

## Review Article

# Hole Detection for Quantifying Connectivity in Wireless Sensor Networks: A Survey

**Pearl Antil and Amita Malik**

*Computer Science and Engineering, DCRUST, Murthal, Sonapat 131039, India*

Correspondence should be addressed to Pearl Antil; [pearlantil12@gmail.com](mailto:pearlantil12@gmail.com)

Received 31 May 2014; Revised 9 September 2014; Accepted 14 September 2014; Published 7 October 2014

Academic Editor: Tin-Yu Wu

Copyright © 2014 P. Antil and A. Malik. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Owing to random deployment, environmental factors, dynamic topology, and external attacks, emergence of holes in wireless sensor networks is inescapable. Hole is an area in sensor network around which sensors cease to sense or communicate due to drainage of battery or any fault, either temporary or permanent. Holes impair sensing and communication functions of network; thus their identification is a major concern. This paper discusses different types of holes and significance of hole detection in wireless sensor networks. Coverage hole detection schemes have been classified into three categories based on the type of information used by algorithms, computation model, and network dynamics for better understanding. Then, relative strengths and shortcomings of some of the existing coverage hole detection algorithms are discussed. The paper is concluded by highlighting various future research directions.

## 1. Introduction

A wireless sensor network is a network that is made of hundreds or thousands of sensor nodes which are densely deployed in an unattended environment with the capabilities of sensing, wireless communications, and computations [1]. Sensor nodes are tiny, low power devices equipped with processor, memory, radio, actuator, and power supply. Radio transmitters and receivers help sensor nodes to communicate with each other.

Some of the challenges in WSNs are node deployment, energy consumption, node heterogeneity, data aggregation, and fault tolerance [2]. Hole detection is one of the major problems in WSN. Holes affect the network capacity and perceptual coverage of the network. Due to limited battery the nodes may die with passage of time. In case of random deployment, there is a huge possibility that all areas of target region are not covered properly leading to formation of holes. Detection of holes is important because of their negative and damaging effects.

WSNs have myriad of interdisciplinary applications such as weather forecasting, battlefield surveillance, threat identification [3], health monitoring [4], environment monitoring [5], and wild life monitoring. All those applications that

demand random deployment and uncontrolled environment suffer from holes problem [6]. Thus, hole detection can be useful for all disciplines of sciences and engineering.

The rest of the paper is organized as follows. Section 2 discusses different types of holes in WSNs and importance for hole detection algorithms. Section 3 proposes a novel taxonomy of coverage hole detection algorithms and reviews some of the existing coverage hole detection algorithms referring to taxonomy. Conclusions and future research directions are presented in Section 4.

## 2. Holes in WSN

An area where a group of sensor nodes stops working and does not take part in data sensing and communication is termed as a hole in the network [7]. Holes are the barriers for communication. Holes have a huge impact on the performance of the network. Hole detection identifies damaged, attacked, or inaccessible nodes. If there is a hole in the network then data will be routed along the hole boundary nodes again and again which will lead to premature exhaustion of energy present at these nodes [8]. This will ultimately increase the size of hole in the network. Detection

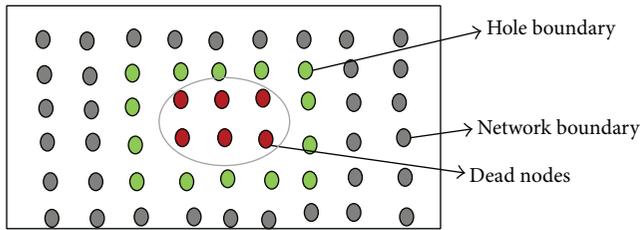


FIGURE 1: Hole and its boundary.

of holes avoids the additional energy consumption around holes because of congestion. It assures long network life and adequate quality of service.

One of the most prominent problems in wireless sensor networks is detection of network boundary, that is, the nodes that are present on the boundary of network and hole boundary [7], that is, the nodes that surround a hole as shown in Figure 1.

Holes can be classified with respect to their mobility (mobile or static), lifetime (persistent or temporary), purposes (intentional or unintentional), affected function (functional or nonfunctional), and cause of anomaly [9] as shown in Figure 2. Holes formed in static sensor network are called *static holes*. Mobile nodes create and break connections while moving, thus creating *mobile holes*. External events such as wildfire or heavy rain destroy sensor nodes in an area leading to formation of *persistent holes*. Inefficient sleep wake-up scheduling of sensor nodes can form *temporary holes* in the target region. *Unintentional holes* are formed when a node accidentally lacks some physical capabilities. *Intentional holes* are formed when some part of target field is not sensed by any node because nodes in that area are intentionally scheduled to be in sleep state to save energy. If, due to failure of some nodes sensing, communication and processing tasks of sensor network are affected, then the corresponding hole formed is functional hole. On the other hand if some nonfunctional tasks such as security are affected then nonfunctional holes are formed.

