithms have been proposed to implement the above mentioned ideas in a distributed context, see [10], [11]. But there are two disadvantages of these algorithms. On one hand, these homology based algorithms are all dependent on the assumption that the communication radius of a sensor is smaller than $\sqrt{3}$ times the sensing radius of the sensor. When such assumption is not satisfied, it is possible that Rips complex may miss some special coverage holes (defined as triangular holes in Section III). In order to assess the accuracy of Rips complex based coverage hole detection, it is thus of paramount importance to determine the proportion of the area of missed coverage holes. To the best of our knowledge, we are the first to investigate this problem. On the other hand, these algorithms try to verify coverage or detect coverage holes by homology computation, which is generally of high complexity especially for large scale networks. So it is necessary to design an efficient algorithm to detect coverage holes.

The main contributions of our paper are as follows. First, we present the relationships between Čech complex and Rips complex in terms of coverage hole under different ratios between communication and sensing radii of a sensor. We find that when the communication radius is at least two times sensing radius, if there is a hole in Rips complex, there must be a hole in Čech complex. A hole in a Čech complex missed by a Rips complex must be bounded by a triangle. Based on that, a formal definition of triangular and non-triangular hole is presented.

Second, for triangular holes, we derive the closed-form expressions for lower and upper bounds of the proportion of their area under a homogeneous setting. Such proportion is related to the ratio between communication and sensing radii of each sensor and three different ratios between communication and sensing radii are investigated.

Third, for non-triangular holes, an efficient homology based distributed algorithm is proposed to detect them. In the algorithm, the Rips complex is first constructed for a given WSN. Then some vertices and edges are deleted without changing the number of holes in the original Rips complex. After that, the edges lying on the boundary of holes will be detected. Then coarse boundary cycles can be discovered. Finally all boundaries of the non-triangular holes are found by minimizing the length of coarse boundary cycles.

The remainder of the paper is organized as follows. Section II presents the related work. In Section III, the network model and the formal definition of triangular and non-triangular hole are given. Upper and lower bounds on the proportion of the area of triangular holes under different ratios between communication and sensing radii are computed in Section IV. Section V describes the homology based distributed algorithm for non-triangular holes detection. In Section VI, performance evaluation of the bounds and the algorithm is given. Finally, Section VII concludes the paper.

II. RELATED WORK

Since this paper aims to evaluate the ratio of the area of coverage holes missed by homology based approaches and to design coverage hole detection algorithms, we present the related work in two aspects: analytical coverage ratio evaluation and coverage hole detection approaches.

A. Analytical coverage ratio evaluation

Extensive research has been done to analyze coverage ratio of a WSN. In [12], the authors studied the coverage properties of large-scale sensor networks and obtained the fraction of the area covered by sensors. The sensors are assumed to have the same sensing range and are distributed according to a homogeneous Poisson point process (PPP) in plane. In [13], the authors studied how the probability of k-coverage changes with the sensing radius or the number of sensors, given that sensors are deployed as either a PPP or a uniform point process. In [14], the coverage problem in planar heterogeneous sensor networks are investigated and analytical expressions of coverage are derived. Their formulation is more general in the sense that sensor can be deployed according to an arbitrary stochastic distribution, or can have different sensing capabilities or can have arbitrary sensing shapes. In [15], a point in a plane is defined to be tri-covered if it lies inside a triangle formed by three nodes, and the probability of tri-coverage is analyzed. None of them considered triangular holes, we provided some initial results about the ratio of the area of triangular holes in [16] and further improve them in this paper.

B. Coverage hole detection approaches

Coverage hole detection approaches can be generally classified into three categories: location-based, range-based and connectivity-based.

Location-based approaches are usually based on computational geometry with tools such as Voronoi diagram and Delaunay triangulations, to discover coverage holes [1]–[3]. Range-based approaches attempt to discover coverage holes by using only relative distances between neighbouring sensors [5], [6]. These two types of approaches need either precise location information or accurate distance information, which restricts their applications since such information is not easy to obtain in many settings.

In connectivity-based approaches, homology-based schemes attract particular attention due to its powerfulness for coverage hole detection. As a pioneer work, in [9], Ghrist and his collaborators introduced homology to detect coverage holes. They first introduced a combinatorial object, Čech complex, which can capture all coverage holes. Unfortunately, this object is very difficult to construct even if the precise location information of sensors is provided. Thus, they introduced another more easily computable complex, Rips complex. This complex can be constructed with the sole knowledge of the connectivity graph of the network and gives an approximate coverage by simple algebraic calculations. Then their work is followed by [7], [8], [17], [18], where a relative homological criterion for coverage is presented. But these homology based approaches are centralized. The first steps of implementing the above ideas in a distributed way were taken in [10]. It is shown that combinatorial Laplacians are the right tools for distributed computation of homology groups, and thus can be used for decentralized coverage verification. The combinatorial Laplacians can be used to detect absence of holes or a single hole. But when there are multiple holes close to each other in WSNs, it is not clear how to distinguish them. To address such limitations, a gossip like decentralized algorithm was proposed in [19] to compute homology groups, but its convergence is slow as stated in [19]. In [11], the authors first presented a decentralized scheme based on combinatorial Laplacians to verify whether there is a coverage hole or not in a WSN. For the case when there are coverage holes, they further formulated the problem of localizing coverage holes as an optimization problem for computing a sparse generator of the first homology group of the Rips complex corresponding to the sensor network. But it is possible that some cycle found by their algorithm contains multiple holes next to each other. More recently, for the purpose of coverage verification, a novel distributed algorithm for homology computation was proposed in [20] based on reduction and co-reduction of simplicial complex. For the case when there are more than one holes, the authors proposed to first find the homology generator of the reduced complex and then use their algorithm to localize holes. The algorithm is quite original but with high complexity since it requires to construct all the simplices. All these homology based algorithms try to verify coverage or detect coverage holes by computing homology either in a centralized or distributed way, but computing homology is

generally of high complexity especially for large scale sensor networks. Our algorithm does not try to compute homology to localize holes, it simplifies the Rips complex of a WSN by deleting some vertices and edges without changing homology and makes the Rips complex nearly planar. Then it can be efficient to detect coverage holes.

III. MODELS AND DEFINITIONS

Consider a collection of stationary sensors (also called nodes) deployed randomly in a planar target field. As usual, isotropic radio propagation is assumed. Each sensor monitors a region within a circle of radius R_s and may communicate with other sensors within a circle of radius R_c. Note that this assumption is mainly for analyzing the proportion of the area of triangular holes, it is not necessary for the algorithm proposed in Section V.

In addition, some other assumptions are as follows.

1) There are sensors located on the external boundary of the target field. They are known as fence sensors and other sensors are referred to as internal sensors. Each fence sensor has two fence neighbours. This is also the general assumption in many homology based algorithms [7]–[9], [11].
2) Although sensors are not aware of their locations, every sensor can know whether it is a fence or an internal node by using the mechanisms presented in [5] or other methods as in [21]. In fact, it is a conventional assumption adopted by many existing range-based methods [5], [22] or connectivity methods [11], [21].
3) Internal sensors are distributed in the planar target field according to a homogeneous PPP with intensity λ.
4) Each sensor has a unique ID.
5) The network has only one connected component.

Before defining Čech complex and Rips complex, it is necessary to give a brief introduction to the tools used in the paper. For further readings, see [23]–[25]. Given a set of points V, a k-simplex is an unordered set $[v_0, v_1, ..., v_k] \subseteq V$ where $v_i \neq v_j$ for all $i \neq j$. So a 0-simplex is a vertex, a 1-simplex is an edge and a 2-simplex is a triangle with its interior included, see Fig. 1. The faces of this k-simplex consist of all $(k\text{-}1)$-simplex of the form $[v_0, ..., v_{i-1}, v_{i+1}, ..., v_k]$ for $0 \leq i \leq k$. An abstract simplicial complex is a collection of simplices which is closed with respect to inclusion of faces.
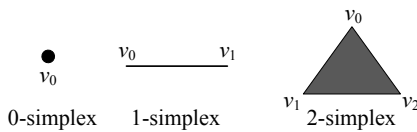

Fig. 1. 0-, 1- and 2-simplex

Let $\mathcal{V}$ denote the set of sensor locations in a WSN and $\mathcal{S} = \{s_v, v \in \mathcal{V}\}$ denote the collection of sensing ranges of these sensors. For a location v, $s_v = \{x \in \mathbb{R}^2 : \|x - v\| \leq R_s\}$. Then Čech complex and Rips complex can be defined as follows [7], [8].

Definition 1 (Čech complex). *Given a finite collection of sensing ranges $\{s_v, v \in \mathcal{V}\}$, the Čech complex of the collection, $\check{C}(\mathcal{V})$, is the abstract simplicial complex whose k-simplices correspond to non-empty intersections of $k + 1$ distinct elements of $\{s_v, v \in \mathcal{V}\}$.*

Definition 2 (Rips complex). *Given a finite set of points $\mathcal{V}$ in $\mathbb{R}^n$ and a fixed radius ϵ, the Rips complex of $\mathcal{V}$, $\mathcal{R}_\epsilon(\mathcal{V})$, is the abstract simplicial complex whose k-simplices correspond to unordered $(k + 1)$-tuples of points in $\mathcal{V}$ which are pairwise within Euclidean distance ϵ of each other.*

According to the definition, the Čech complex and Rips complex of the WSN, respectively denoted by $\check{C}_{R_s}(\mathcal{V})$ and $\mathcal{R}_{R_c}(\mathcal{V})$, can be constructed as follows: a k-simplex $[v_0, v_1, \cdots, v_k]$ belongs to $\check{C}_{R_s}(\mathcal{V})$ whenever $\cap_{l=0}^k s_{v_l} \neq \emptyset$ and a k-simplex $[v_0, v_1, \cdots, v_k]$ belongs to $\mathcal{R}_{R_c}(\mathcal{V})$ whenever $\|v_l - v_m\| \leq R_c$ for all $0 \leq l < m \leq k$.

