

Chapter 5

Coverage and Connectivity

A major advantage of wireless sensor networks (WSNs) over wired networks is the potential for ad hoc deployment of the network. If the monitoring of a dangerous environment is required, then one may not be able to deploy a wired network. However, a WSN may be deployed in even the most inhospitable of domains, and the sensor data can then be gathered and monitored from a remote location. A likely ad hoc deployment method for distributing the sensor nodes in such a hazardous terrain is to airdrop them over the region of interest. In a chemical monitoring application, for instance, wireless sensor nodes may be airdropped over an area which could be unsafe to manually deploy the nodes and then self-organize them into a network for this monitoring task.

The problem with this deployment method, as well as other random deployment methods often used with WSNs, is that one has no full control over where the nodes will be located. Consider, as an example, the application of battlefield surveillance in which a large number of sensor nodes, designed to detect and locate the presence of enemy snipers [30], are airdropped over the battlefield. Nodes may use acoustic sensing to detect gunshots fired by snipers and approximate the location of a shooter by aggregating the time of arrival (ToA) data from several sensors. Crucial to this application, as well as many others, is the often implicit assumption that the deployment of sensor nodes is complete such that any event occurring within the sensor field will be detected. However, due to the randomness of a deployment there is no guarantee of complete coverage, raising questions on how to verify the absence of holes in the sensor coverage within the region of interest, and how to locate such holes in coverage if they do exist.

The coverage of the area of interest is one of the Quality of Service (QoS) metrics in WSNs [18] as it describes how well the sensing field is sensed or covered by the sensor nodes. A point in the region of interest is considered to be covered if one or more sensors can measure the phenomena of interest at that point. If every point in the sensing field is covered by at least one sensor node, we say that the coverage is complete, and such coverage has degree of one. Sometimes specific

applications require higher fault redundancy or sensor network measurement accuracy, resulting in higher degree of coverage where multiple sensor nodes cover point(s) in the sensing field.

Related to coverage is connectivity—another important characteristic of WSNs. As described later in the chapter, sensor networks can be modeled using graphs where nodes are equivalent to vertices and communication links are represented by corresponding edges. If the equivalent graph is connected then we consider the sensor network to be connected. The graph (or sensor network) is connected if there is an edge path (communication path) between any two vertices (nodes) in the graph (network). Otherwise, the graph (network) is disconnected and some nodes are not able to communicate with the rest of the network. Disconnected network nodes are basically useless for the network since the information on those nodes is not accessible by the network anymore. Similarly, as with the coverage in the sensor network, connectivity can have higher degrees that allows for more robust networks where removal or failure of some nodes does not cause the network to become disconnected.

Notions of coverage and connectivity are, in most practical applications, not independent of each other. Most applications have strict requirements for both of these network characteristics. It is often required to provide a certain level (quality) of coverage, while maintaining the network connectivity. This translates into constrained optimization problems, discussed later in the chapter, where type of coverage determines optimization cost function and connectivity relates to constraints. Similarly, it might be required to deploy a robust network that is fault-tolerant with a higher degree of connectivity, while the sensing area is still fully covered.

In this chapter, we review important notation and results related to sensor network coverage and connectivity and tools that are commonly used in modeling of sensor networks coverage and connectivity. We present basic graph notations that are used in WSN modeling including tools used in coverage modeling and provide few examples of optimal coverage under sensor network connectivity constraints.

5.1 Modeling Sensor Networks Using Graphs

In this section, we cover modeling of wireless sensor network using certain mathematical constructs known as graphs and simplicial complexes. This mapping of the sensor network allows users to perform mathematical and computational analysis of the network's topology, to analyze network coverage and connectivity, and develop algorithms that deal with those issues. Such modeling is, in general, applicable to both networks with known coordinates and coordinate-free networks.

5.1.1 Communication Graphs

A *graph*, defined as $G = (V, E)$ describes a set V of vertices and a set E of edges that connect the vertices. In most cases in WSN applications, it is presumed that all communication links between sensor nodes are bidirectional. This means that if node A is within the communication range of node B , then the reverse is also true. For this reason, this discussion of graphs may be restricted to deal only with *undirected simple graphs*. A graph is called a simple graph if it does not contain multiple edges between pairs of vertices or self-loops, which connect a vertex to itself, and it is called undirected if all edges are bidirectional. For simplicity, the term graph is often used to refer to such graphs. Figure 5.1 illustrates the graph $G = (V, E)$ with vertex set $V = \{1, 2, 3, 4\}$, and edge set $E = \{12, 13, 14, 24, 34\}$. Note also that the edge set for an undirected graph may consist of unordered pairs of vertices such that the edge 12 is equivalent to the edge 21.

Here, we define terms such as vertex degrees and order of a graph [54] that are used in WSNs modeling.

Definition 5.1 The degree of vertex v in a graph G , $d(v)$ is the number of edges incident to v . If the maximum degree in a graph is equal to the minimum degree in the graph, then the graph is called regular graph. The graph is called k -regular if the common degree is k .

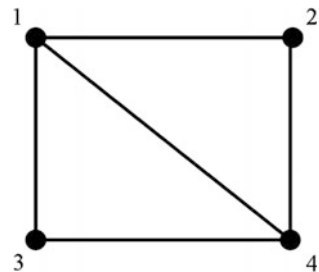
For example, the degree of vertex 1 in the graph in Fig. 5.1 is 3, while the degree of vertex 3 is 2.

Definition 5.2 The order of a graph G , $n(G)$, is the number of vertices in graph G . The size of a graph G , $e(G)$, is the number of edges in graph G .

Definition 5.3 [54] A graph G is connected if it has a u, v -path whenever $u, v \in V(G)$; otherwise, G is disconnected. If graph G has a u, v -path, then vertex u is connected to vertex v in G .

Directed Graphs Communication links in sensor networks does not need to be symmetric. If the node A can hear the node B, it does not necessary mean that the node A can hear the node B. This can happen due to several reasons including node

Fig. 5.1 Example of a simple graph with four vertices and five edges



failure, different energy levels at the node that force radio to operate at different power levels, etc. A model that represents such a network is called directed graph.

Definition 5.4 [54] A directed graph G is a triple consisting of a vertex set $V(G)$, an edge set $E(G)$, and a function that assigns to each edge an ordered pair of vertices. The first vertex of the ordered pair is the tail of the edge, and the second is the head.

An example of a directed graph is given in Fig. 5.2, where there is an edge from 3 to 1, from 3 to 2, and from 3 to 4. There is also an edge from 1 to 2, and from 4 to 2.

Definition 5.5 An adjacency matrix $A(G)$ of a directed graph G is a matrix that at the position i, j has the number of edges from v_i to v_j .

For example, an adjacency matrix for the graph in Fig. 5.2 is given by

$$A(G) = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \end{matrix} & \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix} \end{matrix}. \quad (5.1)$$

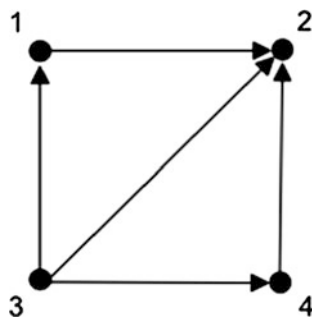
If the graph is weighted, then for each pair of vertices u, v , there is a assigned weight a_{uv} , [34]. We consider a graph with weights that are real, satisfying $a_{uv} = a_{vu}$, $a_{uv} \neq 0$ if and only if u and v are adjacent vertices, and $a_{uv} \geq 0$. In case of weighted graphs, the adjacency matrix is $A(G) = [a_{uv}]_{u,v \in V(G)}$ and the degree of a vertex v is equal to a sum of all weights that are adjoining to the vertex v , i.e., $d(v) = \sum_u a_{uv}$.

Definition 5.6 A complement of a graph G is a graph \bar{G} that has the same set of vertices and a complement set of edges, i.e., there is an edge between any two vertices in \bar{G} if there is no edge between the same vertices in G .

Note that together, a graph G and its complement \bar{G} form a complete graph with all possible edges, see Fig. 5.3.

WSNs are commonly mapped to corresponding graphs that represent connectivity of the nodes and can be used to solve certain problems. The set of sensor

Fig. 5.2 Example of a directed graph



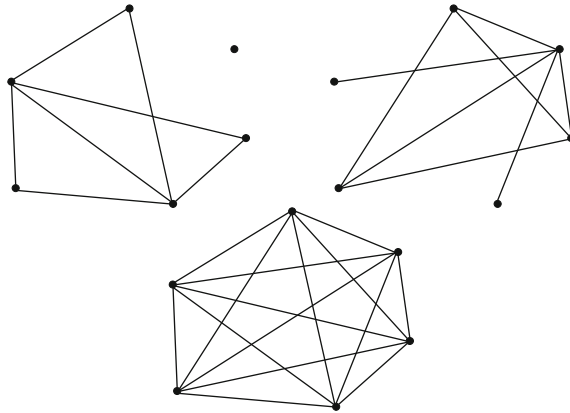


Fig. 5.3 A graph G , its compliment \bar{G} and a complete graph $G + \bar{G} = K_n$

nodes N in the network becomes the set of vertices in the graph. Two nodes are then connected with a graph edge if there is a direct communication link between them. By direct communication link, we mean only those nodes that can communicate directly without multi-hopping. This is illustrated in Fig. 5.4 below where a sensor network is represented as a graph.

Assuming that communication range between sensor nodes is r_c and that the communication model is binary (when inside the communication range nodes can communicate, outside the range they cannot), then any two nodes whose Euclidean distance is less than r_c have a corresponding graph edge in the equivalent communication graph. Equivalently, if two nodes lie in each other's communication

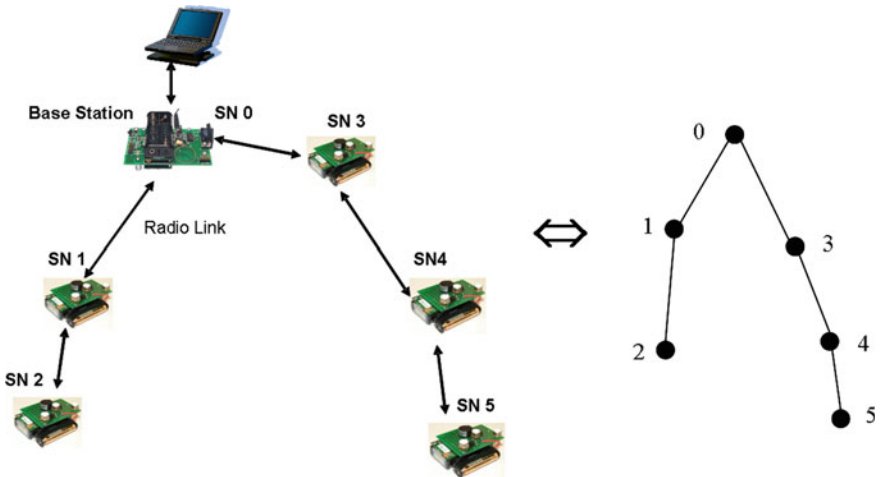
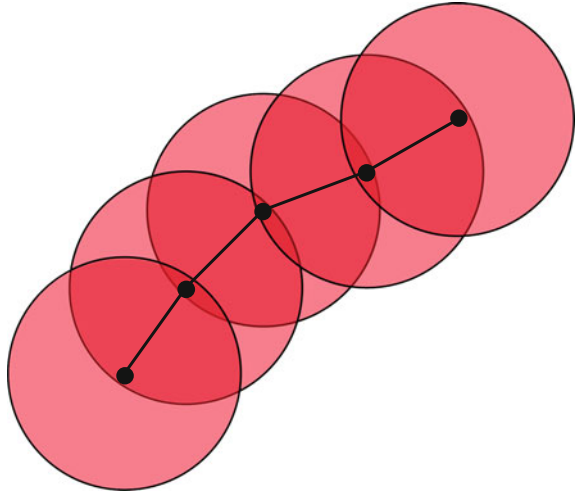


Fig. 5.4 Mapping a sensor network to a graph

Fig. 5.5 A sensor network and its communication graph —a unit disk graph



disk, then there is an edge between them. Therefore, the sensor network is modeled using a specific type of graph known as a *unit disk graph*, which admits a graph edge between two vertices if their Euclidean distance is less than a fixed threshold. For communication networks, this graph is commonly referred to as the *communication graph* of the network. An example of a communication graph for a simple network is given in Fig. 5.5. The disks around each node represent the communication range r_c of the nodes.