Jabeur et al. [9] discussed PLMS (physical/logical/malicious/semantic), a cause-based taxonomy in detail, where sensor holes are studied from a cause-effect-solution perspective. *Semantic holes* mainly relate to collection, processing, and routing of data. Some of the reasons for their formation are inaccurate data collection and absence of specialized sensors. *Logical holes* are created due to cluster based approaches where a sensor node may not get support from its neighbors belonging to the same cluster. *Physical holes* may arise due to limited processing capabilities of sensor nodes called processing holes, through excessive consumption of energy at some nodes leading to formation of energy holes, or due to limited sensing range of nodes creating sensing holes. Coverage and routing holes come under physical holes category. *Coverage holes* are the result of unplanned deployment of nodes because of which target area is not covered properly. They may also occur due to changing topology due to which some sensor nodes may move over time leading to coverage holes. *Routing holes* occur in wireless

sensor networks if either there is insufficient number of nodes or the nodes that are available cannot take part in routing [10]. These holes have a direct effect on the routing performance of the network. These holes mostly occur in geographic greedy forwarding because of local minimum phenomenon.

Holes formed by malicious behavior of some nodes in the network are called malicious holes. Further categorization of malicious node includes jamming, sink, and trust holes. *Jamming holes* occur if some high frequency signals block the radio frequency used for communicating by sensor nodes. Denial of service attacks causes *Sink/black holes* [11] where an adversary announces attractive routes to sink and thus succeeds in deceiving other nearby nodes which may select this node as their next hop. Trust hole is a new type of holes proposed by author that comes under *malicious holes* category. Sensors assign weighted trust values based on different parameters to their neighbors. Trust weight threshold values are application dependent. We can see a trust hole in Figure 3, delimited by sensors having weight values less than or equal to 0.5.

Out of all the types of holes discussed so far, coverage holes are the most important to detect as they play a vital role in assuring good QoS. They help us to identify whether each point in sensing field has the required degree of coverage or not. They are the indicators of general health of sensor networks and help in identifying geographic characteristics of target region [12]. Hole boundaries can determine areas of interest like fire, flood, or earthquake. They help us to find out locations where more nodes need to be deployed and thus assist in patching the holes. Hole detection ensures reliability by preventing data loss.

Coverage hole detection aids in identifying alternative communication pathways and assists in regulating data traffic flow. Absence of hole detection algorithms will eventually cause problems in routing. Thus, the rest of the paper is focused on identifying different ways to detect coverage holes in sensor networks.

### 3. Taxonomy of Coverage Hole Detection Algorithms

There has been a large body of research on detection of coverage holes in WSNs over the last few years. In Figure 4, coverage hole detection algorithms have been classified into different categories: (A) based on the type of information used, (B) based on computational model, and (C) based on network dynamics. In this section, we analyze and summarize some of the typical hole detection algorithms of each category.

*3.1. Approaches Based on Type of Information Used.* The efforts invested so far in the field of coverage hole detection based on the type of information used by the algorithms can be divided into three approaches, namely, topological, statistical, and geometrical approaches [13].

*3.1.1. Geographical Approach.* This approach assumes that exact location of sensor nodes is known beforehand. Each

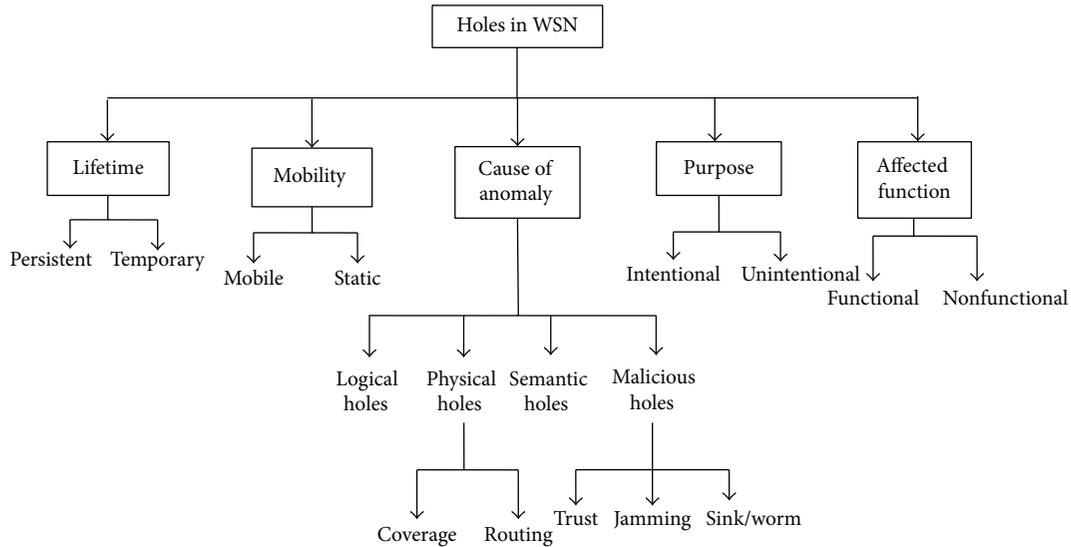


FIGURE 2: Types of holes.

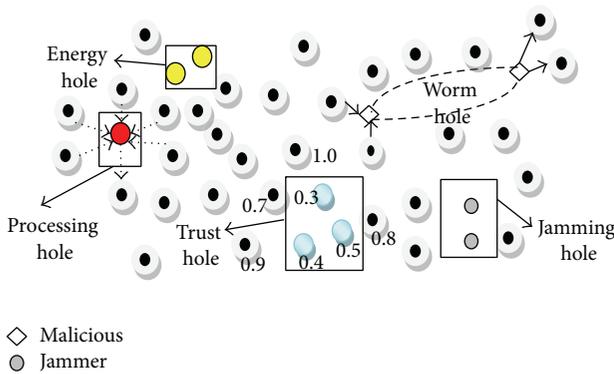


FIGURE 3: Sensor network with different holes.

node knows its location either with the help of special location hardware such as GPS [12] or by using scanning devices, thereby increasing size and structure of sensor nodes. It is also known as location based approach.