Fig. 2 shows a WSN, its Čech complex and two Rips complexes for two different values of R_c. Depending on the ratio R_c over R_s, the Rips complex and the Čech complex may be close or rather different. In this example, for $R_c = 2R_s$, the Rips complex sees the hole surrounded by $2, 3, 5, 6$ as in the Čech complex whereas it is missed in the Rips complex for $R_c = 2.5R_s$. At the same time, the true coverage hole surrounded by $1, 2, 6$ is missed in both Rips complexes.
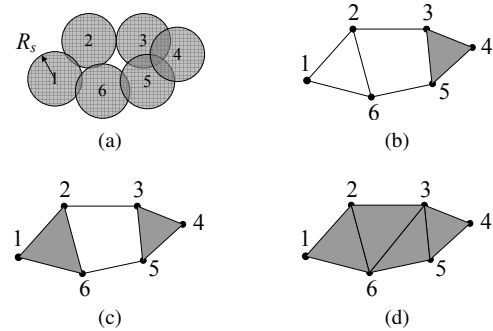

Fig. 2. (a) a WSN, (b) Čech complex, (c) Rips Complex under $R_c = 2R_s$, (d) Rips Complex under $R_c = 2.5R_s$

In fact, as proved in [7], any coverage hole can be found in Čech complex. Unfortunately, the construction of Čech complex is of very high complexity even if the precise location information of nodes is provided. So a more easily computable tool, Rips complex, is used. But Rips complex can not always capture all coverage holes. To be more specific, there exist following relations between Čech complex and Rips complex:

$$\mathcal{R}_{R_c}(\mathcal{V}) \subset \check{C}_{R_s}(\mathcal{V}) \subset \mathcal{R}_{2R_s}(\mathcal{V}), \text{ if } R_c \leq \sqrt{3}R_s. \quad (1)$$

According to (1), some relationships between Čech complex and Rips complex in terms of coverage hole can be derived as illustrated in the following corollaries. For convenience, define $\gamma = R_c/R_s$.

Corollary 1. *When $\gamma \leq \sqrt{3}$, if there is no hole in Rips complex $\mathcal{R}_{R_c}(\mathcal{V})$, there must be no hole in Čech complex $\check{C}_{R_s}(\mathcal{V})$.*

Corollary 2. *When $\gamma \geq 2$, if there is a hole in Rips complex $\mathcal{R}_{R_c}(\mathcal{V})$, there must be a hole in Čech complex $\check{C}_{R_s}(\mathcal{V})$.*

Corollary 3. *When $\sqrt{3} < \gamma < 2$, there is no guarantee relation between Rips complex $\mathcal{R}_{R_c}(\mathcal{V})$ and Čech complex $\check{C}_{R_s}(\mathcal{V})$ in terms of holes.*

Please refer to [26] for the proof.

From the discussion above, a hole in a Čech complex not seen in a Rips complex must be bounded by a triangle. Based on this observation, a formal definition of 'triangular hole' and 'non-triangular hole' is given as follows.

Definition 3 (Triangular and non-triangular hole)**.** *For a pair of complexes $\check{C}_{R_s}(\mathcal{V})$ and $\mathcal{R}_{R_c}(\mathcal{V})$, a triangular hole is an uncovered region bounded by a triangle formed by three nodes v_0, v_1, v_2, where v_0, v_1, v_2 can form a 2-simplex which appears in $\mathcal{R}_{R_c}(\mathcal{V})$ but not in $\check{C}_{R_s}(\mathcal{V})$. Other holes are non-triangular.*

For triangular holes, it is impossible to detect them with only connectivity information, so we want to analyze the proportion of the area of such holes in a target field. For non-triangular holes, we aim to design a distributed algorithm to discover the boundaries of these holes.

IV. BOUNDS ON PROPORTION OF THE AREA OF TRIANGULAR HOLES

For triangular holes, we aim to derive the proportion of their area. In this section, the conditions under which any point in the target field is inside a triangular hole are first given. In Section III, it is found that the proportion of the area of triangular holes is related to the ratio γ. Three different cases are considered for the proportion computation. For each case, the upper and lower bounds of the proportion are derived.

A. Preliminary

Lemma 1. *For any point in the target field, it is inside a triangular hole if and only if the following two conditions are satisfied:*

1) *the distance between the point and its closest node is larger than R_s.*
2) *the point is inside a triangle: the convex hull of three nodes with pairwise distance less than or equal to R_c.*

Fig. 3 gives an example to show a triangular hole. The blanket region inside the triangle is a triangular hole since it is not covered by any node and is bounded by a triangle.
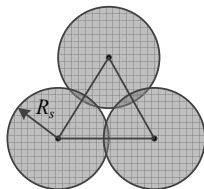

Fig. 3. An example of a triangular hole.

Lemma 2. *If there exists a point O which is inside a triangular hole, then $R_s < R_c/\sqrt{3}$.*

Lemma 3. *Let O be a point inside a triangular hole and l denote the distance between O and its closest neighbour, then $R_s < l \leq R_c/\sqrt{3}$.*

Please refer to [16] for the proof.

A PPP whose intensity is proportional to the Lebesgue measure is stationary in the sense that any translation of its atoms by a fixed vector does not change its law. Thus without considering border effect [27], any point has the same probability to be inside a triangular hole as the origin O. This probability in a homogeneous setting is also equal to the proportion of the area of triangular holes. We borrow part of the line of proof from [15] where a similar problem is analyzed.

We consider the probability that the origin O is inside a triangular hole. Since the length of each edge in the Rips complex must be at most R_c, only the nodes within R_c from the origin can contribute to the triangle which bounds a triangular hole containing the origin. Therefore, we only need to consider the PPP constrained in the closed ball $B(O, R_c)$ which is also a homogeneous PPP with intensity λ. We denote this process as Φ. In addition, $T(x, y, z)$ denotes the property that the origin O is inside the triangular hole bounded by the triangle with points x, y, z as vertices. When n_0, n_1, n_2 are points of the process Φ, $T(n_0, n_1, n_2)$ is also used to denote the event that the triangle formed by the nodes n_0, n_1, n_2 bounds a triangular hole containing the origin. In addition, we use $T'(n_0, n_1, n_2)$ to denote the event that the nodes n_0, n_1, n_2 can not form a triangle which bounds a triangular hole containing the origin.

Let $\tau_0 = \tau_0(\Phi)$ be the node in the process Φ which is closest to the origin. There are two cases for the origin to be inside a triangular hole. The first case is that the node τ_0 can contribute to a triangle which bounds a triangular hole containing the origin. The second case is that the node τ_0 can not contribute to any triangle which bounds a triangular hole containing the origin but other three nodes can form a triangle which bounds a triangular hole containing the origin, as illustrated in Fig. 4. So the probability that the origin is inside a triangular hole can be defined as

$$p(\lambda, \gamma) = P\{O \text{ is inside a triangular hole}\}$$
$$= P\{\bigcup_{\{n_0, n_1, n_2\} \subseteq \Phi} T(n_0, n_1, n_2)\} \qquad (2)$$
$$= P\{\bigcup_{\{n_1, n_2\} \subseteq \Phi \setminus \{\tau_0(\Phi)\}} T(\tau_0, n_1, n_2)\} + p_{sec}(\lambda, \gamma)$$

where

$$p_{sec}(\lambda, \gamma) = P\{\bigcup_{\substack{\{n_0, \cdots, n_4\} \\ \subseteq \Phi \setminus \{\tau_0(\Phi)\}}} T(n_0, n_1, n_2) \mid T'(\tau_0, n_3, n_4)\}$$

denotes the probability of the second case. $p_{sec}(\lambda, \gamma)$ is generally very small and obtained by simulations.

In the following parts, we will analyze this probability in three different cases.

B. Case $0 < \gamma \leq \sqrt{3}$

Theorem 1. *When $0 < \gamma \leq \sqrt{3}$, $p(\lambda, \gamma) = 0$.*

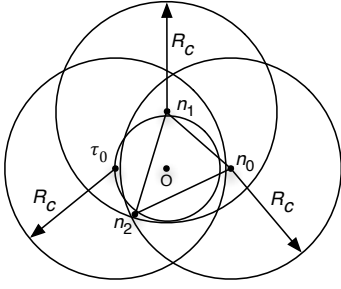*Proof:* It is a direct corollary from Lemma 2. ∎

Fig. 4. An example showing that the node closest to the origin τ_0 does not contribute to a triangle which bounds a triangular hole containing the origin because the distance to n_0 is larger than R_c and the triangle formed by τ_0, n_1, n_2 does not contain O. In contrast, n_0, n_1, n_2 can form a triangle which bounds a triangular hole containing O. Here we assume the distance between τ_0 and O is larger than R_s.