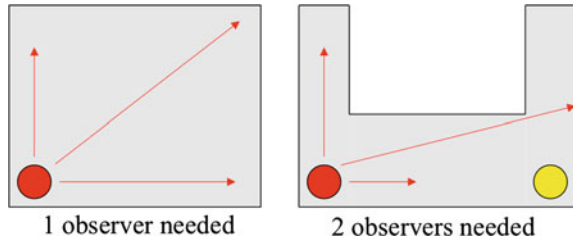
5.2 Coverage

One of the fundamental problems in the field of WSNs is the coverage. Coverage is an important indicator of the Quality of Service (QoS) in a sensor network [33]. The coverage problem can be approached in a number of ways due to the broad range of possible sensor network applications, but the central goal is to determine how well a set of sensor nodes monitors a given area.

The sensor coverage problem is closely related to the art gallery problem [37], a well-known visibility problem from computational geometry. The objective of this problem, which was originally proposed in August 1973 by mathematician Victor Klee, is to determine a minimum number of observers needed for surveillance of an art gallery such that every point can be seen by at least one observer.

This problem is illustrated in Fig. 5.6. Note that for any convex region, such as that to the left, only one observer is required. For the region to the right, two observers are needed to monitor the entire area since one observer to the lower left corner cannot see the portion at the top right.

Fig. 5.6 The art gallery problem example



This problem is shown to be NP-hard in [31]. Reference [7] proves that $\lfloor n/3 \rfloor$ observers are always sufficient and the solution is sometimes optimal in 2D space, where n is the number of sides in the polygonal region being monitored, and the floor function $\lfloor x \rfloor$ is defined as the largest integer not greater than x (for example, $\lfloor 3.2 \rfloor = 3$, $\lfloor -3.2 \rfloor = -4$). This result is commonly referred to as Chvatal’s art gallery theorem. A linear-time algorithm for locating these $\lfloor n/3 \rfloor$ observers is given in [27]. An approximation algorithm for the 3D case is proposed in [32] which is within $O(\log n)$ of the optimal solution.

In [16], coverage is classified into three main categories: blanket coverage, barrier coverage, and sweep coverage. The goal of blanket coverage, as name suggests, is to maximize the coverage of area of interest. This is the common objective for a majority of WSN monitoring applications. Barrier coverage attempts to minimize the probability of an intruder penetrating the barrier and is important security feature in WSN applications. An example of barrier coverage is border patrol, where a nation attempts to keep people from crossing its borders illegally [3, 29]. Finally, sweep coverage is the equivalent of a barrier moving across the area of interest where a set of sensors sweeps specific area. This incorporates aspects of both blanket coverage and barrier coverage. These three types of coverage are shown below in Figs. 5.7, 5.8, and 5.9, respectively.

A further issue to consider with coverage problems is point coverage versus area coverage. Either the sensor nodes can focus coverage on certain areas or targets, referred to as point coverage, or they can be spread out in an attempt to cover an

Fig. 5.7 Blanket coverage

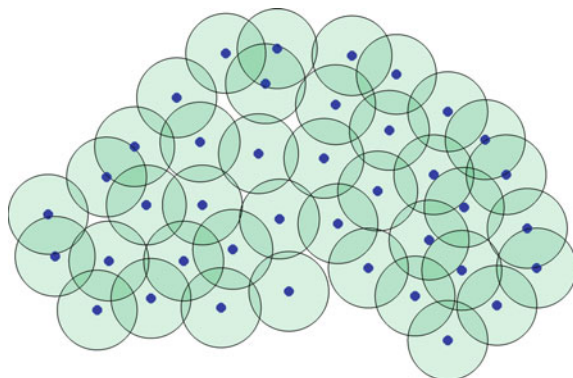
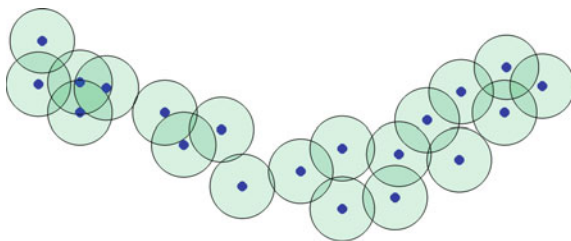
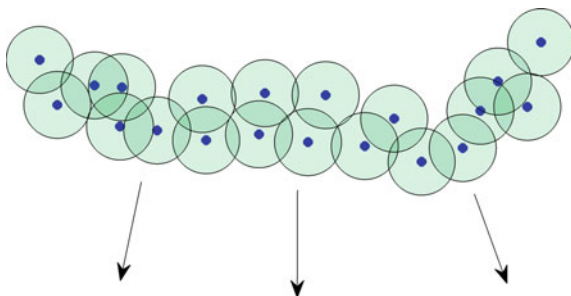


Fig. 5.8 Barrier coverage**Fig. 5.9** Sweep coverage

entire area, referred to as area coverage. Typically, in most common WSN applications, area coverage is used. However, there are some situations, such as in tracking applications, where point coverage may be more appropriate.

The topic of sensing coverage is often approached by studying how to position the sensors such that the area of interest is sufficiently monitored. A non-complete coverage in WSNs (coverage with holes) allows for intrusion into the system and presents a security issue where a WSN needs to deal with intrusion detection in some other way [2, 42]. Several authors have considered the problem of optimally positioning sensor such that a region is maximally covered [6, 23, 26, 58]. The central idea is to spread out the nodes as much as possible to improve sensing coverage over a region of interest. Other results have focused on finding a minimal number of sensors needed to monitor an area [1, 38]. This approach is especially useful in an over-deployed network where one can find a minimal subset of the sensors that will maintain the desired level of coverage and power off all other nodes to reduce power consumption [47, 48, 52, 57].

5.2.1 Coverage Holes

The problem of locating coverage holes in a coordinate-free network has been studied in [13–15]. A strictly graph theoretic approach is used to locate nodes on the boundary of holes; however, no clear definition is given of what constitutes a hole for the proposed method. Additionally, [14] provides a graph theoretic method for

using only local connectivity information (i.e., no coordinates) to develop a rough sketch of the layout of the network. The relationship between sensing coverage and node connectivity in sensor networks is given in [57]. This fundamental result states that *if the communication range of the sensor nodes is at least twice the sensing range then complete sensing coverage of a convex region implies connectivity of the network*. A generalization of this result to higher degrees of coverage, often referred to as k -coverage, is presented in [52]. A sensor field is called k -covered if every point is covered by at least k sensors. Under the same assumption, that the communication range be at least twice the sensing range, k -coverage implies k -connectivity.

Local connectivity information, without any positioning information, was used in [19, 40, 45] to determine if a region is covered or not [23]. Mathematical models of homology [45] were used to determine the coverage. A further extension of this work [19] shows how to detect the boundary of a hole in the network. This method only works for sensor networks that have a single hole and for networks having multiple holes it likely will not identify all holes. Furthermore, there is no guarantee that the algorithm will exactly locate even single holes. These references present a method of determining coverage without knowing the positions of the sensors. The major limitation is that there must be a certain ratio between the sensing range and the communication range of the sensor nodes or connectivity information alone is not enough to imply sensing coverage.

Whereas much study has been focused on how to improve coverage in sensor networks, research in [25] seeks to identify any holes in coverage that may be present in the existing sensor network deployment. Only after verifying that coverage is not sufficient would attempts be made to improve coverage or patch holes.

If the area of interest is not completely covered, it is called *coverage holes*, or simply *holes*. It is in these holes that an event may go undetected. Figure 5.10 gives an example of a sensor network with fully 1-covered sensor field and a sensor network with a coverage hole.

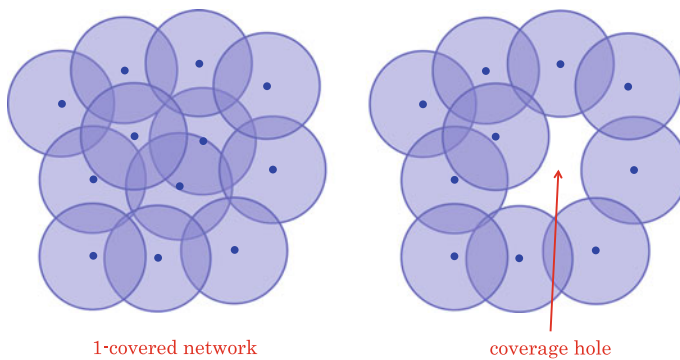


Fig. 5.10 Fully covered (1-covered) sensor field (*left*) and deployment with a coverage hole (*right*)

After determining the existence of holes it is also desirable to locate the holes so that, if necessary, they can be patched. Detecting the location of holes may be useful for other reasons as well. The presence of a hole may indicate a feature of the terrain, such as a lake, or some other phenomenon, such as a fire, which cannot be patched by the simple deployment of additional sensors. Also, messages are more likely to be routed through nodes on hole boundaries, depleting their available battery power more quickly, which could finally enlarge the hole [56] and degrade the network performance.

Assuming a simple, omnidirectional sensing model, each sensor node has a certain sensing range r_s and a transmission, or communication, range r_c .

In real-life applications, sensing and communication models are not always omnidirectional. Such models also heavily depend on environmental conditions.

Example 5.1 The IRIS sensor nodes are equipped with omnidirectional antennas capable of transmitting signal up to 500 m. That capability is produced in environments where there are no obstructions and the EM signal is allowed to propagate without interference. In the case of an urban environment, the sensing range would be considerably less; moreover, other factors such as multipath signal reflections, shadowing, and path loss due to non-free space propagation also affect the accuracy of the model. The received signal strength gradually decreases with the distance from the transmitter. The received signal strength is inversely proportional to the square of the distance between the transmitting and receiving antennas. Mathematically, it can be written as

$$\frac{P_r}{P_t} = \left[\frac{\sqrt{G_t \lambda}}{4\pi d} \right]^2, \quad (5.2)$$

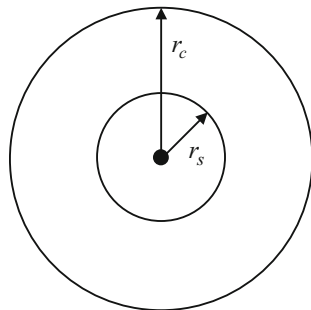
where, P_r is the received power, P_t is the transmitted power, λ is the wavelength of the signal, d is the distance between receiver and transmitter, and G_t is the product of the transmit and receive antenna field radiation pattern. Because the antennas on the nodes are omnidirectional, the factor G_t is equal to 1.