Yang and Fei [14] proposed a hole detection and adaptive geographical routing (HDAR) algorithm to detect holes and to use this information to deal with local minimum problem. If the angle between two adjacent edges of a node is greater than 120 degrees, then it begins hole detection algorithm. Here, the ratio of network distance over the Euclidean distance is used as metric to detect a hole; that is,  $Length_{pro}(p, x)/dist_{euc}(p, x)$  is the hole detection ratio. If for a node the value of hole detection ratio is greater than a predefined threshold, then it is on boundary. One of the main advantages is that a single node can efficiently detect the hole. After detecting hole, it advertises this information to nodes in vicinity which can adaptively adjust the forwarding direction.

**3.1.2. Topological Approach.** Also called as connectivity based approach, it uses only the available connectivity information

of network to detect holes [13]. This approach requires no location information and works even for dense networks. There is no assumption about node distribution. One of the algorithms based on topological approach to detect coverage holes within wireless sensor networks was given in [15] and later improved in [16]. Authors proposed a distributed cooperative scheme based on the fact that nodes at the hole or network boundary have smaller degrees than interior nodes. It deals with static, uniformly distributed nodes with each node having a unique id. If the degree of a node is lower than the average degree of its 2-hop neighbors, then it makes a decision whether it is on hole boundary or not. If yes, then it sends messages informing its status to its 1-hop neighbors who may also be on hole boundary. The algorithm is scalable, but approach produces poor results for randomly deployed dense networks. If there are not sufficient nodes surrounding a hole then output produced is not accurate.

Martins et al. [17] used the concepts of Rips complex and Cech complex to discover coverage holes. If communication radius is greater than or equal to twice the sensing radius and there is a hole in Rips complex, then there must be a hole in Cech complex. Authors classified holes to be either triangular or nontriangular. The distributed algorithm proposed by authors is capable of detecting nontriangular holes and the area of triangular holes. After constructing neighbor graph, each node checks whether there exists a Hamiltonian cycle in graph. If not, then node is on the hole boundary. After making decision, each node broadcasts its status to its neighbors. The algorithm further finds cycles bounding holes.

Zeaddly et al. [18] proposed a hop based approach to find holes in sensor networks. There are three phases, namely, information collection phase where each node exchanges information to build a list of x-hop neighbors, path construction phase where communication links between sensor nodes in list of x-hop neighbors are identified, and finally path checking phase where paths are examined to infer boundary and inner nodes. If the communication path of

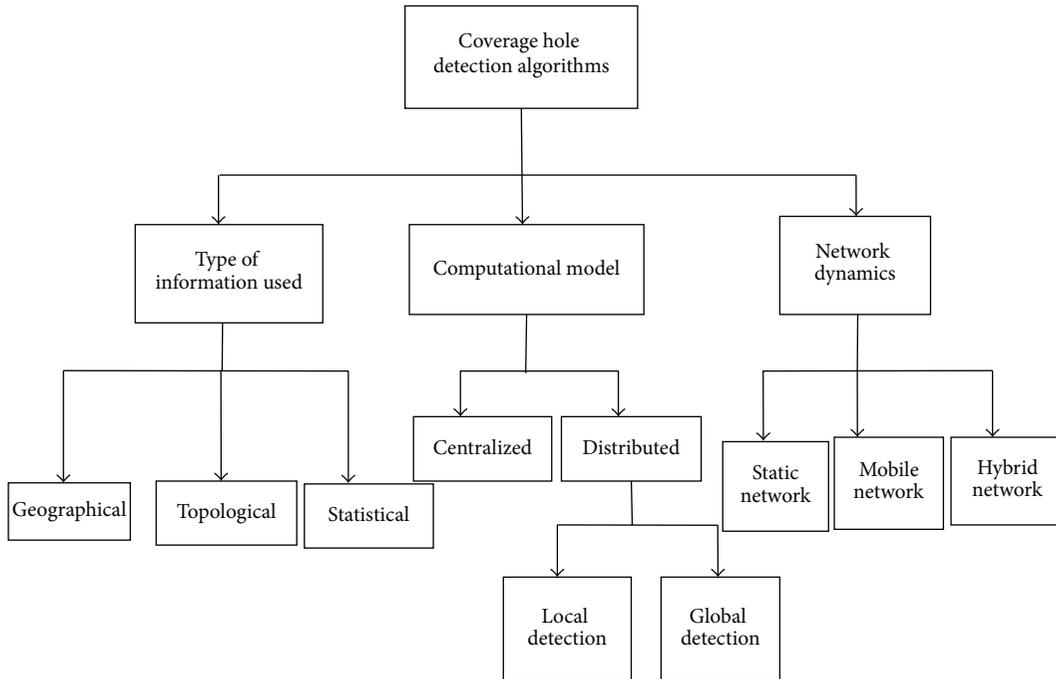


FIGURE 4: Coverage hole detection algorithm categorization in WSN.

x-hop neighbors of a node is broken, then it is boundary node. Algorithm works for node degree of 7 or higher which is better than some of the other approaches, but there is a huge communication overhead involved to identify x-hop neighbors.