C. Case $\sqrt{3} < \gamma \le 2$

Theorem 2. *When $\sqrt{3} < \gamma \le 2$, $p_l(\lambda, \gamma) < p(\lambda, \gamma) < p_u(\lambda, \gamma)$, where*

$$p_l(\lambda, \gamma) = 2\pi\lambda^2 \int_{R_s}^{R_c/\sqrt{3}} r_0 dr_0 \int_{\alpha_0}^{\alpha_1} d\theta_1 \int_{r_0}^{R_1(r_0, \theta_1)} \tag{3}$$
$$e^{-\lambda\pi r_0^2} \times e^{-\lambda|S^+(r_0, \theta_1)|}(1 - e^{-\lambda|S^-(r_0, r_1, \theta_1)|})r_1 dr_1$$

and

$$p_u(\lambda, \gamma) = 2\pi\lambda^2 \int_{R_s}^{R_c/\sqrt{3}} r_0 dr_0 \int_{\alpha_0}^{\alpha_1} d\theta_1 \int_{r_0}^{R_1(r_0, \theta_1)} \tag{4}$$
$$e^{-\lambda\pi r_0^2} \times e^{-\lambda|S^+(r_0, \theta_1)|}(1 - e^{-\lambda|S^-(r_0, r_0, \theta_1)|})r_1 dr_1$$
$$+ p_{sec}(\lambda, \gamma)$$

and

$$\alpha_0 = 2\arccos(R_c/(2r_0))$$
$$\alpha_1 = 2\arcsin(R_c/(2r_0)) - 2\arccos(R_c/(2r_0))$$
$$R_1(r_0, \theta_1) = \min(\sqrt{R_c^2 - r_0^2 \sin^2\theta_1} - r_0\cos\theta_1,$$
$$\sqrt{R_c^2 - r_0^2 \sin^2(\theta_1 + \alpha_0)} + r_0\cos(\theta_1 + \alpha_0))$$
$$|S^+(r_0, \theta_1)| = \int_{\alpha_0}^{\theta_1} \int_{r_0}^{R_1(r_0, \theta)} r dr d\theta$$
$$|S^-(r_0, r_1, \theta_1)| = \int_{\theta_{2l}}^{-\alpha_0} \int_{r_0}^{R_2(r_0, r_1, \theta_1, \theta_2)} r dr d\theta_2$$
$$\theta_{2l} = \theta_1 - \arccos\frac{\cos(R_c/R) - \cos\theta_1\cos\theta_0}{\sin\theta_1\sin\theta_0}$$
$$R_2(r_0, r_1, \theta_1, \theta_2) = \min(\sqrt{R_c^2 - r_0^2\sin^2\theta_2} - r_0\cos\theta_2,$$
$$\sqrt{R_c^2 - r_1^2\sin^2(\theta_2 - \theta_1)} + r_1\cos(\theta_2 - \theta_1))$$

$p_{sec}(\lambda, \gamma)$ *is obtained by simulations.*

Since the proof is tedious, we only give the idea and main steps here. See Appendix A for detailed computation.

For the lower bound, we only consider the first case that the closest node τ_0 can contribute to a triangle which bounds a triangular hole containing the origin. The main idea is to first fix the closest node τ_0, and then sequentially decide the

regions where the other two nodes may lie in, and finally do a triple integral.

Using polar coordinates, we assume the closest node τ_0 lies on (d_0, π). Once the node τ_0 is determined, the other two nodes must lie in the different half spaces: one in $H^+ = \mathbb{R}^+ \times (0, \pi)$ and the other in $H^- = \mathbb{R}^+ \times (-\pi, 0)$. Assume n_1 lies in H^+ and n_2 lies in H^-. Since the distance to τ_0 is at most R_c, n_1 and n_2 must also lie in the ball $B(\tau_0, R_c)$. Furthermore, the distance to the origin is at most R_c and larger than d_0, they should also lie in the region $B(O, R_c) \backslash B(O, d_0)$. Therefore, n_1 must lie in $H^+ \bigcap B(\tau_0, R_c) \bigcap B(O, R_c) \backslash B(O, d_0)$ and n_2 must lie in $H^- \bigcap B(\tau_0, R_c) \bigcap B(O, R_c) \backslash B(O, d_0)$. In addition, considering the distance between n_1 and n_2 should be at most R_c and the origin should be inside the triangle formed by τ_0, n_1 and n_2, n_1 must lie in the shadow region $A^+ = H^+ \bigcap B(\tau_0, R_c) \bigcap B(O, R_c) \backslash B(O, d_0) \bigcap B(M_2, R_c)$, shown in Fig. 5. M_2 is one intersection point between the circle $C(O, d_0)$ and the circle $C(\tau_0, R_c)$, such intersection point must exist in this case since $R_c = \gamma R_s \le 2R_s < 2d_0$.
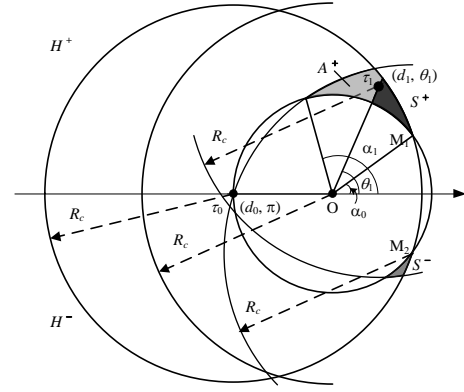

Fig. 5. Illustration of regions A^+, S^+ and S^- in the case $\sqrt{3} < \gamma \le 2$

Ordering the nodes in A^+ by increasing polar angle so that $\tau_1 = (d_1, \theta_1)$ has the smallest angle θ_1. And assume the nodes τ_0, τ_1 and another node $\tau_2 \in H^- \bigcap B(\tau_0, R_c) \bigcap B(0, R_c) \backslash B(0, d_0)$ can form a triangle which bounds a triangular hole containing the origin, then τ_2 must lie to the right of the line passing through τ_1 and O, denoted by $H^+(\theta_1)$ which contains all points with polar angle $\theta \in (\theta_1 - \pi, \theta_1)$. In addition, the distance to τ_1 is less than R_c. So the node τ_2 must lie in the region S^-, as illustrated in Fig. 5.

$$S^-(\tau_0, \tau_1) = S^-(d_0, d_1, \theta_1) = H^- \bigcap B(\tau_0, R_c)$$
$$\bigcap B(0, R_c) \backslash B(0, d_0) \bigcap H^+(\theta_1) \bigcap B(\tau_1, R_c)$$

Assume only τ_0, τ_1 and nodes in $S^-(\tau_0, \tau_1)$ can contribute to the triangle which bounds a triangular hole containing the origin, we can get a lower bound of the probability that the origin is inside a triangular hole. It is a lower bound because it is possible that τ_1 can not contribute to a triangle which bounds a triangular hole containing the origin, but some other nodes with higher polar angles in the region A^+ can contribute to such a triangle. E.g. in Fig. 6, if there is no node in S^- but there are some nodes in S'^-, then τ_1 can not contribute

to any triangle which bounds a triangular hole containing the origin, but τ_1' can form such a triangle with τ_0 and another node in S'^-.
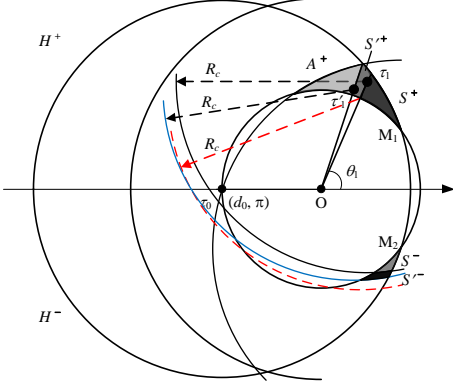

Fig. 6. Illustration of regions S'^+ and S'^- in the case $\sqrt{3} < \gamma \leq 2$

Next we prove the upper bound. As discussed in Section IV-A, there are two cases for the origin being inside a triangular hole. As for the second case that the closest node τ_0 can not but some other nodes can contribute to a triangle which bounds a triangular hole containing the origin, it is not easy to obtain a closed-form expression for such probability, we can get it by simulations. Simulation results show that this probability is less than 0.16% at any $\gamma \leq 3$ with any intensity λ. So we still focus on the probability of the first case.

Still consider the nodes in A^+, each node (d, θ) corresponds to an area $|S^-(d_0, d, \theta)|$. The higher is the area $|S^-(d_0, d, \theta)|$, the higher is the probability that there is at least one node in $S^-(d_0, d, \theta)$, consequently the probability of the first case will be higher. It can be seen from Fig. 5 that the closer to θ_1 is θ and the closer to d_0 is d_1, the higher is the area $|S^-(d_0, d, \theta)|$, so the largest area $|S^-(d_0, d, \theta)|$ is $|S^-(d_0, d_0, \theta_1)|$. Based on that, the upper bound can be derived.

As can be seen, the expression for lower bound is closed-form, while the expression for upper bound is not exactly closed-form since it includes a non-analytical part $p_{sec}(\lambda, \gamma)$. For lower bound and the closed-form part for upper bound, we use numerical integration to approximate the triple integrals. For $p_{sec}(\lambda, \gamma)$, we get it by simulations since it is generally very small, it has little impact on the derived upper bound.

D. Case $\gamma > 2$

Theorem 3. *When $\gamma > 2$, $p_l(\lambda, \gamma) < p(\lambda, \gamma) < p_u(\lambda, \gamma)$, where*

$$
\begin{aligned}
p_l(\lambda, \gamma) = 2\pi\lambda^2 \Big\{ &\int_{R_s}^{R_c/2} r_0 dr_0 \int_0^\pi d\theta_1 \int_{r_0}^{R_1'(r_0,\theta_1)} \\
&e^{-\lambda\pi r_0^2} \times e^{-\lambda|S^+(r_0,\theta_1)|}(1 - e^{-\lambda|S^-(r_0,r_1,\theta_1)|})r_1 dr_1 \\
+ &\int_{R_c/2}^{R_c/\sqrt{3}} r_0 dr_0 \int_{\alpha_0}^{\alpha_1} d\theta_1 \int_{r_0}^{R_1(r_0,\theta_1)} e^{-\lambda\pi r_0^2} \\
\times &e^{-\lambda|S^+(r_0,\theta_1)|}(1 - e^{-\lambda|S^-(r_0,r_1,\theta_1)|})r_1 dr_1 \Big\}
\end{aligned}
\tag{5}
$$

and

$$
\begin{aligned}
p_u(\lambda, \gamma) = 2\pi\lambda^2 \Big\{ &\int_{R_s}^{R_c/2} r_0 dr_0 \int_0^\pi d\theta_1 \int_{r_0}^{R_1'(r_0,\theta_1)} \\
&e^{-\lambda\pi r_0^2} \times e^{-\lambda|S^+(r_0,\theta_1)|}(1 - e^{-\lambda|S^-(r_0,r_0,\theta_1)|})r_1 dr_1 \\
+ &\int_{R_c/2}^{R_c/\sqrt{3}} r_0 dr_0 \int_{\alpha_0}^{\alpha_1} d\theta_1 \int_{r_0}^{R_1(r_0,\theta_1)} e^{-\lambda\pi r_0^2} \\
\times &e^{-\lambda|S^+(r_0,\theta_1)|}(1 - e^{-\lambda|S^-(r_0,r_0,\theta_1)|})r_1 dr_1 \Big\} \\
+ &p_{sec}(\lambda, \gamma)
\end{aligned}
\tag{6}
$$

and

$$
\begin{aligned}
R_1'(r_0, \theta_1) = \min(&\sqrt{R_c^2 - r_0^2 \sin^2\theta_1} - r_0 \cos\theta_1, \\
&\sqrt{R_c^2 - r_0^2 \sin^2\theta_1} + r_0 \cos\theta_1)
\end{aligned}
$$