Consider a set $S \subset \mathfrak{R}^2$ that represents a sensing area of interest. A network of N equal sensor nodes is deployed over S with sensor locations at (x_i, y_i, z_i) . A sensing function is given by $p_i(q)$ where p_i is the probability that the event $q \in S$ will be detected. Such probability represents the sensing model and in its simplest form the probability can take the form of a uniform probability density function [20]

$$p_i(q) = \begin{cases} 1, & d_i \leq r_s \\ 0, & d_i > r_s \end{cases} \quad (5.3)$$

or

Fig. 5.11 Sensing and communication range of a sensor node



$$p_i(q) = \begin{cases} P_t \left[\frac{\sqrt{G_i \lambda}}{4\pi d_i} \right]^2, & d_i \leq r_s \\ 0, & d_i > r_s \end{cases} \quad (5.4)$$

where r_s is the sensing radius of the omnidirectional antenna and d_i is a distance between the location of a sensor node i and specific sensing point in the field q , i.e.,

$$d_i = \|(x_i, y_i, z_i) - q\|. \quad (5.5)$$

In case of a uniform sensing probability function, the radius r_s around each node is referred to as its sensing disk, see Fig. 5.11. It is within this disk that the sensor will detect the desired phenomenon with probability of one. This is sometimes referred to as a binary sensing model since an event is either detected or it is not detected. Probabilistic models are also discussed in the literature [21, 58] and may be more realistic than binary sensing model. The same is true for communication models that affect connectivity of WSNs. However, when coverage and connectivity are discussed, usually the simplest binary models are considered (sensors can either sense or not, nodes can either communicate with other nodes or not).

5.3 Connectivity

Deployment of wireless sensor networks requires minimization of cost, reduction in computation and communication, high-degree of sensing area coverage, while maintaining a connected network. If the network is not connected, information cannot flow from deployed sensors towards the network gateway or a base station. If a network is modeled as a graph, then network connectivity is equivalent to graph connectivity, i.e., there exists a path between any two vertices (nodes) in the graph. Equivalently to the degree of coverage, there is a degree of network connectivity

where k -connectivity means that removing any $k - 1$ nodes still leaves the graph connected. Several deployment algorithms that optimize some aspect of coverage, while maintaining network connectivity, are given in [18]. Potential field algorithm provides sensor nodes mobility where coverage can be maximized, while the network is still connected [39]. Decentralized estimation and control of graph connectivity for WSNs with mobile nodes is given in [55].

The concept of potential field is an artificially created field where static and mobile nodes are subject to specifically designed forces. Since the goal is to maximize the coverage while maintain the connectivity, two forces were suggested in [39]: F^R —a repulsive force that move mobile sensor nodes away from each other to maximize the coverage; and F^A —an attractive force that pulls nodes towards each other to stay connected. Both forces are inversely proportional to the square of the distance between node pairs and are given by

$$F^R(i, j) = \frac{-K_r}{(\text{dist}[(x_i, y_i), (x_j, y_j)])^2} \frac{(x_i, y_i) - (x_j, y_j)}{\text{dist}[(x_i, y_i), (x_j, y_j)]}$$

$$F^A(i, j) = \begin{cases} \frac{-K_a}{(\text{dist}[(x_i, y_i), (x_j, y_j)] - R_c)^2} \frac{(x_i, y_i) - (x_j, y_j)}{\text{dist}[(x_i, y_i), (x_j, y_j)]}, & \text{for critical connect.} \\ 0, & \text{otherwise} \end{cases} \quad (5.6)$$

where (x_i, y_i) is an x, y location of a sensor node i , and K_r and K_a are design constants. Nodes are initially positioned at one place, and then start repelling from each other. When they are only k -connected neighbors left for a single node (critical connection), the attractive force starts pulling sensor nodes, thus preventing further reduction in sensor nodes connectivity. The nodes reach equilibrium state when sum of all forces acting on a node is equal to zero. The method provides an elegant, distributed solution for deployment of mobile sensor nodes under k -connectivity constraints.

Distributed algorithm that applies virtual forces is presented in [22]. Such algorithm maximizes the coverage and maintains uniform distribution of sensor nodes. The method assumes that nodes have information about their coordinates. The virtual force is proportional to the distance between sensor nodes and *expected density* of sensor nodes in the sensing field. The force is given by

$$F(i, j) = \frac{\mu(i)}{\mu^2(R_c)} (R_c - \text{dist}[(x_i, y_i), (x_j, y_j)]) \frac{(x_i, y_i) - (x_j, y_j)}{\text{dist}[(x_i, y_i), (x_j, y_j)]}, \quad (5.7)$$

where $\mu(i)$ is the local node density the at sensor node i , and the expected density $\mu(R_c)$. The expected density depends on communication radius and is given by

$$\mu(R_c) = \frac{N\pi R_c^2}{A}, \quad (5.8)$$

with N being the number of sensor nodes, R_c communication radius, and A the sensing area. The algorithm provides movement of the nodes based on virtual forces and finishes when there are no new movements of nodes.

A centralized approach using virtual forces is given in [58, 60]. Each sensor has three virtual forces that act upon the node: repulsive force from the obstacles, attractive force from the sensing areas of high interest, and forces from other nodes. The specific forces from other nodes can either be repulsive or attractive, depending on their mutual distance and design parameters. The force applied to each sensor node is given by

$$F_i = \sum_{j \in N_i} F_{ij} + F_i^R + F_i^A, \quad (5.9)$$

where F_{ij} is the force (repulsive or attractive) from node j to node i , F_i^R is the repulsive force applied to the node, and F_i^A is the attractive force applied to the node. The force F_{ij} is given by

$$F_{ij} = \begin{cases} w_A (\text{dist}[(x_i, y_i), (x_j, y_j)] - d_{th}) \angle \vec{ij}, & \text{if } \text{dist}[(x_i, y_i), (x_j, y_j)] > d_{th} \\ 0, & \text{if } \text{dist}[(x_i, y_i), (x_j, y_j)] = d_{th} \\ \frac{w_B}{\text{dist}[(x_i, y_i), (x_j, y_j)]} \angle \vec{ij} + \pi, & \text{if } \text{dist}[(x_i, y_i), (x_j, y_j)] < d_{th} \end{cases}, \quad (5.10)$$

where \vec{ij} is the vector from node i to node j , d_{th} is the threshold value for the mutual distance between nodes, and w_A and w_B are the constants determining amplitude of the forces. The method provides centralized deployment algorithm with controlled connectivity of the network. The sign of the force changes when distance between nodes crosses d_{th} threshold, allowing for controlled connectivity of the network. The computational complexity of the virtual force algorithm is $O(nmk)$ for a network of n nodes deployed on the $m \times k$ grid.

5.3.1 Graph Laplacian

The Laplacian matrix of a network communication graph is of fundamental importance in network connectivity, network diameter, mean distance, number of connected components of a graph and numerous other applications, [34]. We review a standard graph theory terminology that includes adjacency matrix and the Laplacian matrix of graphs. In Sect. 5.1.1 we defined an adjacency matrix $A(G)$ of a directional graph G as a matrix that at the position i, j has the number of edges from v_i to v_j . We also introduced a degree of a vertex v_i .

Let $D(G)$ be a diagonal matrix indexed by a vertex set $V(G)$, $D(G) = \text{diag}\{d(v_1), d(v_2), \dots, d(v_n)\}$. The *Laplacian* matrix of a graph G is defined as

$$Q(G) = D(G) - A(G). \quad (5.11)$$

The Laplacian matrix is also called *Kirchhoff* matrix or admittance matrix and it finds its application in electrical networks. The Laplacian matrix is positive semi-definite and symmetric matrix with the additional property that all rows sum to zero (show this property as an exercise). The characteristic polynomial of $Q(G)$ is given by

$$\mu(G, \lambda) = \det(\lambda I - Q), \quad (5.12)$$

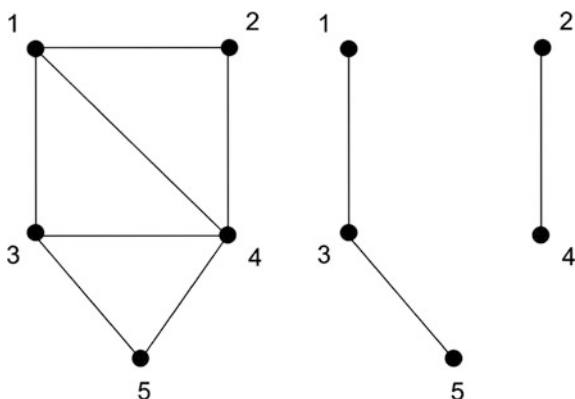
where I is the identity matrix of same dimensions as Q . The characteristic polynomial roots are the Laplacian eigenvalues $\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$ where n is the order of G .

Example 5.2 Consider two graphs shown in Fig. 5.12.

The adjacency matrix A_1 (for the graph on the left) and matrix A_2 (for the graph on the right) are given by

$$A_1 = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix}, A_2 = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}. \quad (5.13)$$

Fig. 5.12 Connected (*left*) and disconnected (*right*) graph



Corresponding Laplacian matrices are:

$$\begin{aligned}
 Q_1 &= \begin{bmatrix} 3 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 & 2 \end{bmatrix} - \begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix} \\
 &= \begin{bmatrix} 3 & -1 & -1 & -1 & 0 \\ -1 & 2 & 0 & -1 & 0 \\ -1 & 0 & 3 & -1 & -1 \\ -1 & -1 & -1 & 4 & -1 \\ 0 & 0 & -1 & -1 & 2 \end{bmatrix} \tag{5.14}
 \end{aligned}$$

$$\begin{aligned}
 Q_2 &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \\
 &= \begin{bmatrix} 1 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ -1 & 0 & 2 & 0 & -1 \\ 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 1 \end{bmatrix} \tag{5.15}
 \end{aligned}$$

and corresponding Laplacian eigenvalues are:

$$\mu(G_1) : 0, 1.58, 3, 4.41, 5 \tag{5.16}$$

$$\mu(G_2) : 0, 0, 1, 2, 3. \tag{5.17}$$

Note that the graph G_2 is disconnected and has two components, and its first two eigenvalues are equal to zero. The following theorem [34] relates spectrum of the Laplacian graph matrix and graph connectivity.

Theorem 5.1 *Let G be a weighted graph with all weights nonnegative. Then:*

- (a) $Q(G)$ has only real eigenvalues,
- (b) $Q(G)$ is positive semi-definite,
- (c) Its smallest eigenvalue is $\lambda_1 = 0$ and a corresponding eigenvector is $[1, 1, \dots, 1]^T$. The multiplicity of 0 as an eigenvalue of $Q(G)$ is equal to the number of components of graph G .

From Theorem 5.1, it follows that $\lambda_1 = 0$, $\lambda_2 > 0$ if and only if the graph G is connected. Therefore, the eigenvalues of the graph Laplacian serves as a measure of graph connectivity. The number $\lambda_2(G)$ is also called *algebraic connectivity* of the graph G as it directly relates to its connectivity [12]. If $\lambda_2 = 0$ then the graph (network) is disconnected. If $\lambda_2 > 0$, the graph (network) is connected, with the second eigenvalue λ_2 being a measure of the graph (network) connectivity. The following Theorem [34] lists few useful bounds on Laplacian eigenvalues.

Theorem 5.2 *Let $G(V, E)$ be a graph of order n . The following bounds are valid:*

- (a) $\lambda_2 \leq \frac{n}{n-1} \min\{d(v)\}$
- (b) $\lambda_n \leq \max\{d(u) + d(v)\}$ where $uv \in E(G)$
- (c) $\sum_{i=1}^n \lambda_i = 2|E(G)| = \sum_v d(v)$
- (d) $\lambda_n \geq \frac{n}{n-1} \max\{d(v)\}$.