Chu and Ssu [19] proposed a decentralized boundary detection (DBD) algorithm to identify sensor nodes near a hole or obstacle in WSN using topological approach. Each node knows its three-hop neighbors by exchanging HELLO messages and one-hop and two-hop node information. There is no UDG constraint. Each node then constructs 2-hop neighbor graph. If cycle exists in such a graph, then there is a hole in network. For dealing with hole which is not included totally inside 2-hop neighbor graph, another rule based on contour structure was developed. Detection of broken contour line implies either network boundary or an obstacle.

There are very few algorithms that can detect boundaries of small holes. Dong et al. [20] proposed a distributed algorithm that can accurately detect boundaries of small holes in the network. The first step of algorithm is to reduce the connectivity graph by using vertex deletion and edge deletion so as to obtain a skeleton graph. Thereafter, skeleton graph is further partitioned to get coarse inner boundary cycles. Each cycle either encloses a hole of graph or corresponds to outer boundary. The coarse outer boundaries are then further refined to get fine grained boundary cycles. There is no assumption related to node density. The authors further proved the correctness of hole detection.

**3.1.3. Statistical Approach.** Some statistical function is applied on data collected from neighboring nodes and

then a Boolean function decides whether nodes are at hole boundary or not. It is based on the assumption that distribution of nodes follows some statistical function. There is no need of GPS, but it requires high node density; that is, average degree must be 100 or higher [21]. In practice, such dense uniform deployment is not practical. The concept used by Fekete et al. [21] is that boundary nodes have lower average degrees than interior nodes. Average density is the metric used to detect holes where actual node degree is compared with a predefined threshold value to infer its position. Another metric used by Fekete et al. [21] is centrality index, where high value is assigned to inner nodes and less value to outer nodes.

*Discussion.* We have considered many algorithms under different categories each having its own limitations. Geometrical approach for hole detection requires GPS enabled sensor and is expensive. They consume a lot of energy and it is not practical for sensors to know their exact location in hostile environment. Considering huge applications of WSN, these approaches have limited scope. Statistical approaches provide optimal performance, but they are computationally expensive. Owing to the challenges in wireless sensor networks it is not desirable that nodes perform complex mathematical and statistical calculations [13]. Topological approach provides realistic results but involves communication overhead. Some of the algorithms do not work for small network degrees.

**3.2. Algorithms Based on Computational Model.** On the basis of computational model used, coverage hole detection algorithms can be classified into two categories, namely, centralized and distributed.

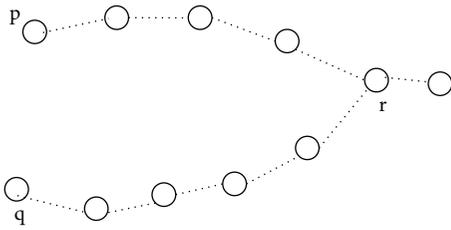


FIGURE 5: Nearest common ancestor of  $p$  and  $q$  is node  $r$ .

**3.2.1. Centralized.** Centralized algorithms run on one or more nodes at centralized location. Bldg and Funke [28] presented a centralized hole detection algorithm based on connectivity information of the network. Authors select a set of nodes and then identify whether isolevels generated from each node are broken or continuous. Algorithm is based on the fact that isolevels are always broken at either the network boundary or the hole boundary. It identifies the nodes near boundaries but gives no idea of how they might be connected. Moreover, it requires dense deployment. Ghrist and Muhammad [25] have introduced a centralized technique for detecting coverage holes based on homology, an algebraic topology variant. It uses only available connectivity information to detect single level coverage holes. Time complexity is  $O(n^5)$ , with  $n$  being the number of nodes. Approach gives no guarantee to detect hole boundary accurately.

**3.2.2. Distributed.** In decentralized algorithms, multiple nodes work together to efficiently detect hole in the network resulting in uniform division of workload. Senouci et al. [24] proposed a hole detection and healing (HEAL), a distributed and localized algorithm which can detect coverage holes using distributed coverage hole detection and can heal the detected holes using virtual forces concept. Attractive and repulsive forces act inside hole healing area (HHA) to recover from holes without any side effects. The computational complexity of this algorithm is  $O(v^2)$ , where  $v$  is the average number of 1-hop neighbors. HEAL deals with holes of various forms and sizes and provides cost effective and accurate solution for hole detection and healing. Kröllner et al. [30] developed a distributed algorithm to detect holes in large sensor networks using quasiunit disk graph. Algorithm does not perform well in sparse networks where identification of even one flower is difficult.

Wang et al. [27] presented a simple, distributed boundary recognition algorithm based on repetitive network flooding. Nearest common ancestors (NCA) are used to detect holes. In Figure 5, node  $r$  is the nearest common ancestor of nodes  $p$  and  $q$ . Algorithm builds the shortest path tree rooted at some arbitrary nodes. Nodes where shortest paths of distinct homotopy types meet past the hole are called cut nodes.

After determining cuts in shortest path tree, coarse inner boundary which is a shortest cycle enclosing the composite hole is detected. Network is then flooded with shortest cycle to identify external nodes, that is, nodes whose hop counts to cycle are locally maximal. Finally coarse inner boundary is refined. There is no UDG constraint. Huge communication

overhead is involved for synchronizing nodes before tree is constructed. Moreover, algorithm fails if there are multiple adjacent holes.