$p_{sec}(\lambda, \gamma)$ *is obtained by simulations.*
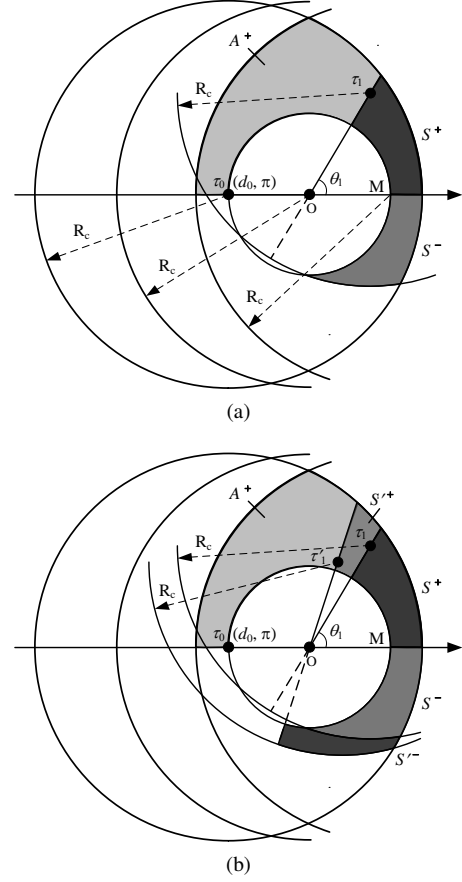

(a)

(b)

Fig. 7. Illustrations of regions in case $\gamma > 2$. (a) the regions A^+, S^+ and S^- (b) the regions S'^+ and S'^-

In this case, we can use the same method as in Section IV-C to get the lower and upper bounds, shown in (5) and (6) respectively. But we need to consider two situations $R_s < d_0 \leq R_c/2$ and $R_c/2 < d_0 \leq R_c/\sqrt{3}$. In the first situation, $d_0 \leq R_c/2$ means that the ball $B(O, d_0)$ is included in the ball $B(\tau_0, R_c)$. The illustrations for the regions A^+, S^+, S^-, S'^+ and S'^- are shown in Fig. 7. In addition, the lower limit of integration for θ_1 is 0 and the upper limit is π. The second situation is the same as that in Section IV-C.

V. DISTRIBUTED COVERAGE HOLE DETECTION ALGORITHM

For non-triangular holes, we aim to design an efficient distributed algorithm to detect their minimum boundary cycles. The basic idea is that for the Rips complex of a WSN, we try to delete some vertices and edges without changing the homology while making the Rips complex more sparse and nearly planar. Then it is easy to find boundary edges (1-simplices), each of which has at most one neighbour. Finally such edges are connected in some order to form boundary cycles.

More specifically, our algorithm includes five components: weight computation, vertex and edge deletion, boundary edge detection, coarse boundary cycles discovery and boundary cycles minimization, as shown in Fig. 8. An example is used to illustrate the procedures of this algorithm in Fig. 9. In weight computation component, the Rips complex of the WSN is first constructed, shown in Fig. 9(a), then each node computes its weight independently. The definition of weight of a node will be presented in the next part. After obtaining the weight, each node continues to determine whether it is deletable or not according to some rule defined hereafter. Fig. 9(b) shows the result of vertex deletion. Furthermore, some special edges may be deleted. Fig. 9(c) shows the process of such special edge deletion. After the second component, many boundary edges can be found, as the bold lines shown in Fig. 9(d). But it is possible that some other boundary edges have not been found. Then in the third component, all or nearly all boundary edges will be found after deleting some edges, see Fig. 9(e~j). Then coarse boundary cycles can be easily discovered, as shown in Fig. 9(k). The found boundary cycles may not be minimum. In this case, coarse boundary cycles will be minimized in the final component as shown in Fig. 9(l).
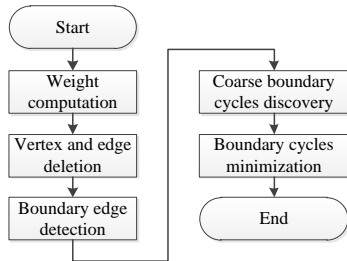

Fig. 8. Flow chart of the algorithm

A. Definitions

We say that a i-simplex $[v_{i0}, v_{i1}, ..., v_{ii}]$ is part of a j-simplex $[v_{j0}, v_{j1}, ..., v_{jj}]$ if $[v_{i0}, v_{i1}, ..., v_{ii}] \subset [v_{j0}, v_{j1}, ..., v_{jj}]$. So the vertex $[v_0]$ and $[v_1]$ is part of the edge $[v_0, v_1]$. The edge $[v_0, v_1]$ is part of the triangle $[v_0, v_1, v_2]$. In addition, we use $E(v)$ to denote all the edges that the node v is part of and $T(v)$ to denote all the 2-simplices that the node v is part of.

Definition 4 (Index of a 2-simplex). *The index of a 2-simplex $\triangle$ is the highest dimension of the simplex that the 2-simplex is part of, denoted by $I_\triangle$.*

Definition 5 (Weight of a node). *The weight of a fence node is defined to be 0. For any internal node v, if there exists one edge in $E(v)$ which is not part of any 2-simplex, the weight w_v of node v is set to be 0; if not, the weight is the minimum index of all the 2-simplices in $T(v)$, that is $w_v = \min_{\triangle \in T(v)} I_\triangle$*

The weight of an internal node is an indicator of the density of surrounding nodes. If the weight of an internal node is 0, the node must be on the boundary of a coverage hole. The larger the weight is, the higher is the probability that the node is not on the boundary of a coverage hole.

We also use the definition of simple-connectedness graph as in [28]. Let G be a simple graph with vertex set $V(G)$ and edge set $E(G)$. A cycle C is a sub-graph of G if it is connected and each vertex in C has degree two. The length of a cycle C is the number of its edges, denoted by $|E(C)|$. The cycle space $C(G)$ of a graph G contains all the cycles in G. The addition of two cycles C_1 and C_2 is defined as $C_1 \oplus C_2 = (E(C_1) \cup E(C_2)) \setminus (E(C_1) \cap E(C_2))$. The triangle cycle subspace $C_T(G)$ of G is the set of all 3-length cycles in $C(G)$.

Definition 6 (Simple-Connectedness Graph). *A connected graph G is of simple connectedness if its cycle space $C(G)$ is empty, or for any cycle C in $C(G)$, there exists a set of 3-length cycles $T_0 \subseteq C_T(G)$ such that $C = \sum_{T \in T_0} T$, which means all cycles in $C(G)$ can be triangulated.*

Let X be a vertex (or edge) set in a graph G, we use $G[X]$ to denote the vertex-induced (or edge-induced) sub-graph by X. The neighbour set of a vertex v in G is denoted by $N_G(v)$. The neighbouring graph $\Gamma_G(v)$ of vertex v is denoted as $G[N_G(v)]$. The neighbouring graph $\Gamma_G(e)$ of an edge $e = [u, v]$ is defined as $G[N_G(u) \cap N_G(v) \cup \{u, v\}] - e$. The neighbour set of k-simplex $[v_0, v_1, ..., v_k]$ is defined as $\bigcap_{i=0}^{k} N_G(v_i)$.

Definition 7 (Deletion of a k-simplex in Rips complex $\mathcal{R}(\mathcal{V})$). *A k-simplex $[v_0, v_1, \cdots, v_k]$ is deleted in a Rips complex $\mathcal{R}(\mathcal{V})$ means that the simplex and all simplices which the simplex is part of are deleted from $\mathcal{R}(\mathcal{V})$.*

Based on definitions above, we can give the definition of HP (Homology Preserving) transformation.

Definition 8 (HP Transformation). *A HP transformation is a sequential combination of vertex (or edge) deletion as follows: a vertex (or edge) x of G is deletable if neighbouring graph $\Gamma_G(x)$ (1) has two or more vertices; (2) is connected and (3) is a simple-connectedness graph.*

Theorem 4. *HP transformations do not change the number of coverage holes in Rips complex of a WSN.*