The following result describes behavior of eigenvalues when additional edges are inserted in the communication graph [34]. This corresponds to a scenario when nodes move closer to each other, thus increasing their connectivity.

Theorem 5.3 *The eigenvalues of graph G and $G' = G + e$ satisfy*

$$0 = \lambda_1(G) = \lambda_1(G') \leq \lambda_2(G) \leq \lambda_2(G') \leq \dots \leq \lambda_n(G) \leq \lambda_n(G'). \quad (5.18)$$

The number of spanning trees is also related to the spectrum of Laplacian. Let $\kappa(G)$ be the number of spanning trees of the graph G of order n . The number of spanning trees is related to eigenvalues by

$$\kappa(G) = \frac{1}{n} \lambda_2(G) \lambda_3(G) \dots \lambda_n(G). \quad (5.19)$$

The second eigenvalue of a graph Laplacian is closely related to the graph diameter (the longest shortest path between any two graph vertices, i.e., the largest number of vertices that must be traversed to travel from one vertex to another), which relates to the sensor network coverage area. The upper and lower bounds on graph diameter are given by [34]

$$\frac{4}{n\lambda_2(G)} \leq \text{diam}(G) \leq 2 \left\lceil \frac{\Delta + \lambda_2(G)}{4\lambda_2(G)} \ln(n-1) \right\rceil, \quad (5.20)$$

where Δ is the maximum degree in graph G , $\Delta = \max\{d(v)\}$.

Applications of Laplacian eigenvalue-based analysis include mobile sensor networks, robotics swarms, cooperative control, consensus networks, and others. Most applications utilize the fact that the measure of network connectivity is the second

smallest eigenvalue of the Laplacian matrix of the graph. For instance, in [43], a decentralized connectivity maintenance control strategy for mobile robotic systems/networks is considered, i.e., a group of N agents with single-integrator dynamics

$$\dot{p}_i = u_i, \quad (5.21)$$

where p_i and u_i are position and control input, respectively, for the i -th agent. In [53], the connectivity maintenance control algorithm is given by

$$u_i = \frac{\partial \lambda_2}{\partial p_i}. \quad (5.22)$$

To improve the system stability from the connectivity point of view, it is proposed in [43] a modified control algorithm where control signal increases as the algebraic connectivity of the graph decreases (graph becomes “*less connected*”). The modified controls algorithms is given by

$$u_i = \operatorname{csch}^2(\lambda_2 - \varepsilon) \frac{\partial \lambda_2}{\partial p_i}, \quad (5.23)$$

where csch is the standard hyperbolic cosecant function and ε is the desired lower-bound for the value of λ_2 . This approach allows the control signal to increase when connectivity in the network is reduced.

5.4 Coverage Models Using Voronoi Diagrams

In Chap. 2 we provided an overview of Voronoi diagrams. Voronoi diagrams can be used for studying WSN deployment, coverage control, and hole patching. A vector-based algorithm was proposed in [51] where sensors move from densely covered areas to sparsely covered areas. Sensors act on each other with repulsive force. The virtual force from neighboring sensors will move the sensor such that the mutual distance is close to average distance between nodes. During each movement, each sensor calculates future coverage within its Voronoi cell. If the future movement will not improve the coverage, the sensor will not move to the target location and will instead move to the midpoint position between its target location and new location. Voronoi cell also has repulsive forces from the cell boundary, thus pushing the sensor node toward the inside of the region.

Voronoi-based algorithm moves sensor nodes towards the maximum coverage. It is a greedy algorithm where each sensor checks for holes within its Voronoi cell. If the hole is detected, the node moves towards its farthest vertex in the Voronoi cell. The distance between the farthest Voronoi vertex and its new location is equal

to the sensing range with a maximum moving distance of the node being equal to the half of the communication range.

A combination of mobile and static nodes offers reduced cost while networks still has flexibility of mobile networks. Voronoi diagrams can be used for a hole detection in such networks [50]. Static nodes map their Voronoi diagrams and calculate coverage holes within their Voronoi cells. Static nodes then start the bidding process in which a mobile node will move to patch the hole. If there is a hole, a static sensor chooses the Voronoi vertex that is the farthest from the sensor and calculates related bid. The bid is given by

$$B_i = \pi(d - r_s)^2, \quad (5.24)$$

where d is the distance between the sensor node and the Voronoi vertex, and r_s is the sensing range. Mobile nodes also have their own base price. The static node finds the mobile node whose base price is lower than its bid. The mobile node receives all bids from its neighbors, and based on the highest bid moves to heal the coverage hole. Once the mobile node accepts the bid, it updates its base price with that specific bid. This bidding-price model guarantees that the total size of holes will decrease over time, meaning no mobile node will patch the one hole, and create larger one at the same time.

5.5 Simplicial Complexes

A k -simplex is defined as the convex hull of a set of $k + 1$ points in \mathbb{R}^n . More intuitively, a k -simplex is simply an unordered $(k + 1)$ -tuple of points. For example, a 0-simplex (σ_0) is defined as a single point and a 1-simplex (σ_1) is defined as an unordered pair of points. These are identical to vertices and edges in graph theoretic terminology. A 2-simplex (σ_2), comprised a triple of points is informally a triangle that is “filled in” so that it is not hollow inside. A 3-simplex (σ_3), also called a tetrahedron, is defined by a set of four points, and is again, solid inside. In general, any k -simplex has a solid interior, though it is difficult to visualize this in higher dimensions. Examples of the first several k -simplices are illustrated in Fig. 5.13 [40].

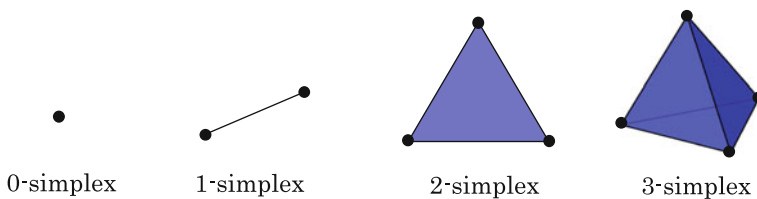


Fig. 5.13 Examples of a 0-simplex, a 1-simplex, a 2-simplex, and a 3-simplex

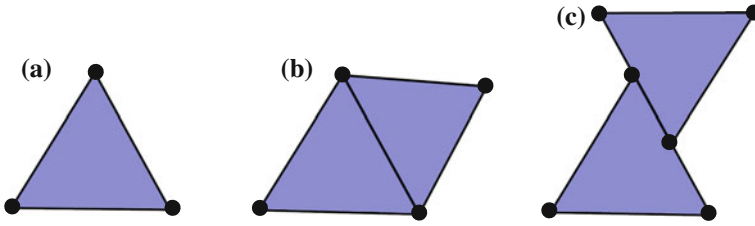


Fig. 5.14 Examples of simplicial complexes (a, b) and a non-simplicial complex (c)

This concept of k -simplices may also be extended to include collections of attached simplices, which are known as *simplicial complexes*. A simplicial complex K is a finite collection of simplices that satisfies the following two conditions:

1. For any simplex $\sigma_i \in K$, each face of σ_i is also in K ;
2. Two k -simplices $\sigma_1, \sigma_2 \in K$ must intersect at a common face.

While (a)–(c) in Fig. 5.14 are collections of k -simplices; it is clear from the definition that (c) is not a simplicial complex because it violates condition 2.

It is evident from the above discussion that there is a strong similarity between graphs and simplicial complexes. In fact, a simplicial complex is in a sense just the generalization of a graph to higher dimensions. If we define the r -skeleton of the simplicial complex K as the collection of all k -simplices of K for which $k \leq r$, then it is clear that the 1-skeleton of K is a graph, also called the *underlying graph* of K , since it is simply a set of vertices and edges.

5.5.1 From WSNs to Simplicial Complexes

We introduce here two important simplicial complexes and their respective planar counterparts and show how each may be used to model a WSN [40]. The sensor network is mapped to one of these simplicial complexes which correspond to sensing coverage. Then, the simplicial complex can be analyzed to find the number and location of holes, which relate to holes in the actual sensor network. Such simplicial complexes can model the coverage of the sensor networks and related quality of service in terms of coverage completeness.

Čech Complex The Čech complex, also known as the nerve complex [4], can be defined for a set of points X with a parameter $r > 0$. The points x_0, x_1, \dots, x_n are defined as 0-simplices. The 1-simplex $[x_0x_1]$ is defined if the r -balls centered at x_0 and x_1 intersect. If the r -balls centered at x_0, x_1 , and x_2 have a common intersection, then the 2-simplex $[x_0x_1x_2]$ exists in the Čech complex. In general, the k -simplex $[x_0x_1x_2 \dots x_k]$ exists if and only if the r -balls centered at x_0, x_1, \dots, x_k have a nonempty common intersection.

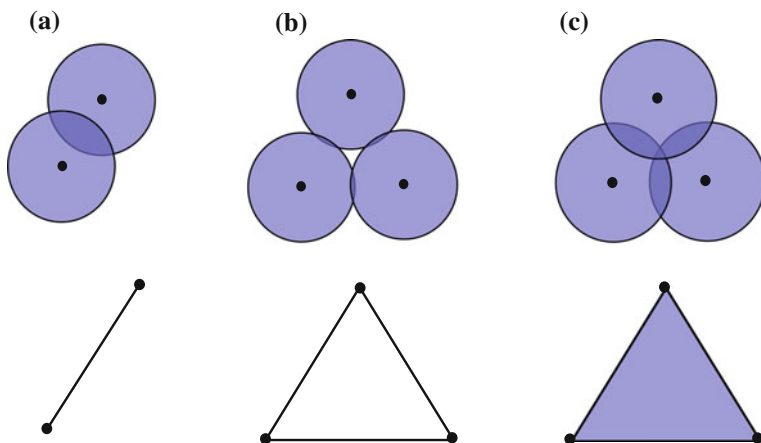


Fig. 5.15 Constructing the Čech complex from the union of sensing disks

It is obvious that this construction fits well for modeling coverage in a WSN. To observe this, one only needs to consider that r from the above construction equals the sensing radius r_s . For example, when three sensing disks have a common intersection we add a 2-simplex to the Čech complex which signifies that the entire area inside the polygon (i.e., triangle) formed by these three sensor nodes is covered. The inclusion of higher dimensional simplices indicates higher degrees of coverage. Figure 5.15 illustrates the construction of simplices from the union of coverage disks. In (a), a 1-simplex is constructed since the disks have a common intersection. The disks in (b) do not have a common intersection and the corresponding triangle is not filled, while the disks in (c) do have a common intersection and the corresponding filled triangle is constructed (2-simplex) [40].

An important benefit of this simplicial complex is that it can be computed even if the nodes have different sensing ranges, as long as the sensing model is a circle around the node. For theoretical analysis purposes it is usually assumed that all nodes have identical sensing ranges, but do note that this assumption is, in general, not required. However, a potential disadvantage of this simplicial complex construction is that precise distances between nodes must be known. If these distances are not precisely known this could lead to errors in the constructed Čech complex.

Rips Complex [40] The Rips complex, sometimes also called the Vietoris complex, was originally introduced by Vietoris in 1927 [49]. Given the point set X with a diameter r , the Rips complex is defined as follows. The k -simplex $[x_0x_1x_2 \dots x_k]$ is defined in the Rips complex if the pairwise distance between each of $x_0, x_1, x_2, \dots, x_k$ is less than r . For example, when the distance between two points x_0 and x_1 is less than r , then we define the 1-simplex $[x_0x_1]$. Likewise, if the pairwise distance between the three points at x_0, x_1 , and x_2 are all less than r , then we define the 2-simplex $[x_0x_1x_2]$.