On the basis of nodes that invoke hole detection algorithm, we can further classify distributed algorithms into two categories, namely, *local detection* and *global detection*. A hole occurs when several adjacent nodes in sensor network fail and is defined as the convex hull of the region containing failed sensors [22]. Holes affect their immediate neighbors, as well as remote nodes whose routing options are reduced. Based on the nodes which invoke hole detection algorithm, there are two categories of hole detection algorithms, namely, local detection and global detection. In local detection, nodes that are immediately affected by the hole initiate hole detection algorithm. However, in global detection, hole is detected by a node that is far away and is not affected directly by damage caused by hole.

*(a) Local Detection.* Corke et al. [22] presented, analyzed, and implemented two distributed algorithms for hole detection. In local detection, node initialization phase is necessary to know about initial connectivity state. In the initialization phase, each node sends out a ping requesting neighbor information. Nodes that reply with their id are added to the list of neighbors. Neighbors are then pinged periodically to verify integrity of network. If neighboring node gives no response to pinged message, then it is regarded as dead. When dead count exceeds dead threshold, then node marks itself as sitting on damage perimeter. During damage extent computation, each node finds out the overall convex hull incrementally by calculating convex hulls at each node and sharing the hulls until there are no further changes to the hull set. Wang et al. [27] have developed a distributed algorithm where each node first checks whether it is a stuck node or not using TENT rule. If a node is closer to target than all its 1-hop neighbors, then it is a stuck node. The stuck nodes are marked as hole boundary nodes. Boundhole algorithm then builds routes around hole boundary to route packets. Approach needs only angle information within its 1-hop neighbors; therefore detection is local.

*(b) Global Detection.* In global method, routing information is used to detect the holes. In paper [22], path density is the metric used to detect presence of holes. Each node randomly broadcasts a diagnostic packet containing source coordinate, destination coordinate, previous coordinate, distance travelled, and message id. Distance travelled and previous coordinate are updated at each hop. Message id is used to ignore duplicate messages. When message reaches destination, path density is calculated as the ratio of straight line distance from source to destination to actual distance travelled. Low ratio implies that a hole exists in the network.

*Discussion.* Limited storage is one of the challenges of sensor networks; thus, algorithms that collect whole data at some node to perform centralized computations are infeasible. Centralized approach suffers from problem of single point of failure. Due to dynamic topology and exogenous disturbances, WSNs need dense deployment. Protocols designed

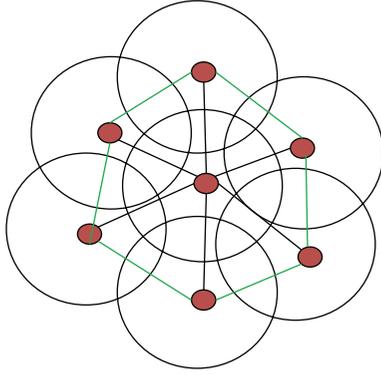


FIGURE 6: 3MeSH ring.

for such dense networks should be distributed in order to accommodate scalable architecture. Centralized algorithms consume more energy to achieve higher percentage of hole area recovered as compared to distributed algorithm. On the other hand, distributed algorithms are more complex than centralized as they need to run on many nodes throughout the network. Global approach helps to overcome ambiguity often faced by local detection. Local detection algorithms run periodically to give information about health of sensor network. They can be used to look for changes from initial state.

**3.3. Algorithms Based on Network Dynamics.** Algorithms discussed in this section are classified based on network dynamics in sensor network as static sensors, mobile sensors, and hybrid sensors.

**3.3.1. Static Sensors.** Li et al. [34] proposed 3MeSH (triangular mesh self-healing hole detection algorithm), which is a distributed coordinate-free hole detection algorithm. It is assumed that each node has uniform sensing radius  $R$  and communication radius  $2R$ . Initially a subset of active nodes is selected. An active node  $x$  is neighbor of active node  $y$  if they are between  $R$  and  $2R$  distance apart. Nodes that lie within the sensing range of an active node are called redundant nodes. Connectivity information is collected by each active node from its neighbors. If node detects presence of 3MeSH ring as shown in Figure 6 defined by all its neighbors, then it is a boundary node. For detecting large holes, nodes are allowed to collect connectivity information from nodes further away but at the cost of increased complexity. 3MeSH-DR is an improved version of 3MeSH algorithm presented in [31].

**3.3.2. Mobile Sensors.** Mobile sensor nodes can move around after initial deployment. The main objective in mobile WSN is to maximize coverage. Topology of sensor network is affected by mobile nodes as they form new connections and break old ones. If the coverage area of a node can be covered by its neighboring node, then only a node can move to another area. In [26], authors have used Voronoi diagrams to detect coverage holes. Voronoi diagram is a diagram of boundaries around each sensor such that every point within sensor's

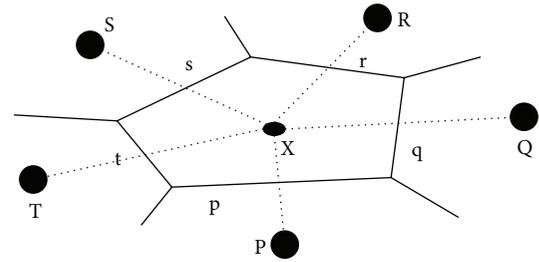


FIGURE 7: Voronoi cell.