Proof: In order to prove HP transformations do not change the number of coverage holes in Rips complex of a WSN, we only need to prove that in the process of any HP transformation, there is no new coverage holes created and no two coverage holes merged. If a new coverage hole is created when a vertex v (or edge e) is deleted, then the boundary cycle of the new coverage hole must be a cycle in $\Gamma_G(v)$ (or $\Gamma_G(e)$), which means $\Gamma_G(v)$ (or $\Gamma_G(e)$) is not a simple-connectedness graph. It is contrary to the third condition in
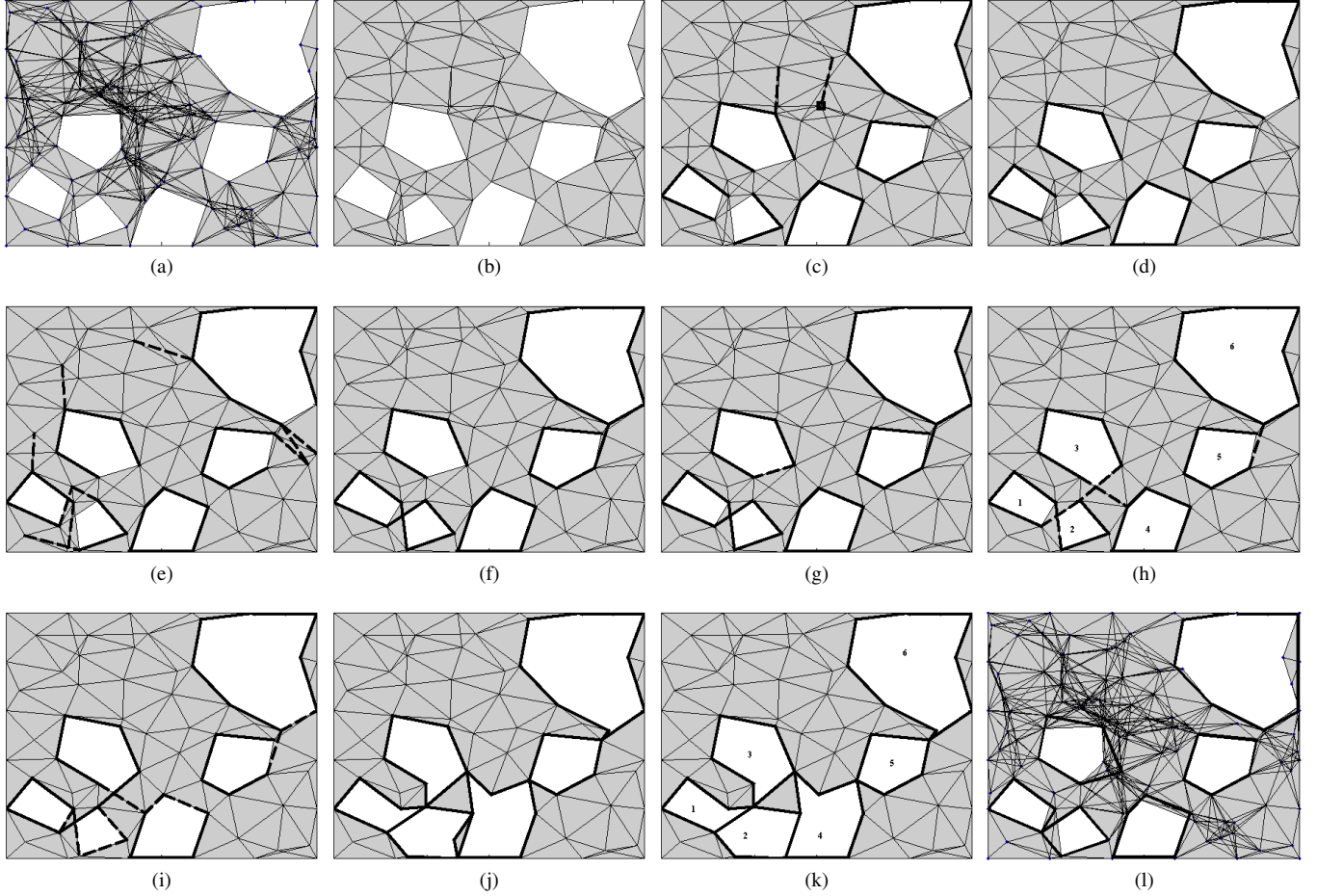
Fig. 9. Procedures of the boundary detection algorithm. (a) Rips complex of a WSN, (b) after vertex deletion, (c~d) edge deletion, (e~j) boundary edge detection, (k) coarse boundary cycles discovery, (l) boundary cycles minimization

HP transformation, so there is no new coverage hole created. Furthermore, if two coverage holes are merged when a vertex v (or edge e) is deleted, then the neighbour graph $\Gamma_G(v)$ ($\Gamma_G(e)$) must not be connected, which is contrary to the second condition in HP transformation. So no two coverage holes are merged. The situations where two holes would merge after a vertex v deletion are shown in Fig. 10(a) and (b). Fig. 10(b) also shows the situation for edge e deletion. Consequently, the number of coverage holes will not be changed in the process of any HP transformation. ∎
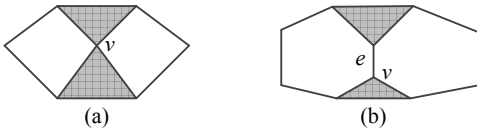

Fig. 10. Illustration of merge of two holes when deleting a vertex or edge

B. Weight computation

In this component, each node first constructs its simplices to form the Rips complex of the WSN and then computes its weight. For any fence node, its weight is 0. For any internal node, theoretically the node needs to construct all the simplices

which it is part of. As we consider WSNs in a planar target field, each internal node only needs to construct all its 1-simplices and 2-simplices and their neighbours. This can also reduce the computation complexity. In order to do this, the node needs to obtain all its 1- and 2-hop neighbours information. This can be achieved by two broadcasts of hello message. In the first one, each node broadcasts its id. When it gets all the IDs of its 1-hop neighbours, it continues to broadcast a hello message containing the IDs of its 1-hop neighbours. After receiving the neighbour list of its neighbours, the node can obtain its $E(v)$, the set of edges and $T(v)$, the set of 2-simplices. It can also get the neighbours of each simplex. For any $e \in E(v)$, let $n(e)$ denote the neighbour set of e. For any $t \in T(v)$, let $n(t)$ denote the neighbour set of t. Then the weight of node v can be computed as in Algorithm 1.

C. Vertex and edge deletion

In this component, we conduct maximal vertex deletion without changing the number of coverage holes in the original WSN and also delete some special edges if such edges exist. For vertex deletion, we only consider internal nodes, fence nodes will never be deleted.

1) Vertex deletion: As explained in Section V-A, the larger the weight is, the higher is the probability that the node does

---

Algorithm 1 Weight computation (for internal node v)

Begin

  if $\exists e \in E(v), n(e)$ is empty **then**

    $w_v = 0$

  else if $\exists t \in T(v), n(t)$ is empty **then**

    $w_v = 2$

  else

    $w_v = 3$

  end if

END

---

not lie on the boundary. Meanwhile, if the deletion of a vertex may create a new coverage hole, it must not be deleted no matter how high the weight is. So we have such a rule for vertex deletion. If the weight of a vertex is smaller than 3, it should never be deleted. Otherwise, the vertex continues to check whether it is deletable or not according to HP transformation. After the verification, the vertex broadcasts a message indicating that it can be deleted or not. After receiving the status of all its neighbours, each deletable vertex continues to check whether it should be deleted. The weight of any deletable vertex must be 3. We assume that the vertex with a lower ID has the priority to be deleted first. So each deletable vertex just needs to check whether its ID is the lowest among all its deletable neighbours. If so, it should be deleted. Otherwise, it should not be deleted. Algorithm 2 gives the detailed process for vertex deletion. According to the rule, two neighbouring vertices will not be deleted simultaneously, so each vertex can make the decision independently. When a vertex is deleted, it broadcasts a message to its neighbours. All its neighbours will modify their simplices accordingly and compute their weights again. The procedure of vertex deletion terminates until no vertex can be deleted in the Rips complex. Fig. 9(b) gives the final result after vertex deletion.

Algorithm 2 Vertex deletion (for internal node v)

Begin

  if $w_v < 3$ **then**

    node v can not be deleted

  else if node v is not deletable according to HP transformation **then**

    node v can not be deleted

  else if the ID of node v is the smallest among all its deletable neighbours **then**

    node v is deleted

  end if

END

---

2) Edge deletion: After vertex deletion, it seems natural to delete all edges which are deletable according to HP transformation. We do not run in this way. On one hand, this may not be useful since deleting edges which are far from coverage holes does not help the detection of boundary cycles of coverage holes. On the other hand, deleting all such edges will increase the complexity of the algorithm.

Fortunately, it has been proven in [29] that it is possible to make the Rips complex planar by deleting vertices and edges

if there is no hole in the Rips complex. For a planar Rips complex, the edge which has at most one neighbour must be on the boundary. This inspires us to check the edges which have at most one neighbour even if it may not be easy to make a Rips complex planar when the Rips complex has holes. It is interesting to find that most edges which have at most one neighbour lie on the boundary of a coverage hole, such as the solid bold edges shown in Fig. 9(c). But there exist also some special such edges not lying on the boundary, such as the dashed bold edges shown in Fig. 9(c). We try to delete such special edges.

We call edges having at most one neighbour as boundary edges, and call boundary edges not lying on the boundary of a coverage hole as special boundary edges. Considering that special boundary edges do not lie on the boundary of a coverage hole, deleting them will not produce new boundary edges. Then we design a rule for deleting special boundary edges. For a boundary edge $[u, v]$ which has only one neighbour w[1], if deleting $[u, v]$ will not make $[u, w]$ and $[v, w]$ be boundary edges, then the edge $[u, v]$ can be deleted. We can check that the rule is HP since the neighbouring graph of the edge $[u, v]$ is connected with three vertices u, v, w and without a cycle. It means that deletion of such edges does not change the number of coverage holes in Rips complex. Fig. 11 shows the result of deleting such edges, which is an enlarged version of Fig. 9(c). Some edges lying on the boundary may also be deleted according to the rule. This is not a big issue, because deletion of such edges just enlarges the current coverage holes. It can be solved in the boundary cycles minimization component.

After edge deletion, some vertices may be deleted again, such as the vertex denoted by a bold square in Fig. 9(c). If such a case happens, we can continue to do vertex deletion until no more vertex or edge can be deleted. Fig. 9(d) shows the result after edge deletion.
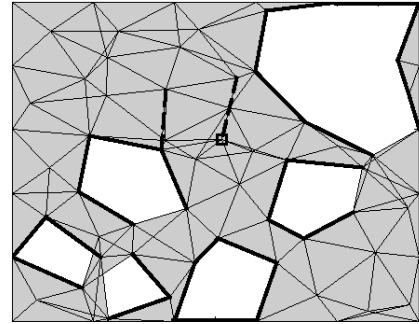

Fig. 11. Special boundary edge deletion

D. Boundary edge detection

As explained in last section, in a planar Rips complex, it is easy to detect all boundary edges by checking whether they have at most one neighbour or not. Then boundary edges are connected sequentially to form boundary cycles. It is thus important to detect boundary edges. In our case, after last step, we can find that some edges lying on the boundary have

---

[1]The boundary edge having no neighbour must be on the boundary.

not been found. This is due to that the Rips complex near coverage holes is not planar. We try to make it planar.