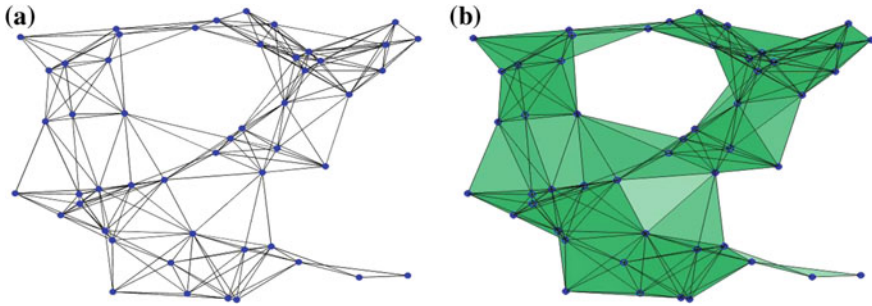


Fig. 5.16 Communication graph (a) and its induced Rips complex (b)

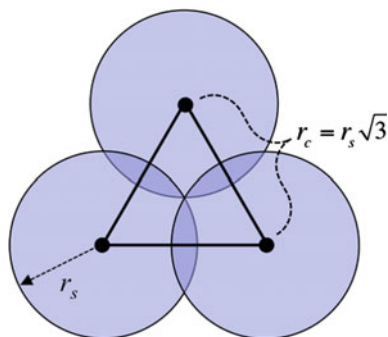
Note that the definition of the Rips complex is very similar to that of the unit disk graph from Sect. 5.1.1. It is apparent from the two definitions that using parameter r for each, the 1-skeleton of the Rips complex is equivalent to the unit disk graph. The Rips complex can be considered as a sort of “higher dimensional” version of the unit disk graph.

This similarity in the Rips complex and the unit disk graph leads to an alternate approach to constructing the Rips complex which is extremely useful for sensor networks and was exploited in [45] to construct the Rips complex in a coordinate-free network. Given the communication graph of the network, which by definition is a unit disk graph, the graph can be expanded to include higher dimensional simplices as follows. Any complete graph on k vertices, that is, the graph in which all k vertices are pairwise adjacent, corresponds to a $(k - 1)$ -simplex in the Rips complex. Thus, any triangles (K_3) in the communication graph are “filled in” to become 2-simplices, any tetrahedrons (K_4) are “filled in” to become 3-simplices, and so on. Since no locations are needed to create the communication graph the Rips complex for a sensor network can be constructed entirely from local connectivity data. Figure 5.16 shows an example communication graph and its induced Rips complex.

5.5.2 Comparison of Čech Complex and Rips Complex

While the Čech complex maps the coverage provided by the union of sensing coverage disks into a corresponding simplicial complex, this relationship is not as well defined for the Rips complex whose construction is based solely on the communication range of the nodes and does not take into account their sensing range at all. In [45], the ratio $r_c \leq \sqrt{3} \cdot r_s$ is used since for a set of three nodes with maximum pairwise distance for communication range of r_c , the sensing radius must be at least $r_c/\sqrt{3}$ for the sensing disks to have common intersection as shown in Fig. 5.17 [40].

Fig. 5.17 Relationship between r_c and r_s for the Rips complex



It can also be shown that the radius- r Rips complex has the same 1-skeleton as the radius- $r/2$ Čech complex. Thus, assuming $r_s = r_c/2$, then the Rips complex and the Čech complex are equivalent with the possible exception with three communicating nodes without a common intersection of their sensing disks. In other words, triangle-shaped holes, such as that in Fig. 5.15b, are not possible in the Rips complex since based on the above assumption the nodes are all within communication range, and they would immediately be “filled in” as 2-simplices.

Figure 5.18 shows examples of Rips and Čech complexes created using the above assumption. Notice that the only difference is that the triangle-shaped holes in the Čech complex do not exist in the Rips complex as mentioned above. This could be acceptable in some applications since the triangular holes represent areas with very small coverage holes, and larger holes will exist in both complexes. The drawback to using this assumption, however, is that if this ratio is not exact, then this equivalence between two simplicial complexes will not hold.

Even with the help of either of the above assumptions, the Čech complex and Rips complex are still not equivalent. For this reason, results in [45] instead show how to infer Čech data by “squeezing” it between two Rips complexes constructed at different radii. This idea may be useful for verifying coverage, but it is not as

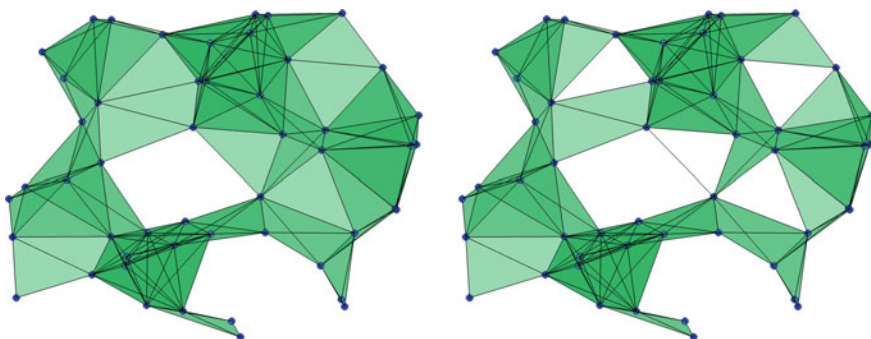


Fig. 5.18 Rips complex (*left*) and Čech complex (*right*) for $r_s = r_c/2$

helpful in localization of the holes. If the sensor nodes' coordinates are known, then the Čech complex may be used to capture the exact coverage of the sensing disks. Otherwise, if sensor nodes' coordinates are not known, then for any ratio between sensing and communication radii the Rips complex will only approximate the Čech complex.

5.5.3 Subcomplexes with Planar Topology

We introduce subcomplexes of both the Čech complex and the Rips complex [40]. The significance of these subcomplexes is that their underlying graphs are planar, meaning that they have no crossing edges. As a result, both simplicial complexes are 2-complexes, which mean that the largest k -simplex in the complex is a 2-simplex and k -simplices of higher dimensions cannot exist in the complex. These subcomplexes can be considered as the maximal subcomplexes that retain the number and shape of the holes in the original complex.

Alpha-Shape Complex [40] Closely related to the Čech complex is the Alpha-shape complex, which is originally due to Herbert Edelsbrunner [11]. To construct this complex, one must first compute the Voronoi diagram Vor_X for the point set X . Next, for each point $x \in X$ its Alpha-cell $A(x, r)$ can be defined as the intersection $\text{Vor}_X(x) \cap B(x, r)$, where $\text{Vor}_X(x)$ is the Voronoi cell of x and $B(x, r)$ is an open ball of radius r centered at x . The Alpha-shape complex $C_\alpha(X, r)$ is defined by the nerve, or Čech complex of the set of Alpha-cells $A(x, r)$. That is, a k -simplex in the Alpha-shape complex corresponds to the non-empty intersection of $k + 1$ alpha-cells.

Figure 5.19 shows the Voronoi diagram and sensing disks for a set of points. The collection of Alpha-cells, which is the intersection of each sensing disk and its Voronoi cell, is shown in Fig. 5.20. Finally, the nerve of these alpha-cells is taken to create the alpha-shape complex shown in Fig. 5.20. Notice that the two holes in the alpha-shape complex clearly correspond to the holes in the union of sensing disks from Fig. 5.19.

It can be shown that for a given parameter r and a set of points X there is a homotopy equivalence between the two complexes, i.e., the thickness can be ignored (i.e., higher dimensional k -simplices), and the holes stayed preserved in the complex as shown in Fig. 5.21. Also, as may be expected with the use of the Voronoi diagram, the Alpha-shape complex is a subcomplex of the Delaunay complex and so its underlying graph must be planar.

Maximal Simplicial Complex [40] Note that it is also highly desirable to have a planar subcomplex of the Rips complex, analogous to the relationship between the Čech and Alpha-shape complexes. The basic goal is to remove a subset of the edges in the communication graph such that the new graph is planar and the number and

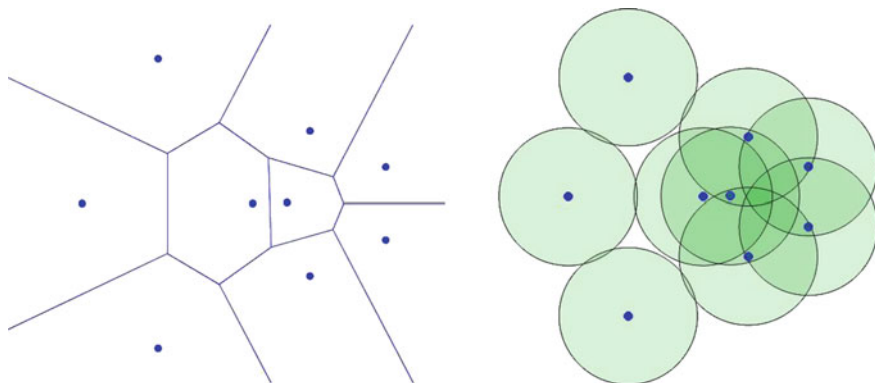


Fig. 5.19 Voronoi diagram (*left*) and union of sensing disks (*right*) for a set of points

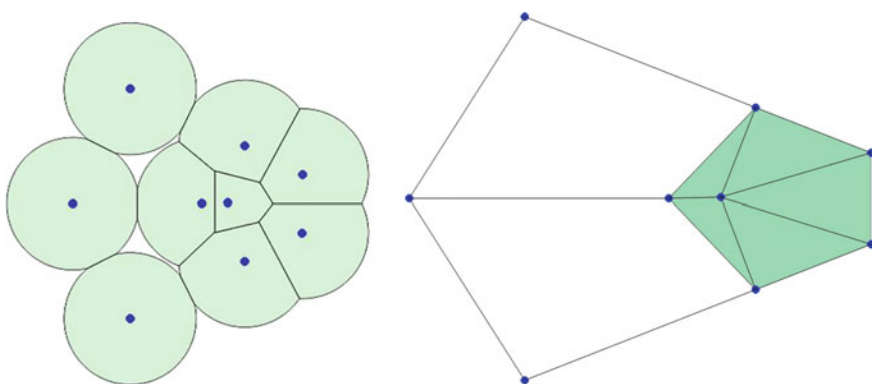


Fig. 5.20 Alpha cells (*left*) and alpha shape complex (*right*) for point set from Fig. 5.19

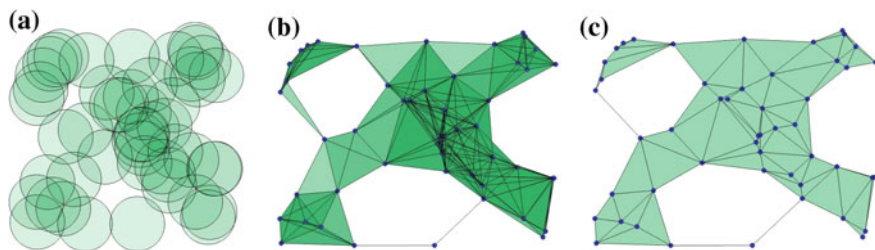


Fig. 5.21 Union of sensing disks (a), Čech complex (b), and alpha shape complex (c) for a set of sensor nodes

shape of holes is maintained in the induced Rips complex. In general, this is a difficult problem since the embedding is not known and removal of the wrong edges will have an effect on the holes in the induced Rips complex.