FIGURE 8: Voronoi diagram.

boundary is closer to that sensor than any other sensor in the network [35]. Figure 7 shows a Voronoi cell. Nodes P, Q, R, S, and T are Voronoi neighbors of X and a, b, c, d, and e represent Voronoi edges. These edges are the vertical bisector of line connecting X to its Voronoi neighbors.

A polygon constructed from all these Voronoi edges is called Voronoi cell of node X. We can see a Voronoi diagram in Figure 8.

To detect hole, each node checks whether its Voronoi polygon is covered by its sensing area. If not, then coverage hole exists. After detecting hole, one of the three proposed algorithms is used to find target location where a mobile node is to be moved so as to heal the hole. In vector based algorithm (VEC), sensor nodes are pushed from dense regions to sparse regions so that nodes are evenly distributed. Voronoi based algorithm (VOR) is a pull algorithm which pulls nodes towards sparse regions. In minimax algorithm, target location is at the center of its Voronoi polygon. Minimax algorithm achieves maximum coverage at the cost of increased computational overhead.

Zhao et al. [32] proposed a coverage hole detection method (CHDM) by mathematical analysis. It is assumed that network consists of mobile nodes each with sensing radius  $r$  and communication radius  $2r$ . A node  $p$  is defined as neighbor of node  $q$  if it lies in its communication range. On the basis of central angle between neighbor sensors, the authors presented different cases to find coverage holes in communication circle around a redundant movable node. To patch hole, a redundant node is moved to an appropriate position inside the hole.

TABLE I: Characteristics of various coverage hole detection algorithms.

Researcher	Approach	Computational model	Network dynamics	Detection level	Simulation environment
Yang and Fei [14]	Geographical	Distributed	Static	Local	Easim 3D
Corke et al. [22]	Geographical	Distributed	Static	Both	MATLAB
Fang et al. [23]	Topological	Distributed	Static	Local	MATLAB
Zeadally et al. [18]	Topological	Distributed	Static	Local	GTSNetS
Bi et al. [16]	Topological	Distributed	Static	Local	Unknown
Senouci et al. [24]	Geographical	Distributed	Mobile	Local	Ns-2
Fekete et al. [21]	Statistical	Distributed	Static	Local	Unknown
Ghrist and Muhammad [25]	Topological	Centralized	Static	—	MATLAB
Chu and Ssu [19]	Topological	Distributed	Static	Local	Ns-2
Guiling et al. [26]	Geographical	Distributed	Mobile	Local	Ns-2
Wang et al. [27]	Topological	Distributed	Static	Global	Unknown
Bldg and Funke [28]	Topological	Centralized	Static	—	Unknown
Babaie and Pirahesh [29]	Geographical	Distributed	Hybrid	Local	Unknown
Krölller et al. [30]	Topological	Distributed	Static	Global	SHAWN
Li and Hunter [31]	Topological	Distributed	Static	Local	MATLAB
Zhao et al. [32]	Geographical	Distributed	Mobile	Local	Unknown
Wang et al. [33]	Geographical	Distributed	Hybrid	Local	Ns-2
Martins et al. [17]	Topological	Distributed	Static	Local	MATLAB
Dong et al. [20]	Topological	Distributed	Static	Local	Unknown

3.3.3. *Hybrid Sensors.* Hybrid WSNs consist of both stationary and mobile nodes. Mobile nodes help in healing the coverage holes created by stationary nodes. Babaie and Pirahesh [29] presented a triangular oriented diagram to detect a hole. The authors connected the center of three adjacent sensors to produce triangle and further presented various possibilities of occurrence of holes and then calculated the area which is not covered by any of them. This uncovered area is coverage hole. To patch the coverage hole, a mobile node is moved to incenter or circumcenter depending on whether hole area is more or less than sensing region, respectively. The approach is simple as compared to Voronoi diagram construction. Moreover, it can determine exact area of hole.

Wang et al. [33] proposed a distributed bidding protocol for detecting coverage holes in hybrid sensor networks. Mobile nodes act as hole healing server with certain base price initially set to zero. Base price is the estimate of new coverage hole generated when a mobile node moves to new target location for healing purpose. During advertisement phase, mobile sensors broadcast their locations and base price. In bidding phase, static sensors examine their respective Voronoi cell to detect coverage holes locally. If hole exists, it estimates its size and determines target position inside the hole. Static nodes then calculate the bid using formula  $\pi * (d - \text{sensing-range})^2$ , with  $d$  being the distance between target location and the node that wants to bid. Static sensors send bidding message to the nearest mobile sensor whose base price is less than its bid. Each mobile node compares the received bids and chooses the message with highest bid and moves to the target location specified in message. This bid becomes new base price of mobile node. This process stops

when the bid of any static sensor is not higher than the base price of any mobile sensor.

*Discussion.* The probability of the formation of hole in a high density network is less. Therefore, for dense networks static sensor networks are preferred while in case of sparse density, hybrid and mobile sensor networks give better coverage. Some of the nodes in hybrid and mobile networks can move to patch the coverage holes, thereby improving the network connectivity. Sensors moved through a long distance will consume more energy. If energy of a sensor is so less that it dies shortly after being relocated to target region, then this effort is wasted. Therefore, moving and messaging cost must be kept at minimum while dispatching mobile nodes to target locations in hybrid and mobile sensor networks.