We consider the nodes having one or more boundary edges as boundary nodes and other nodes as non-boundary nodes. First, we need to delete some edges connecting non-boundary nodes and boundary nodes according to HP transformation, such as the dashed bold edges shown in Fig. 9(e). After that, some new boundary edges may be recognized as shown in Fig. 9(f). But it is possible that the new found boundary edges cross with non-boundary edges, as the dashed bold edge in coverage hole 5 in Fig. 9(h), or cross with each other, as the dashed bold edges in coverage holes 1 and 2 in Fig. 9(h). As for the case in coverage hole 5, we can design a similar rule as in last step to delete them. Considering such edges are near the boundary of a coverage hole, deletion of them can make new boundary edges, we design the rule as follows: for a boundary edge $[u, v]$, it has one neighbour w, $[u, w]$ and $[v, w]$ are not boundary edges, if the deletion of the edge $[u, v]$ can make at least one of the two edges $[u, w]$ and $[v, w]$ be boundary edge, then $[u, v]$ can be deleted. Such a rule is HP as explained in last component. As for the crossing boundary edges case, it is illustrated in the next paragraph.

Second, we need to delete some edges connecting boundary nodes according to HP transformation, such as the dashed bold edge in Fig. 9(g). Considering that crossing edges can only exist in 3- or higher dimension simplices, and higer dimension simplices can transfer to several 3-simplices by deleting some edges. We thus only consider 3-simplices, an example is shown in Fig. 12(a). Deleting one edge of a 3-simplex can produce three possible crossing boundary edges cases[2], as shown in the top part of Fig. 12 (b) $\sim$ (d), where bold lines denote boundary edges and other ones denote non-boundary edges. Then we can design corresponding rules to delete some boundary edges. The bottom parts of Fig. 12 (b) $\sim$ (d) give the results after deletion. The rules are also HP as explained in last component.
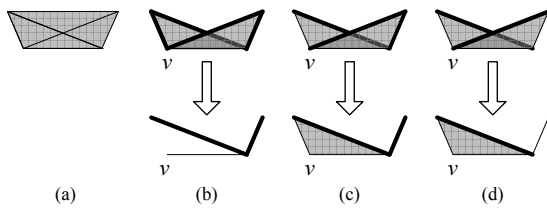

Fig. 12. Illustration of crossing boundary edges

According to such rules, some boundary edges can be deleted, such as the dashed bold edges in Fig. 9(i). From Fig. 9(i), it can be found that certain boundary edge is deleted incorrectly. It is not a big issue as explained in last component. After deletion of such edges, new boundary edges can be found as shown in Fig. 9(j).

In general, after the process above, all boundary edges can be found. But there exists one special case as in Fig. 13(a), where some boundary edges can not be detected. This is due to that for a vertex $v\prime$, as in Fig. 13(b), its neighbouring graph is not a simple-connected graph since the cycle formed by

[2]Here we do not consider the case without crossing boundary edges

v_1, v_2, v_3, v_4 can not be triangulated, but the cycle can be triangulated in the graph induced by $v_1, v_2, v_3, v_4, u\prime$. In this case, no vertex or edge is deletable according to HP transformation and no boundary edges can be found since each edge has two neighbours. Such case has no impact on boundary cycles detection, as illustrated in the next component.
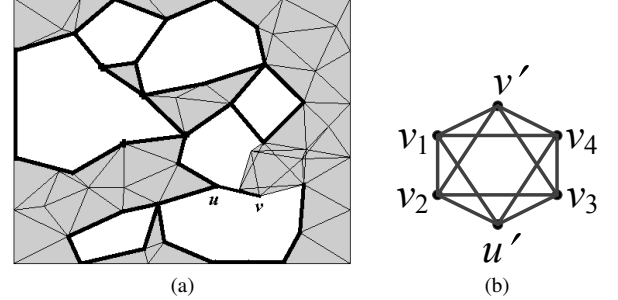

Fig. 13. Special case when some boundary edges can not be detected

E. Coarse boundary cycles discovery

After boundary edges are detected in the former component, it is easy to discover the coarse boundary cycles. We just need to randomly choose one node which has two boundary edges in any boundary cycle. The node initiates the process to find the coarse boundary cycle by sending a message along one of the boundary edges. Then the boundary neighbour continues sending the message along its boundary edges. When the initiating node receives the message coming back along the other boundary edge, it discovers one coarse boundary cycle. Similarly, all coarse boundary cycles can be found, as shown in Fig. 9(k). If only part of boundary edges are detected after former components, we can not transmit the message only along boundary edges. The message will be also transmitted along non-boundary edges and some non-boundary nodes will also need to broadcast the message, which increases the complexity of the algorithm.

As for the special case shown in Fig. 13(a), when the node v receives a message from its boundary neighbour node u, it broadcasts the message to all its neighbours except u. If its neighbour node is a boundary node, then the message can be sent along the boundary edges. If its neighbour node is not a boundary node but it has boundary neighbour nodes, then it can send the message to its boundary neighbour nodes. Else, it will not transmit the message again. In this way, the message goes along boundary edges most of the time and can return to the original node initiating the message.

F. Boundary cycles minimization

It is possible that some coarse boundary cycles found are not minimum, so we need to minimize such cycles. This can be achieved by checking whether there exists a shorter path between any two nodes in the cycle. Since each node has its 1- and 2-hop neighbours information, it can locally check the existence of a shorter path in the cycle in general cases. If there exists, we shorten the cycle and continue to do the same verification until no such case exists. After that, it is

still possible some cycle has not been minimized, such as the coverage hole 2 in Fig. 9(k). So we use the following 2-hop shrinking process to shorten the cycle. For any four adjacent nodes in the cycle, say a, b, c, d, if there exists one node x which is one common neighbour of nodes a, b, c, d, then the cycle can be shortened by using x to replace nodes b and c.

In this way, we can nearly obtain most minimum cycles surrounding coverage holes. In some cases, we can not get the minimum cycles since each node only has its 1- and 2-hop neighbours information. Even so, the boundary cycles discovered in the algorithm can still provide valuable information about coverage holes.

VI. SIMULATIONS AND PERFORMANCE EVALUATION

Performance evaluation of the theoretical bounds obtained in Section IV and the algorithm proposed in Section V is presented in this section.

A. Simulation settings

For bounds evaluation, a disk centered at the origin with radius R_c is considered in the simulations. The probability that the origin is inside a triangular hole is computed. Sensors are randomly distributed in the disk according to a Poisson point process with intensity λ. The sensing radius R_s of each node is set to be 10 meters and γ is chosen from 2 to 3 with interval of 0.2. So the communication radius R_c ranges from 20 to 30 meters with interval of 2 meters. λ is selected from 0.001 to 0.020 with interval of 0.001. For each γ, 10^7 simulations are run under each λ to check whether the origin belongs to a triangular hole.

For performance evaluation of the algorithm, we choose a 100×100 m^2 square area as the target field. The sensing radius R_s of each node is 10 meters. The communication radius R_c is set to be 20 meters and so $\gamma = 2$. There are fence sensors locating along the edges of the square with 20 meters distance between neighbours. Other internal sensors are randomly distributed in the area according to a Poisson point process with intensity λ.

B. Proportion of the area of triangular holes

The probability $p(\lambda, \gamma)$ obtained by simulations is presented with the lower and upper bounds in Fig. 14(a) and 14(b) respectively. The simulation results for $p_{sec}(\lambda, \gamma)$ are shown in Fig. 15, which indicate that $p_{sec}(\lambda, \gamma)$ is always smaller than 0.16% in the simulation settings.

It can be seen from Fig. 14 that for any value of γ, $p(\lambda, \gamma)$ has a maximum at a threshold value λ_c of the intensity. As a matter of fact, for $\lambda \leq \lambda_c$, the number of nodes is small. Consequently the probability that the origin belongs to a triangular hole is relatively small too. With the increase of λ, the connectivity between nodes becomes stronger. As a result, the probability that the origin belongs to a triangular hole increases. However, when the intensity reaches the threshold value, the origin is covered with maximum probability. $p(\lambda, \gamma)$ decreases for $\lambda \geq \lambda_c$. The simulations also show that λ_c decreases with the increase of γ.

On the other hand, it can be seen from Fig. 14(a) and 14(b) that for a fixed intensity λ, $p(\lambda, \gamma)$ increases with the increases of γ. That is because when R_s is fixed, the larger R_c is, the higher is the probability of each triangle containing a coverage hole.

Furthermore, the maximum probability increases quickly with γ ranging from 2.0 to 3.0. It is shown that when $\gamma = 2$, the maximum probability from simulation is about 0.03% and thus it is acceptable to use Rips complex based algorithms to discover coverage holes. While the ratio γ is high to a certain extent, it is unacceptable to use connectivity information only to discover coverage holes.

Finally, it can be found in Fig. 14(a) that the probability obtained by simulation is very well consistent with the lower bound. The maximum difference between them is about 0.5%. Fig. 14(b) shows that probability obtained by simulation is also consistent with the upper bound. The maximum difference between them is about 3%.

In addition, combined with the homology based algorithm, the analytical results can be used for planning of WSNs. For example, assume a WSN is used to monitor a planar target field and the ratio $\gamma = 2$, according to the analytical upper bounds, we can see that the maximum proportion of the area of triangular holes under $\gamma = 2$ is about 0.06%, which can be neglected. It means that as long as the non-triangular holes can be detected by the homology based algorithm and covered by additional nodes, we can say the target field is covered. But if $\gamma = 3$, it can be seen from the analytical upper bounds that the maximum proportion of the area of triangular holes is about 11%, which means that even if the non-triangular holes can be detected and covered, it is still possible that about 11% of the target field is uncovered. Under this situation, some approaches like increasing the intensity of nodes need to be used in order to make most of the target field covered.