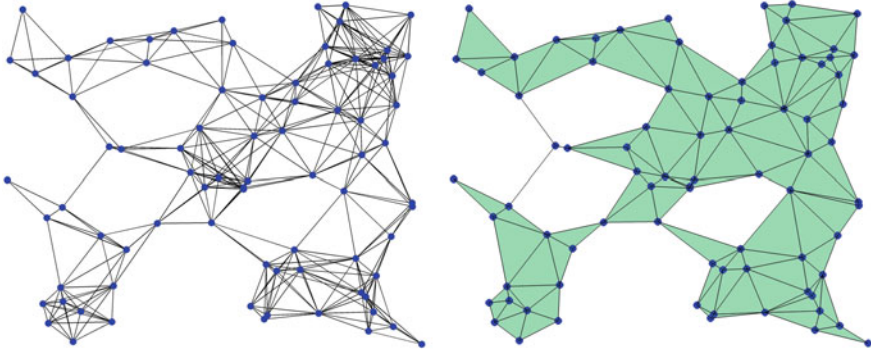


Fig. 5.22 Communication graph (*left*) and its maximal simplicial complex (*right*)

Researchers in [35] have proposed a solution to this problem by examining all crossing edges in the communication graph and carefully eliminating one edge in the crossing pair while still maintaining the structure of the topological holes. Authors refer to the planar subgraph as a *maximal simplicial subgraph*. After obtaining this subgraph, all that is left is to “fill in” all the triangles to obtain the induced Rips complex, which they call the *maximal simplicial complex*.

Figure 5.22 shows an example communication graph and its maximal simplicial complex obtained from the communication graph. The relative position of holes is maintained in the maximal simplicial complex.

5.6 Simplicial Homology and Coverage Holes

Homology is a mathematical method for detecting and counting holes in a topological space using algebra. A basic background on homology theory that is relevant to the sensor networks coverage is presented here, while a more complete development of homology theory can be found in [24, 36].

One way to define homology is as a vector space. For a simplicial complex K , the 1-dimensional homology $H_1(K)$ can be used to determine its holes. This can be represented as a vector space whose basis represents cycles in K surrounding its holes. Thus, the dimension of this vector space $|H_1(K)|$, often called the first Betti number of K , gives the number of cycles that corresponds to the number of holes. Holes of higher dimension may also be found using homology. For example, the second Betti number for a sphere S , which is the dimension of $H_2(S)$, is 1 because the sphere is empty inside. For the sensor network coverage problems, one-dimensional holes correspond to holes in sensing coverage.

Computing the first Betti number of a simplicial complex is important in the analysis of coverage completeness since it provides information about a number of

holes that exist in the network. Two software packages for computational homology are PLEX [46] and CHomP [8]. The typical Betti number implementations involve computing the rank of matrices, which for very large networks, can be prohibitively slow. However, this method does not take into account that the simplicial complexes are not arbitrary but are constrained by the physical nature of the sensor network. A simple arithmetic formula for computing the first Betti number of a simplicial complex K is given by:

$$b_1 = b_0 - \sum_{k=0}^n (-1)^k |(k\text{-simplices in } K)|, \quad (5.25)$$

where n corresponds to the largest k -simplex in K , and b_0 is the zeroth Betti number, which is equivalent to the number of connected components in the underlying graph, or 1-skeleton, of K . This follows from the Euler characteristic χ for a simplicial complex which can be defined as the alternating sum of Betti numbers

$$\chi = \sum_{k=0}^{\infty} (-1)^k b_k. \quad (5.26)$$

For an arbitrary simplicial complex, Betti numbers of any dimension may be nonzero. However, all the simplicial complexes that relate to sensor networks have constraints due to the disk-shaped sensing and communication ranges which have been assumed. Therefore, all Betti numbers of dimension larger than one will be zero if the nodes lie in the plane. This leads to the simplified formula

$$\chi = b_0 - b_1, \quad (5.27)$$

where b_0 is equivalent to the number of connected components in the underlying graph. A second definition of the Euler characteristic for a simplicial complex K using the alternating sum of the number of k -simplices is given by

$$\chi = \sum_{k=0}^n (-1)^k |(k\text{-simplices in } K)|, \quad (5.28)$$

where n corresponds to the largest k -simplex in K . From here follows the Eq. (5.25).

The drawback of (5.25) in computing coverage holes is that one must know all higher dimensional simplices, whereas with the matrix intensive computations, only the $(k + 1)$ -simplices and lower dimensions are needed to find the k -th Betti number. Thus, only the sets of 0-, 1-, and 2-simplices would be needed to compute b_1 .

However, for the planar subcomplexes described above, the following simplified result can be used:

$$b_1 = b_0 - |V(G)| + |E(G)| - |(2\text{-simplices in } K)|, \quad (5.29)$$

where G is the underlying graph of the simplicial complex K . Under assumption that the 1-skeleton of K is a connected graph, which is typically the case, then $b_0 = 1$ and so the computation can be simplified even further to

$$b_1 = 1 - |V(G)| + |E(G)| - |(2\text{-simplices in } K)|. \quad (5.30)$$

Note that this formula can also be derived from Euler's formula for connected planar graphs. Euler's formula for a connected planar graph can be written as $V - E + F = 2$, where V is the number of vertices, E is the number of edges, and F is the number of faces. This formula, however, accounts for an additional face known as the infinite, or exterior face, which is not needed for sensor networks applications and can be discarded. Also, it is clear that the number of faces F in the planar underlying graph of one of the planar subcomplexes is the sum of the number of 2-simplices in the complex and the number of holes. This is due to the obvious fact that every face in the underlying graph is either a 2-simplex or a hole.

Example 5.3 Consider the simple complex illustrated in Fig. 5.23. The simplicial complex has seven 0-simplices (or vertices), nine 1-simplices (or edges), and two 2-simplices. Inserting these values into Eq. (5.30), we see that $b_1 = 1 - 7 + 9 - 2 = 1$, as expected by observation of the simplicial complex (Fig. 5.23).

In applications related to sensor networks hole coverage, the alpha-shape complex and the maximal simplicial complex have advantage over other complexes for multiple reasons. First, algorithms for computing the number of holes in the complex are computationally efficient by utilizing simple arithmetic calculations. Second, divide-and-conquer algorithms are often suitable for planar graphs. Furthermore, it allows for coverage holes to be defined as faces in the underlying planar graph. If the underlying graph is not planar, holes may not be uniquely identifiable as is shown in Fig. 5.24. Notice that the hole is not uniquely defined since one could argue that cycle A, B, D, cycle A, B, C, or cycle A, B, C, D are all the correct way to identify the hole.

Fig. 5.23 Example simplicial complex with $b_1 = 1$

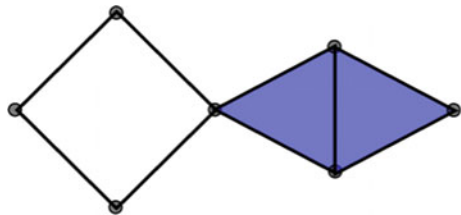
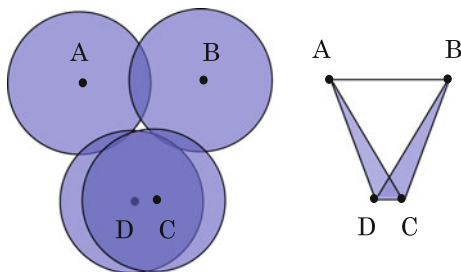


Fig. 5.24 Sensing disks (*left*) and corresponding simplicial complex (*right*) which contains a hole that is not uniquely identifiable



5.7 K -Coverage

Coverage in WSNs applications is usually associated with 1-coverage where it is desired to have *at least one* sensor node covering every point of interest. If 1-coverage is not satisfied, then there is a hole in the sensor network coverage. The term *hole* here also refers to a 1-hole, meaning there is not at least one sensor node covering a certain area in the set of interest. A more robust network to node failures and other node related errors might have k -coverage requirements, i.e., that every point in the area of interest should be covered by at least k sensor nodes.

Definition 5.7 Consider a sensor network consisting of a set Z sensor nodes and a region on interest S . A subset $P \subset S$ is said to be k -covered if and only if for every point $p \in P$, there are at least k sensor nodes from Z that cover the point p .

Looking at the construction of the Čech complex it is evident that only a single point must be covered by $k + 1$ disks to create a k -simplex (e.g., see Fig. 5.15c). It is required to show that the entire area inside a polygon formed by k vertices is k -covered. This implies that for any single polygon vertex the other $k - 1$ vertices must all lie within its sensing disk and leads us back to the Rips complex, which admits a k -simplex if the pairwise distance between any set of k vertices is less than parameter r . Thus, a construction of the Rips complex using the parameter r_s allows one to determine k -coverage. For example, any 3-simplex in this simplicial complex comprised four vertices and, of course, the area inside the polygon formed by these vertices is 4-covered. Note that this method gives a sufficient (but not necessary) condition for k -coverage since there may still be other k -covered portions of the network which do not entirely fill the area inside a polygon formed by a subset of nodes. However, this is much like the case for 1-coverage where a hole in the simplicial complex looks large, but in reality, is much smaller when considering the actual union of sensing disks (e.g., see Fig. 5.15b).

5.8 Coverage Control

The problem of deploying sensors to satisfy application requirements is called coverage control problem [5]. The coverage control can be static, or offline, and dynamic, or online. In the static coverage, the sensor node locations are pre-calculated before the deployment, such that once deployed, the sensor network satisfies pre-assigned coverage mission requirements. This is equivalent to the facility location optimization problem in operations research. The dynamic coverage control includes mobile sensor networks, Fig. 5.25, where the coverage control is calculated online and the position of sensor nodes is adjusted according to specified criteria. Furthermore, the dynamic coverage control can be centralized with a central controller (or a base station) providing control inputs to the whole network, or distributed where each sensor node adjusts its position according to the locally executed algorithm.

Another example of dynamic coverage control is a future WSN usage in weather monitoring and hurricane tracking. Even though hurricanes can be tracked by satellites, precise wind speed, precipitation, and pressure of the storm-affected area can be monitored and detected only using close-range sensing devices. Presently such monitoring and data gathering is done using NOAA reconnaissance airplane with probes that record air pressure, humidity, temperature, and wind speed that weather scientists use to predict storm surge, place and time when the storm hits the land.

Future hurricane monitoring systems will consist of a network of ground sensors and mobile sensors with coverage that can dynamically adapt to changes in the hurricane path and strength; see Fig. 5.26.

Several optimal control problems related to sensor networks were formulated in [5]. An optimal coverage under constraints of imprecise detection and terrain properties where the number of sensor nodes is minimized was presented in [9].