#### 4. Summaries and Outlook

In this paper, taxonomy is proposed which gives an exhaustive classification of coverage hole detection algorithms. In the last few years, many innovative ideas have been provided by researchers to solve coverage hole detection problem in WSNs, but research in this direction is still in growing phase. Table 1 summarizes the characteristics of various coverage hole detection algorithms in aspect of approach used, computational model, network dynamics, detection level, and simulation environment. From Tables 2 and 3, it can be seen that each algorithm has its own strengths and shortcomings, but none is absolutely the best. Table 4 discusses some of the application suggestions for coverage hole detection algorithms.

TABLE 2: Strengths and shortcomings of coverage hole detection algorithms.

Researcher	Year	Concept used	Advantages	Disadvantages
Wang et al. [33]	2003	Using Voronoi diagram	Each node needs to monitor small area around it; low communication complexity	Performance depends on ratio of mobile to static sensors; no obstacle modeling
Guiling et al. [26]	2004	Using computational geometry	Extensible to large deployments because communication and movements are local	Poor performance on initial clustered deployment and lower communication range
Fekete et al. [21]	2004	Average density and centrality density	Avoids GPS	Complex computations; unrealistic assumption on sensor distribution and density
Ghrist and Muhammad [25]	2005	Using homology	Works in coordinate free sensor networks	Not scalable; complex calculations
Bldg and Funke [28]	2005	Using isosets and isocontours	Works for both continuous and discrete cases and avoids GPS	Requires high network density and huge overhead in selecting seed nodes and constructing isolevels
Krölller et al. [30]	2006	Using combinatorial structures	Supports quasi-UDG	Fails in sparse n/w and deals with complex combinatorial structures
Fang et al. [23]	2006	Using stuck nodes	No assumption about node distribution	High message complexity and UDG constraint
Wang et al. [27]	2006	Using shortest path tree	No UDG constraint; works for random distribution	Needs high node density, high communication overhead due to flooding and cannot detect multiple adjacent holes
Bi et al. [16]	2006	Cooperative neighbor and coverage ratio	Scalable	Not suitable for randomly deployed dense networks
Corke et al. [22]	2007	Using convex hulls	Can discover holes based on normal message traffic	Local detection algorithm needs initial state information; global detection algorithm needs location information
Li and Hunter [31]	2008	Using homology and connectivity information	Scalable, large holes can be easily detected	Fails for trivial holes
Dong et al. [20]	2009	Using FGP transformation	Can detect small holes accurately; correctness of hole detection is proved	Communication overhead
Yang and Fei [14]	2010	Using TENT rule	Single node can detect hole efficiently	Needs localized nodes
Zhao et al. [32]	2011	Mathematical analysis	Can find precise location of holes	Needs location information and works only for mobile nodes
Martins et al. [17]	2011	Rips and Cech complexes	Can detect nontriangular holes efficiently	Communication overhead
Zeadally et al. [18]	2012	Using hop based approach	Works for node degree 7 or higher	Communication overhead to identify x-hop neighbors
Babaie and Pirahesh [29]	2012	Using triangular oriented structure	Exact hole area is identified	Needs localized nodes
Chu and Ssu [19]	2014	Using contour lines	No UDG constraint; suitable for dynamic hole detection	Communication overhead
Senouci et al. [24]	2014	Using TENT rule and virtual forces' concept	Minimizes resource consumption	Cannot detect holes at the network boundaries; UDG constraint

Having surveyed various coverage hole detection algorithms, this section discusses some of the future research directions. Authors have made a good attempt to detect holes, but most of the algorithms produce poor results for nonuniformly deployed dense networks. Computational complexities of the topological algorithms have not been analyzed. Further research on optimizing algorithms in hybrid sensor

networks and evaluating tradeoffs between latency and data gathering techniques can provide valuable information to optimize coverage and connectivity. Although the literature is replete with coverage hole detection techniques, so far there is no application specific hole detection protocol. There exist very few protocols that can deal with open holes at the network boundaries.

TABLE 3: Comparison of hole detection algorithms under different parameters.

Researcher	Communication overhead	Node density	Scalability	Mobility adaptation
Wang et al. [33]	Low	Medium	Medium	Yes
Guiling et al. [26]	Low	Medium	High	Yes
Fekete et al. [21]	Medium	Very high	Medium	No
Ghrist and Muhammad [25]	High	Medium	Low	No
Bldg and Funke [28]	High	High	Low	No
Krölller et al. [30]	High	High	Medium	No
Fang et al. [23]	High	Medium	Medium	No
Wang et al. [27]	High	High	High	No
Bi et al. [16]	High	High	High	No
Corke et al. [22]	Low	Medium	Medium	No
Li and Hunter [31]	High	Medium	High	Yes
Dong et al. [20]	High	Low	Medium	No
Yang and Fei [14]	Low	Medium	Medium	No
Zhao et al. [32]	Low	Medium	High	Yes
Martins et al. [17]	High	Medium	Medium	No
Zeadally et al. [18]	High	Medium	Medium	No
Babaie and Pirahesh [29]	Low	Medium	Medium	Yes
Chu and Ssu [19]	High	Medium	Medium	No
Senouci et al. [24]	Low	Medium	High	Yes

TABLE 4: Application suggestions for coverage hole detection algorithms.