C. Performance of the algorithm

1) Complexity: The computation complexity of each step in the algorithm is shown in Table I. In weight computation, each node needs to construct its 0-, 1- and 2-simplices. For construction of 0- and 1-simplices, each node only needs to know its 1-hop neighbour information, which can be done by a broadcast as explained in Section V-B. For 2-simplices construction, each node needs to obtain its neighbours' 1-hop neighbour information, which is achieved by another broadcast. Then the node continues to determine whether it can form a 2-simplex with any two of its neighbours by checking whether they have a common neighbour. The node needs to check $n(n-1)/2$ times, where n is the number of its 1-hop neighbours. Then the node computes its weight by checking all its 2-simplices of which the maximum number is $n(n-1)/2$. The computation complexity of this component is thus $O(n^2)$.

In vertex deletion part, each node needs to check whether it is deletable or not according to HP transformation. This can be done by checking all its cycles in its neighbouring graph. It can build a spanning tree in its neighbouring graph and check all fundamental cycles in the spanning tree. There are $E - n + 1$ fundamental cycles, where E is the number of edges in its
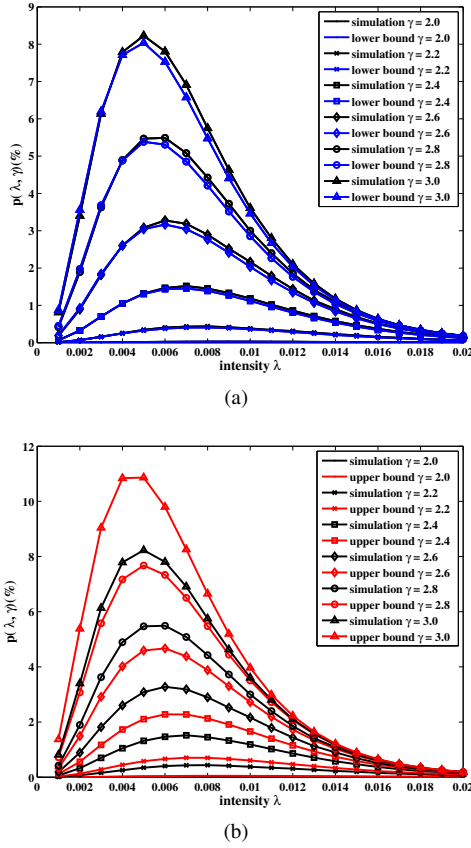
Fig. 14. Proportion of the area of triangular holes (a) simulation results and lower bounds ; (b) simulation results and upper bounds
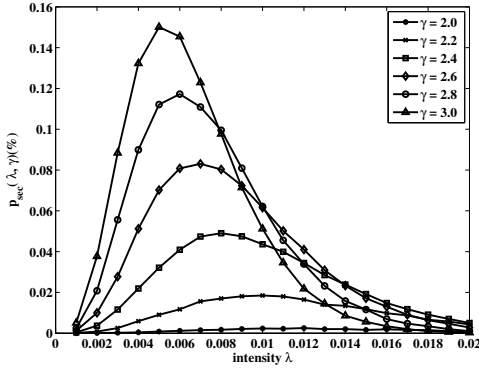

Fig. 15. Simulation results for $p_{sec}(\lambda, \gamma)$.

neighbouring graph, so the worst case computation complexity is $O(n^2)$. Since the node needs to recompute its weight and recheck whether it is deletable when any one of its neighbour is deleted, so the total worst case computation complexity is $O(n^3)$. As for edge deletion, the node only needs to check all its 1-simplices to see whether there exists the case illustrated in Section V-C, so the complexity is $O(n)$. The complexity of this component is thus $O(n^3)$.

In the boundary edge detection component, nodes need to check whether some of their edges can be deleted or not according to HP transformation. For each edge, the worst case computation complexity is $O(n^2)$ as explained in last paragraph, the total worst case computation complexity is

TABLE I
COMPLEXITY OF EACH STEP IN THE ALGORITHM

Step	Complexity
Weight computation	$O(n^2)$
Vertex and edge deletion	$O(n^3)$
Boundary edge detection	$O(n^3)$
Coarse boundary cycles discovery	$O(1)$
Boundary cycles minimization	$O(1)$

thus $O(n^3)$ since there are maximum n edges. The actual complexity is much less than that since for one edge, there are usually very few nodes in its neighbouring graph. In addition, the boundary nodes need to check whether there exist special cases as illustrated in Section V-D. The node needs to check all its 2-simplices, which is of complexity $O(n^2)$ since there are maximum $n(n-1)/2$ 2-simplices. So the complexity of this component is $O(n^3)$.

As for the final two components, each node only needs to broadcast some messages and do some local computations, the complexity is $O(1)$. So the total worst case computation complexity for our algorithm is $O(n^3)$.

2) Comparison with other algorithms: In order to evaluate the performance of our proposed homology based algorithm (denoted as HBA), we compare it with the location based algorithm (denoted as LBA) proposed in [30]. Since location based algorithm can discover both triangular and non-triangular coverage holes, but our algorithm can only detect non-triangular coverage holes, we do not consider those triangular coverage holes in the comparison. It is possible that there exist shorter paths in boundary cycles found by LBA, we first shrink them using 1-hop neighbour information of boundary nodes. After that, we compare those boundary cycles with what our algorithm finds. For some coverage holes, the minimum boundary cycles may not be unique, two boundary cycles are considered to surround the same coverage hole if one cycle can be converted to another one by using only 1-hop neighbours information. We emphasize that only 1-hop neighbours information can be used in the comparison in order to evaluate the accuracy of boundary cycles found by our algorithm. For example, if one cycle c_1 found by our algorithm can not be converted to another cycle c_2 found by LBA using only 1-hop neighbours information but can be converted by using 2-hop neighbours information, we consider the cycle c_1 is not accurate and the corresponding coverage hole is not found.

Based on the method presented above, we set λ to be 0.008, 0.010 and 0.012 to represent sparse, moderate and dense WSNs respectively. For each intensity, 1000 simulations are performed. Simulation results show that when λ is 0.008, there are nine times among the 1000 times when our algorithm can not find all non-triangular coverage holes. In each of the nine times, only one coverage hole is missed. There are 7363 non-triangular holes in total and 7354 ones found by our algorithm. When λ is 0.010 and 0.012, only one time among the 1000 times when our algorithm can not find all coverage holes. And in that time, only one coverage hole is missed. When λ is 0.010, there are 6114 non-triangular holes in total and 6113 ones found by our algorithm. When λ is 0.012, there are 4613

non-triangular holes in total, of which 4612 ones are found. The results are shown in Fig. 16. All these results show that our algorithm can find about 99% coverage holes in about 99% cases.
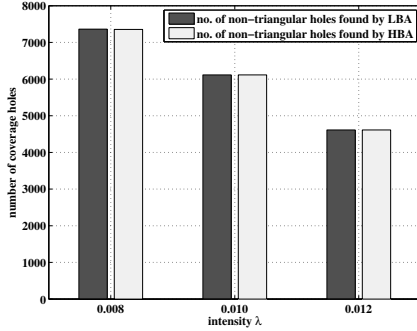

Fig. 16. Performance evaluation of the algorithm

VII. CONCLUSIONS

In this paper, we adopt two types of simplicial complex called Čech complex and Rips complex to capture coverage holes of a WSN. The relationship between Čech complex and Rips complex in terms of coverage hole is first analysed under different ratios between communication radius and sensing radius of a sensor. Based on that, we define two types of coverage holes: triangular and non-triangular hole. For triangular holes, both the lower and upper bounds on the proportion of the area of triangular holes in a WSN are derived. Such proportion is related to the ratio between communication radius and sensing radius of each sensor. When the ratio is no larger than $\sqrt{3}$, there is no triangular hole. When the ratio is between $\sqrt{3}$ and 2, both the theoretical analysis and simulation results show that the proportion is lower than 0.06% under any intensity. It means that the triangular holes can nearly be neglected. When the ratio is larger than 2, the proportion of the area of triangular holes increases with γ. It becomes unacceptable for γ larger than a threshold. In that case triangular holes can not be neglected any more. For non-triangular holes, a homology-based algorithm is proposed to detect them. Simulation results show that the algorithm can detect 99% such holes.

APPENDIX A
PROOF OF THEOREM 2

Proof: : We first prove the lower bound. It can be obtained from (2) that

$$p(\lambda, \gamma) > P\{ \bigcup_{\{n_1,n_2\} \subseteq \Phi \setminus \{\tau_0(\Phi)\}} T(\tau_0, n_1, n_2)\}$$

So for the lower bound, we only consider the case that the closest node τ_0 can contribute to a triangle which bounds a triangular hole containing the origin.

Using polar coordinates, we assume the closest node τ_0 lies on (d_0, π). It is well known that the distance d_0 is a random variable with distribution

$$F_{d_0}(r_0) = P\{d_0 \leq r_0\} = 1 - e^{-\lambda \pi r_0^2} \qquad (7)$$

Therefore the probability of the first case can be given as

$$P\{ \bigcup_{\{n_1,n_2\} \subseteq \Phi \setminus \{\tau_0(\Phi)\}} T(\tau_0, n_1, n_2)\}$$
$$= \int P\{ \bigcup_{\{n_1,n_2\} \subseteq \Phi'_{r_0}} T((r_0, \pi), n_1, n_2)\} F_{d_0}(dr_0) \qquad (8)$$

where Φ'_{r_0} is the restriction of Φ in $B(O, R_c) \setminus B(O, r_0)$.