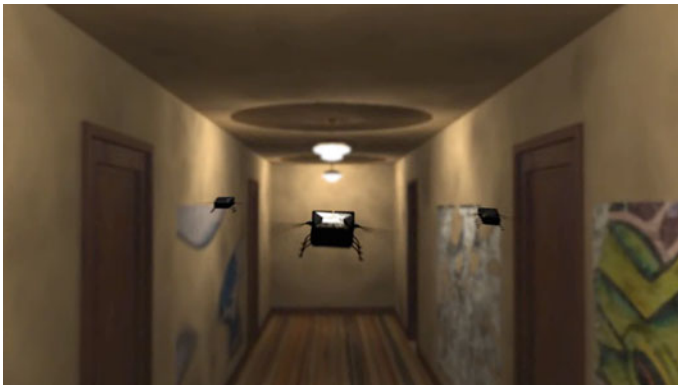


Fig. 5.25 Micro-aerial vehicle based sensor network—a mobile sensor networks with dynamic coverage control

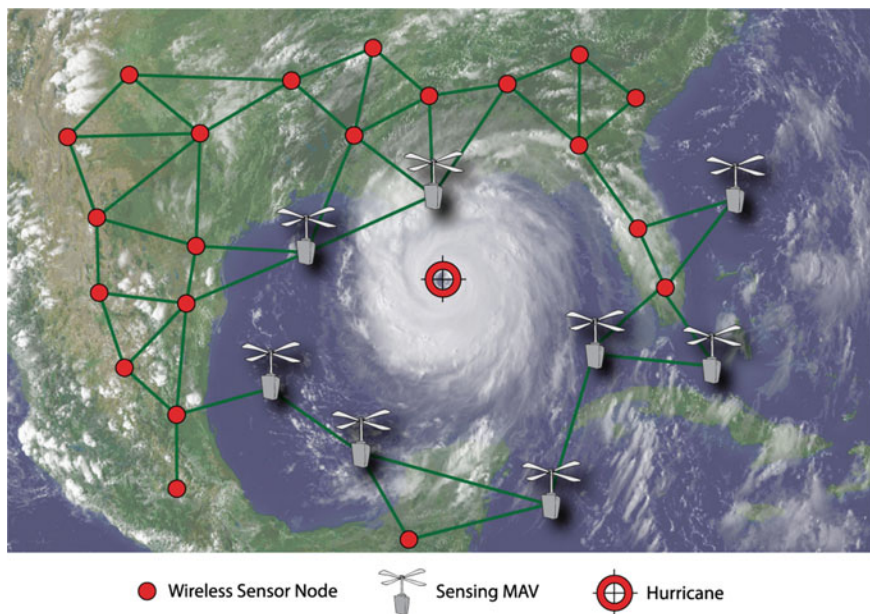


Fig. 5.26 Static and mobile sensor nodes track a hurricane

In [17] problems arising in maintaining coordination and communication between the group of robots and solutions to these problems were discussed. Three models of deployment are introduced to maximize the coverage area within the close range of the mobile nodes, deployment to maximize the probability of detecting a source, and deployment to maximize the visibility of the network. A variety of control methods in multi-vehicle cooperative control using graph theory have been presented in [41]. Optimal coverage control for mobile sensor networks was presented in [10]. The paper uses a Voronoi partitioning and Lloyd descent algorithm but does not consider network connectivity constraints. Two location functions that characterize coverage performance were provided in [9] including a study of their gradient properties via nonsmooth analysis. In most cases, the feasibility sets are assumed to be convex and related optimization problems are convex optimization problems. If network connectivity is considered as well, then underlying optimization problems become, in most cases, non-convex optimization problems.

Example 5.4 Consider a coverage control problem in chemical plants where it is required to provide an optimal coverage of large chemical plants or areas of interest with a large number of static sensor nodes that monitor for a wide spectrum of chemical agents; have several mobile nodes moving over rough plant terrain and track a possible contamination cloud; and allow technicians to adjust the controller of all mobile sensor nodes. This is an optimal coverage control problem with a trade-off between uniform coverage of the whole plant and focused coverage of the contamination cloud; see Fig. 5.27.

Let r be the sensors radio transmission range. Assume that there is a focus point $(X_F(t), Y_F(t))$ where several mobile sensor nodes should converge. In the case of a chemical contamination example, the focus point can be a contamination cloud (center of mass of the cloud). Consider a region of interest S that is a compact set, and a set Z of sensor nodes. A subset $M \subset Z$ is a set of all mobile sensor nodes. An optimal coverage control problem can be formulated by specifying a cost function of interest and constraints that limit the network in terms of geometry, flow, energy, or any other network parameter.

The coverage control problem can be formulated as an optimal control problem, or more precisely, as a linear-quadratic regulator (LQR) problem. The problem is to find an optimal location of mobile nodes M , such that the following cost function is minimized:

$$\min J_1 = R \sum_{i \in M} \text{dist}^2[(x_i, y_i), (X_F(t), Y_F(t))] + Q \sum_{i \in M, j \in Z} \text{dist}^{-2}[(x_i, y_i), (x_j, y_j)], \quad (5.31)$$

where R and Q are the control design parameters or weighting factors, and $\text{dist}[(x_i, y_i), (x_j, y_j)]$ is an Euclidean distance between nodes i and j and is given by

$$\text{dist}[(x_i, y_i), (x_j, y_j)] = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}. \quad (5.32)$$

Note that this is a convex optimization problem since the optimization cost function is convex and there are no constraints. Convex optimization refers to cost function being a convex function and constraint set being a convex set; otherwise, optimization is a non-convex and common operation research methods do not apply. For extra reading on non-convex optimization, we refer to [27]. The parameter R penalizes closeness to the focus point, or tracking of the target. The parameter Q penalizes uniform distribution of nodes across the set S . For example,

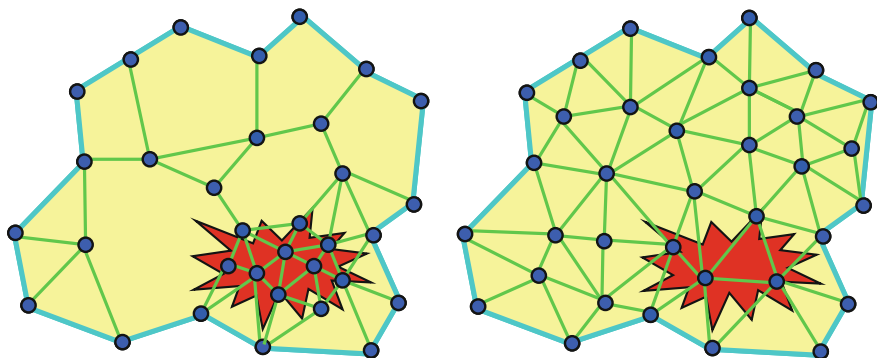


Fig. 5.27 Focused coverage (*left*) versus uniform coverage (*right*)

$R = 10, Q = 10,000$ means that the user is much more concerned with the uniform node distribution than to cover the specific area of interest, Fig. 5.27 (right). On the other hand, $R = 10,000, Q = 10$ indicates that the user wants extensive coverage of the focus point (tracking) rather than uniform node distribution, Fig. 5.27 (left).

The cost function in (5.31) has such specific form because the original problem is a multi-objective optimization problem. One common way to solve this type of problem is to combine the weighted sums of each individual objective function into an aggregate objective function (AOF). Specifically, the two objectives are to minimize the distance between the mobile nodes and the focus point and to maximize the distance between the mobile nodes and the other static and mobile nodes (minimize the inverse of this distance).

The problem formulation, including the cost function that is given in (5.31) has more of a theoretical significance, since it does not consider the network connectivity. Namely, such a solution will optimize the cost function and distribute the nodes accordingly without consideration of the network connectivity. Some sensor nodes can go out of communication range and become useless from the network standpoint. More realistic problem formulation that includes network connectivity can be given as follows.

Given the compact coverage area of interest S , a focus point $(X_F(t), Y_F(t))$, sensor network graph $G(t) = (N(t), E(t))$ and mobile subgraph $G_m(t) = (M(t), E_m(t))$, find an optimal vertex location of the mobile subgraph $G_m(t)$ such that the following cost function is minimized

$$\min J_2 = R \sum_{i \in M} \text{dist}^2[(x_i, y_i), (X_F(t), Y_F(t))] + Q \sum_{i \in M, j \in N} \text{dist}^{-2}[(x_i, y_i), (x_j, y_j)], \quad (5.33)$$

such that the graph $G(t)$ stays vertex k -connected and satisfies the mobile node localization condition. Constants R and Q are control design parameters.

Note that this is a non-convex optimization problem where a feasibility set is, in general, a non-convex set. Similarly, one can formulate suboptimal sensor network coverage control problem with focus/target areas possibly represented by several disjoint sets. Such constraint set is, in general, non-convex.

An optimal deployment that is based on probabilistic models is given in [59]. The sensor locations (x, y) have Gaussian distribution given by its mean and standard deviation, as well as its joint distribution $p_{xy}(x, y)$. The probability for a gridpoint (i, j) to be detected by a sensor at (x, y) is given by $c_{ij}(x, y)$ and the miss probability of the gridpoint is $m_{ij}(x, y) = 1 - c_{ij}(x, y)$. For a set N of deployed static sensors, the total miss probability of the gridpoint (i, j) is given by

$$m_{ij} = \prod_{(x, y) \in N} (1 - c_{ij}(x, y)). \quad (5.34)$$

Assuming that the newly deployed sensor will be placed at (x, y) , the total miss probability is given by

$$m(x, y) = \sum_{(i,j) \in \text{Grid}} m_{ij}(x, y) m_{ij}. \quad (5.35)$$

Based on the calculated total miss probability, one can place the sensor to minimize the miss probability, providing the best possible sensor network coverage from the probability of detection point of view under given assumptions of grid deployment. After every node deployment, above probabilities are updated and recalculated based on new topology. The computational complexity of miss probability algorithm is $O(n^2 m^2)$ for a grid size of $n \times m$.

Questions and Exercises

1. What is the degree of a vertex in a graph G ? Provide an example.
2. Explain when is a graph connected? Draw examples of connected and disconnected graphs.
3. Demonstrate by example that if a graph G is disconnected, then its complement \bar{G} is connected. Show another example that the opposite is not true—if the graph G is connected, then its complement \bar{G} is not necessarily disconnected. Finally, prove this statement rigorously.
4. Study the Chvátal's art gallery theorem and find out what is an upper bound on the minimal number of guards that are required to cover the gallery? Please describe "upper bound on the minimal number" in case of this problem.
5. If the gallery floor plan can be divided into square rooms, what would be the upper bound on the minimal number of guards required to cover the gallery in this case?
6. When one can say that complete sensing coverage of a convex region implies connectivity of the network?
7. Given a graph $V = \{1, 2, 3, 4, 5, 6, 7\}$, $E = \{(1, 2), (1, 3), (1, 5), (2, 3), (3, 5), (4, 7), (6, 7)\}$, using a graph Laplacian check if the graph is connected. Find the number of spanning trees using Laplacian eigenvalues.
8. Given a graph $V = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$, $E = \{(1, 2), (1, 3), (1, 5), (2, 3), (3, 5), (3, 6), (4, 5), (5, 8), (5, 9), (6, 7), (6, 8)\}$, estimate the lower and upper bounds on the graph diameter.
9. For a graph with vertices $V = \{1, 2, 3, 4, 5\}$, draw a complex and a simplicial complex.
10. For a set of points in two-dimensional space with the following coordinates $X = \{(-1, -2), (1, 0), (2, 3), (3, -4), (-3, -5)\}$, draw a Voronoi diagram and derive alpha shape complex.

11. Describe a k -coverage in sensor networks and its relationship with the Rips complex. What does it mean by 2-coverage specifically and in terms of the Rips complex?
12. Use linear quadratic control problem setup similar to the one given in Eq. (5.31) and formulate a sensor network coverage control problem where it is required for mobile nodes to be attracted to the target node, to be repelled from the static sensor nodes, and to be repelled from each other (mobile nodes).
13. How can you extend the previous problem formulation and include constraints that would ensure collision avoidance between mobile sensor nodes?

References

1. S. Adlakha and M. Srivastava, "Critical density thresholds for coverage in wireless sensor networks," *IEEE Wireless Communications and Networking*, vol. 3, pp. 1615–1620, March 2003.
2. A. Agah, S.K. Das, K. Basu, and M. Asadi, "Intrusion detection in sensor networks: a non-competitive game approach," *Third IEEE International Symposium on Network Computing and Applications*, pp. 343–346, 2004.
3. A. Arora, R. Ramnath, E. Ertin, P. Sinha, S. Bapat, V. Naik, V. Kulathumani, H. Zhang, H. Cao, M. Sridharan, S. Kumar, N. Seddon, C. Anderson, T. Hermon, N. Trivedi, C. Zhang, M. Nesterenko, R. Shah, S. Kulkarni, M. Aramugam, L. Wang, M. Gouda, Y. Choi, D. Culler, P. Dutta, C. Sharp, G. Tolle, M. Grimmer, B. Ferreira, and K. Parker, "ExScal: elements of an extreme scale wireless sensor network," *Proc. Of the 11th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications*, pp. 102–108, August 2005.
4. G. Carlsson and V. de Silva, "Topological approximation using small simplicial complexes," (Preprint), 2003.
5. C. G. Cassandras and W. Li, "Sensor networks and cooperative control," *European Journal of Control*, vol. 11, pp. 436–463, 2005.
6. S. Chaudhry, V. Hung, and R. Guha, "Optimal placement of wireless sensor nodes with fault tolerance and minimal energy consumption," *IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, pp. 610–615, October 2006.
7. V. Chvátal, "A combinatorial theorem on plane geometry," *Journal of Combinatorial Theory, Series B*, vol. 18, pp. 39–41, 1975.
8. "CHomP: computational homology project," <http://chomp.rutgers.edu>.
9. J. Cortes and F. Bullo, "Coordination and geometric optimization via distributed dynamical systems," *SIAM Journal on Control and Optimization*, vol. 44, no. 5, 2005, pp. 1543–1574.
10. J. Cortes, S. Martinez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, 2004, pp. 243–255.
11. H. Edelsbrunner, "The union of balls and its dual shape," *Discrete & Computational Geometry*, vol. 13, pp. 415–440, 1995.
12. M. Fiedler, Algebraic connectivity of graphs, *Czech. Math. J.*, vol. 23, no. 98, 1973, pp. 298–305.
13. S. Funke and C. Klein, "Hole detection or: how much geometry hides in connectivity?," *SCG '06: Proceedings of the Twenty-Second Annual Symposium on Computational Geometry*, pp. 377–385, New York, NY, 2006.

14. S. Funke and N. Milosavljevic, "Network sketching or: how much geometry hides in connectivity? – Part II," *Proc. of the 18th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2007.
15. S. Funke, "Topological hole detection in wireless sensor networks and its applications," *DIALM-POMC: Joint Workshop on Foundations of Mobile Computing*, 2005.
16. D.W. Gage, "Command control for many-robot systems," *Proc. of Nineteenth Annual AUVS Technical Symposium*, pp. 22–24, 1992.
17. A. Ganguli, S. Susca, S. Martinez, F. Bullo, and J. Cortes, "On collective motion in sensor networks: sample problems and distributed algorithms," *Proc. of the 44th IEEE Conference on Decision and Control, and the European Control Conference*, 2005, pp. 4239–4244.
18. A. Ghosh and S. Das, "Coverage and connectivity issues in wireless sensor networks," in *Mobile, Wireless, and Sensor Networks: Technology, Applications, and Future Directions*, Ed. R. Shorey, A.L. Ananda, M.C. Chan, and W.T. Ooi, John Wiley & Sons, Inc., 2006.
19. R. Ghrist and A. Muhammad, "Coverage and hole-detection in sensor networks via homology," *Proc. ISPN*, 2005.
20. A. Gusrialdi, T. Hatanaka, and M. Fujita, "Coverage control for mobile networks with limited-range anisotropic sensors," *Proc. 47th IEEE Conference on Decision and Control*, Cancun, Mexico, Dec. 2009.
21. M. Hefeeda and H. Ahmadi, "A probabilistic coverage protocol for wireless sensor networks," *Proc. of IEEE International Conference on Network Protocols (ICNP '07)*, pp. 41–50, Beijing, China, October 2007.
22. N. Heo and P.K. Varshney, "A distributed self-spreading algorithm for mobile wireless sensor networks," *Proc. IEEE Wireless Communications and Networking Conference, (WCNC'03)*, New Orleans, LA, March 2003, pp. 1597–1602.
23. A. Howard, M.J. Matraric, and G.S. Sukhatme, "Mobile sensor network deployment using potential fields: a distributed, scalable solution to the area coverage problem," *International Symposium on Distributed Autonomous Robotics Systems*, June 2002.
24. T. Kaczynski, K. Mischaikow, and M. Mrozek, *Computational Homology, Applied Mathematical Sciences 157*, Springer-Verlag, 2004.
25. J. Kanno, J. G. Buchart III, R. R. Selmic and V. Phoah, "Detecting coverage holes in wireless sensor networks," *Proc. 17th Mediterranean Conference on Control and Automation*, Thessaloniki, Greece, June 24–26, 2009.
26. K. Kar and S. Banerjee, "Node placement for connected coverage in sensor networks", *Proc. of WiOpt 2003: Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks*, 2003.
27. D. Klatté and B. Kummer, *Nonsmooth Equations in Optimization*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 2002.
28. A.A. Kooshesh and B.M.E. Moret, "Three-coloring the vertices of a triangulated simple polygon," *Pattern Recognition*, vol. 25, pp. 443–444, 1992.
29. S. Kumar, T. H. Lai, and A. Arora, "Barrier coverage with wireless sensor networks," *Proc. of the 11th Annual International Conference on Mobile Computing and Networking*, Cologne, Germany, pp. 284–298, 2005.
30. A.K. Lédeczi, A. Nádas, P. Völgyi, G. Balogh, B. Kusy, J. Sallai, G. Pap, S. Dóra, and K. Molnár, "Countersniper systems for urban warfare," *ACM Transactions on Sensor Networks*, vol. 1, no. 2, pp. 153–177, Nov. 2005.
31. D.T. Lee and A.K. Lin, "Computational complexity of art gallery problems," *IEEE Transactions on Information Theory*, vol. 32, no. 2, pp. 276–282, March 1986.
32. M. Marengoni, B.A. Draper, A. Hanson, and R.A. Sitaraman, "System to place observers on a polyhedral terrain in polynomial time," *Image and Vision Computing*, vol. 18, pp. 773–780, 1996.
33. S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. Srivastava, "Coverage problems in wireless ad-hoc sensor networks," *IEEE Infocom 2001*, pp. 1380–1387, April 2001.
34. B. Mohar, "The Laplacian spectrum of graphs," in *Graph Theory, Combinatorics, and Applications*, vol. 2, Ed. Y. Alavi, G. Chartrand, O. R. Oellermann, A. J. Schwenk, Wiley, 1991, pp. 871–898.

35. A. Muhammad and M. Egerstedt, "Connectivity graphs as models of local interactions," *Proc. of the 43rd IEEE Conference on Decision and Control*, pp. 124–129, December 2004.
36. J.R. Munkres, *Elements of Algebraic Topology*, Addison Wesley, 1993.
37. J. O'Rourke, *Art Gallery Theorems and Algorithms*, Oxford University Press, 1987.
38. R. Pandit and P.M. Ferreira, "Determination of minimum number of sensors and their locations for an automated facility: an algorithmic approach," *European Journal of Operational Research*, vol. 63, no. 2, pp. 231–239, 1992.
39. S. Poduri and G.S. Sukhatme, "Constrained coverage in mobile sensor networks," *Proc. IEEE Int. Conf. Robotics and Automation (ICRA '04)*, New Orleans, LA, April 2004, pp. 40–50.
40. S. Ramazani, J. Kanno, R. Selmic, and M. Brust, "Topological and combinatorial coverage hole detection in coordinate-free wireless sensor networks," *International Journal of Sensor Networks*, vol. 21, no. 1, pp. 40–52, 2016.
41. W. Ren, R. W. Beard, and E. M. Atkins, "Information consensus in multivehicle cooperative control," *IEEE Control System Magazine*, vol. 27, no. 2, April 2007.
42. R. Roman, J. Zhou, and J. Lopez, "Applying intrusion detection systems to wireless sensor networks," *Consumer Communications and Networking Conference*, vol. 1, pp. 640–644, January 2006.
43. L. Sabattini, N. Chopra, and C. Secchi, "On decentralized connectivity maintenance for mobile robotic systems," *50th IEEE Conference on Decision and Control*, Orlando, FL, USA, December 2011.
44. R. Selmic, J. Kanno, J. Buchart, N. Richardson, "Quadratic optimal control of sensor network deployment," *Cyberspace Research Workshop*, Shreveport, LA, November 2007.
45. V.de Silva and R. Ghrist, "Homological sensor networks," *Notices of the American Mathematical Society*, vol. 54, no. 1, pp. 10–17, Jan. 2007.
46. "Plex: simplicial complexes in MATLAB," website: <http://comptop.stanford.edu/programs>.
47. S. Slijepcevic and M. Potkonjak, "Power efficient organization of wireless sensor networks," *IEEE International Conference on Communications*, vol. 2, pp. 472–476, June 2001.
48. D. Tian and N. Georganas, "A coverage-preserving node scheduling scheme for large wireless sensor networks," *ACM Workshop of Wireless Sensor Networks and Applications*, Atlanta, GA, October 2002.
49. L. Vietoris, "Über den höheren zusammenhang kompakter Raume und eine klasse von zusammenhangstreuen abbildungen," *Mathematische Annalen*, vol. 97, pp. 454–472, 1927.
50. G. Want, G. Cao, and T. LaPorta, "A bidding protocol for deploying mobile sensors," *Proc. 11th IEEE Int. Conf. Network Protocols (ICNP'03)*, Atlanta, GA, Nov. 2003, pp. 80–91.
51. G. Wang, G. Cao, and T. LaPorta, "Movement-assisted sensor deployment," *Proc. IEEE InfoCom*, Hong Kong, March 2004, pp. 80–91.
52. X. Wang, G. Xing, Y. Zang, C. Lu, R. Pless, and C. Gill, "Integrated coverage and connectivity configuration in wireless sensor networks," *ACM SenSys*, Los Angeles, November 2003.
53. G. Werner-Allen, K. Lorincz, M. Ruiz, O. Marcillo, J. Johnson, J. Lees, and M. Welsh, "Deploying a wireless sensor network on an active volcano," *IEEE Internet Computing*, vol. 10, no. 2, pp. 18–25, April 2006.
54. D.B. West, *Introduction to Graph Theory*, Prentice Hall, Upper Saddle River, New Jersey, 2001.
55. P. Yang, R.A. Freeman, G.J. Gordon, K.M. Lynch, S.S. Srinivasa, and R. Sukthankar, "Decentralized estimation and control of graph connectivity for mobile sensor networks," *Automatica*, vol. 46, pp. 390–396, 2010.
56. F. Yu, Y. Choi, S. Park, Y. Tian, and S. Kim, "A hole geometric modeling in wireless sensor networks," *International Conference on Wireless Communications, Networking and Mobile Computing*, pp. 2432–2435, September 2007.
57. H. Zhang and J.C. Hou, "Maintaining sensing coverage and connectivity in large sensor networks," *Wireless Ad Hoc Sensor Networks: An International Journal*, vol. 1 no. 1–2, pp. 89–124, 2005.

58. Y. Zou and K. Chakrabarty, "Sensor deployment and target localization based on virtual forces," *IEEE INFOCOM 2003*, March 2003.
59. Y. Zou and K. Chakrabarty, "Uncertainty-aware sensor deployment algorithms for surveillance applications," *Proc. IEEE Global Communications Conference*, December 2003.
60. Y. Zou and K. Chakrabarty, "Sensor deployment and target localization in distributed sensor networks," *IEEE Trans. Embedded Comput. Syst.*, vol. 3., no. 1, pp. 61–91, 2004.