Once the node τ_0 is determined, a second node τ_1 must lie in the shadow region A^+ shown in Fig. 5 and a third node τ_2 must lie in the region S^- shown in Fig. 5, as illustrated in Section IV-C. The node $\tau_1 = (d_1, \theta_1)$ is assumed to have the smallest polar angle in A^+, which means that there should be no nodes with a polar angle less than θ_1 in A^+, that is to say no nodes are in the region

$$S^+(\tau_0, \tau_1) = S^+(d_0, \theta_1) = A^+ \bigcap H^+(\theta_1)$$

Since the intensity measure of the PPP in polar coordinates is $\lambda r dr d\theta$, the density F_{τ_1} of τ_1 can be given as

$$F_{\tau_1}(dr_1, d\theta_1) = \lambda r_1 e^{-\lambda |S^+(d_0, \theta_1)|} dr_1 d\theta_1 \qquad (9)$$

The integration domain $D(d_0)$ with respect to parameters (d_1, θ_1) can be easily obtained. From the construction of the region A^+, we can get $\alpha_0 = 2 \arccos(R_c/(2d_0))$ and $\alpha_1 = 2 \arcsin(R_c/(2d_0)) - 2 \arccos(R_c/(2d_0))$. So $\alpha_0 \leq \theta_1 \leq \alpha_1$ and $d_0 < d_1 \leq R_1(d_0, \theta_1)$, where

$$R_1(d_0, \theta_1) = \min(\sqrt{R_c^2 - d_0^2 \sin^2 \theta_1} - d_0 \cos \theta_1,$$
$$\sqrt{R_c^2 - d_0^2 \sin^2(\theta_1 + \alpha_0)} + d_0 \cos(\theta_1 + \alpha_0))$$

Furthermore, the area $|S^+(r_0, \theta_1)|$ can be expressed as

$$|S^+(r_0, \theta_1)| = \int_{\alpha_0}^{\theta_1} \int_{r_0}^{R_1(r_0, \theta)} r dr d\theta$$

As illustrated in Section IV-C, assume only τ_0, τ_1 and nodes in $S^-(\tau_0, \tau_1)$ can contribute to the triangle which bounds a triangular hole containing the origin, we can get a lower bound of the probability that the origin is inside a triangular hole. Based on the assumption, we have

$$P\{ \bigcup_{\{n_1,n_2\} \subseteq \Phi'_{r_0}} T((r_0, \pi), n_1, n_2)\}$$
$$> P\{ \bigcup_{n_2 \subseteq \Phi'_{r_0} \bigcap S^-(\tau_0, \tau_1)} T((r_0, \pi), \tau_1, n_2)\}$$
$$= \iint_{D(r_0)} P\{ \bigcup_{\substack{n_2 \subseteq \Phi'_{r_0} \bigcap \\ S^-(r_0, r_1, \theta_1)}} T((r_0, \pi), (r_1, \theta_1), n_2)\} F_{\tau_1}(dr_1, d\theta_1)$$
$$= \iint_{D(r_0)} P\{\Phi'_{r_0}(S^-(r_0, r_1, \theta_1)) > 0\} F_{\tau_1}(dr_1, d\theta_1)$$
$$= \iint_{D(r_0)} (1 - e^{-\lambda |S^-(r_0, r_1, \theta_1)|}) F_{\tau_1}(dr_1, d\theta_1)$$

$$(10)$$

where $|S^-(r_0, r_1, \theta_1|$ can be expressed as

$$|S^-(r_0, r_1, \theta_1)| = \int_{\theta_{2l}}^{-\alpha_0} \int_{r_0}^{R_2(r_0, r_1, \theta_1, \theta_2)} r dr d\theta_2$$

and

$$\theta_{2l} = \theta_1 - \arccos \frac{\cos(R_c/R) - \cos\theta_1 \cos\theta_0}{\sin\theta_1 \sin\theta_0}$$

$$R_2(r_0, r_1, \theta_1, \theta_2) = \min(\sqrt{R_c^2 - r_0^2 \sin^2\theta_2} - r_0\cos\theta_2,$$

$$\sqrt{R_c^2 - r_1^2 \sin^2(\theta_2 - \theta_1)} + r_1\cos(\theta_2 - \theta_1))$$

Therefore, from (7), (8), (9) and (10), the lower bound shown in (3) can be derived.

As for the upper bound, replace $|S^-(r_0, r_1, \theta_1)|$ by $|S^-(r_0, r_0, \theta_1)|$, we can get the upper bound as illustrated in Section IV-C. ∎

REFERENCES

[1] Q. Fang, J. Gao, and L. Guibas, "Locating and bypassing routing holes in sensor networks," in *Proc. IEEE INFOCOM*, vol. 4, Mar. 2004, pp. 2458–2468.
[2] G. Wang, G. Cao, and T. La Porta, "Movement-assisted sensor deployment," in *Proc. IEEE INFOCOM*, vol. 4, Mar. 2004, pp. 2469–2479.
[3] C. Zhang, Y. Zhang, and Y. Fang, "Localized algorithms for coverage boundary detection in wireless sensor networks," *Wirel. Netw.*, vol. 15, no. 1, pp. 3–20, Jan. 2009.
[4] ——, "Detecting coverage boundary nodes in wireless sensor networks," in *Proc. IEEE International Conference on Networking, Sensing and Control*, Apr. 2006, pp. 868–873.
[5] Y. Bejerano, "Simple and efficient k-coverage verification without location information," in *Proc. IEEE INFOCOM*, Apr. 2008, pp. 897–905.
[6] ——, "Coverage verification without location information," *IEEE Trans. Mobile Comput.*, vol. 11, no. 4, pp. 631–643, Apr. 2012.
[7] V. de Silva, R. Ghrist, and A. Muhammad, "Blind swarms for coverage in 2-d," in *Proc. Robotics: Science and Systems*, Jun. 2005, pp. 335–342.
[8] V. de Silva and R. Ghrist, "Coverage in sensor networks via persistent homology," *Algebraic & Geometric Topology*, vol. 7, pp. 339–358, 2007.
[9] R. Ghrist and A. Muhammad, "Coverage and hole-detection in sensor networks via homology," in *Proc. 4th International Conference on Information Processing in Sensor Networks*, Apr. 2005, pp. 254–260.
[10] A. Muhammad and M. Egerstedt, "Control using higher order laplacians in network topologies," in *Proc. 17th International Symposium on Mathematical Theory of Networks and Systems*, Jul. 2006, pp. 1024–1038.
[11] A. Tahbaz-Salehi and A. Jadbabaie, "Distributed coverage verification in sensor networks without location information," *IEEE Trans. Autom. Control*, vol. 55, no. 8, pp. 1837–1849, Aug. 2010.
[12] B. Liu and D. Towsley, "A study of the coverage of large-scale sensor networks," in *Proc. IEEE MASS*, Oct. 2004, pp. 475–483.
[13] P.-J. Wan and C.-W. Yi, "Coverage by randomly deployed wireless sensor networks," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2658–2669, Jun. 2006.
[14] L. Lazos and R. Poovendran, "Stochastic coverage in heterogeneous sensor networks," *ACM Trans. Sen. Netw.*, vol. 2, no. 3, pp. 325–358, Aug. 2006.
[15] X. Li, D. K. Hunter, and S. Zuyev, "Coverage properties of the target area in wireless sensor networks," *IEEE Trans. Inf. Theory*, vol. 58, no. 1, pp. 430–437, Jan. 2012.
[16] F. Yan, P. Martins, and L. Decreusefond, "Accuracy of homology based approaches for coverage hole detection in wireless sensor networks," in *Proc. IEEE ICC*, Jun. 2012, pp. 497–502.
[17] V. D. Silva and R. Ghrist, "Coordinate-free coverage in sensor networks with controlled boundaries via homology," *International Journal of Robotics Research*, vol. 25, no. 12, pp. 1205–1222, December 2006.
[18] V. de Silva and R. Ghrist, "Homological sensor networks," *Notices of the American Mathematical Society*, vol. 54, no. 1, pp. 10–17, Jan. 2007.
[19] A. Muhammad and A. Jadbabaie, "Decentralized computation of homology groups in networks by gossip," in *Proc. American Control Conference*, Jul. 2007, pp. 3438–3443.
[20] P. Dlotko, R. Ghrist, M. Juda, and M. Mrozek, "Distributed computation of coverage in sensor networks by homological methods," *Applicable Algebra in Engineering, Communication and Computing*, vol. 23, no. 1-2, pp. 29–58, April 2012.
[21] D. Dong, Y. Liu, K. Liu, and X. Liao, "Distributed coverage in wireless ad hoc and sensor networks by topological graph approaches," in *Proc. IEEE ICDCS*, Jun. 2010, pp. 106–115.
[22] G. S. Kasbekar, Y. Bejerano, and S. Sarkar, "Lifetime and coverage guarantees through distributed coordinate-free sensor activation," in *Proc. ACM MobiCom*, Sep. 2009, pp. 169–180.
[23] M. A. Armstrong, *Basic Topology*. Springer, 1983.
[24] J. Munkres, *Elements of algebraic topology*. Addison-Wesley, 1984.
[25] A. Hatcher, *Algebraic Topology*. Camebridge University Press, 2002.
[26] F. Yan, P. Martins, and L. Decreusefond, "Connectivity-based distributed coverage hole detection in wireless sensor networks," in *Proc. IEEE GLOBECOM*, December 2011, pp. 1–6.
[27] C. Bettstetter and J. Zangl, "How to achieve a connected ad hoc network with homogeneous range assignment: an analytical study with consideration of border effects," in *Proc. 4th International Workshop on Mobile and Wireless Communications Network*, Sep. 2002, pp. 125–129.
[28] D. Dong, Y. Liu, and X. Liao, "Fine-grained boundary recognition in wireless ad hoc and sensor networks by topological methods," in *Proc. ACM MobiHoc*, May 2009, pp. 135–144.
[29] A. Vergne, L. Decreusefond, and P. Martins, "Reduction algorithm for simplicial complexes," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 475–479.
[30] B. Tong and W. Tavanapong, "On discovering sensing coverage holes in large-scale sensor networks," Computer Science, Iowa State University., Tech. Rep. TR 06-03, Mar. 2006.