Field	Desirable features	Applicable categorization
Industry (control and monitoring of equipments)	Medium scalability, static or hybrid nodes, indoor performance, easy localization, and easy hole healing	[14, 21, 22, 29, 33]
Military (intrusion detection)	High scalability, outdoor performance, robustness, difficult localization, long network lifetime, and high energy efficiency	[18, 19, 23, 27, 30]
Home (home automation)	Low scalability, indoor performance, easy localization, and static nodes	[14, 22]
Environment monitoring (wildlife habitat monitoring, detecting fires, and monitoring seismic activities, temperature, and pressure)	Low degree of coverage, high scalability, mostly outdoor performance, mostly mobile nodes, high energy efficiency, and difficult localization	[16–18, 20, 23, 31]
Automotive (driving assistance)	High scalability, mobile nodes, outdoor performance, fault tolerance, and good localization	[24, 26, 32, 33]
Agriculture (field monitoring and predicting crop water requirement)	Medium scalability, easy localization, static nodes, and outdoor performance	[14, 22, 24–26, 28]

Future work can also focus on mitigating holes problem after its occurrence and provide an optimal hole healing process. Some of the ways to avoid the hole formation can be using mobile sinks or multiple sinks. Nodes forward their data to sink when mobile sink gets closer, thereby avoiding unnecessary energy wastage in multihop delivery paths. In order to save energy of sensor nodes, multiple sinks can be deployed in target area so as to reduce the distance data has to travel to reach sink. Mobility of nodes can be exploited to patch the coverage holes. Algorithms can be built that give special considerations to nodes that are on the verge

of becoming a hole. Another interesting work can be on developing commercial products to detect holes in WSNs.

Most of the algorithms consider nodes being deployed in 2-dimensional environment, but in real-world applications, sensor nodes are deployed in 3-dimensional environment. So, work can be done in this direction. One of the interesting research issues in the future can be detecting the presence of holes in underwater and underground WSNs. Mathematical modelling and real-world testing must be included in actual steps of research. This will also allow fair comparison among algorithms.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## References

- [1] S. Kalantary and S. Taghipour, "A survey on architectures, protocols, applications and management in wireless sensor networks," *Journal of Advanced Computer Science*, vol. 3, no. 1, 2014.
- [2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, 2002.
- [3] A. Agah, S. K. Das, K. Basu, and M. Asadi, "Intrusion detection in sensor networks: a non-cooperative game approach," in *Proceedings of the 3rd IEEE International Symposium on Network Computing and Applications (NCA '04)*, pp. 343–346, Cambridge, Mass, USA, September 2004.
- [4] L. Schwiebert, S. K. S. Gupta, and J. Weinmann, "Research challenges in wireless networks of biomedical sensors," in *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking (MobiCom '01)*, pp. 151–165, Rome, Italy, July 2001.
- [5] E. S. Biagioni and K. W. Bridges, "The application of remote sensor technology to assist the recovery of rare and endangered species," *International Journal of High Performance Computing Applications*, vol. 17, no. 3, pp. 315–324, 2002.
- [6] R. Bhardwaj and H. Sharma, "Holes in wireless sensor networks," *International Journal of Computer Science and Informatics*, vol. 2, no. 4, pp. 62–63, 2012.
- [7] I. M. Khan, H. Mokhtar, and M. Merabti, "An overview of holes in wireless sensor networks," in *Proceedings of the 11th Annual Post Graduate symposium on the Convergence of Telecommunications, Networking & Broadcasting (PGNET '10)*, Liverpool, UK, June 2010.
- [8] M. K. Watfa and S. Commuri, "Energy-efficient approaches to coverage holes detection in wireless sensor networks," in *IEEE International Symposium on Intelligent Control*, vol. 4302, pp. 131–136, 2006.
- [9] N. Jabeur, N. Sahli, and I. Muhammad, "Survey on sensor holes?: a cause-effect-solution perspective," *Procedia Computer Science*, vol. 19, pp. 1074–1080, 2013.
- [10] N. Ahmed, S. S. Kanhere, and S. Jha, "The holes problem in wireless sensor networks: a survey," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 9, no. 2, pp. 4–18, 2005.
- [11] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: attacks and countermeasures," in *Proceedings of the 1st IEEE International Workshop (SNPA '03)*, May 2003.
- [12] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins, *Global Positioning Systems: Theory and Practice*, Springer, Berlin, Germany, 5th edition, 2001.
- [13] I. Khan, H. Mokhtar, and M. Merabti, "A survey of boundary detection algorithms for sensor networks," in *Proceedings of the 9th Annual Postgraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting*, 2008.
- [14] J. Yang and Z. Fei, "HDAR: hole detection and adaptive geographic routing for ad hoc networks," in *Proceedings of the 19th International Conference on Computer Communications and Networks (ICCCN '10)*, pp. 1–6, August 2010.
- [15] B. Kun, G. Najre, T. Kun, and D. Wanlllo, "Neighborhood-based distributed topological hole detection," in *Proceedings of the IET International Conference on Wireless, Mobile and Multimedia Networks (ICWMMN '06)*, pp. 1–4, Hangzhou, China, 2006.
- [16] K. Bi, K. Tu, N. Gu, and W. Dong, "Topological hole detection in sensor networks with cooperative neighbors," in *Proceedings of the International Conference on Systems*, vol. 00, pp. 1–5, October 2006, no. 60533020.
- [17] P. Martins, F. Yan, and L. Decreusefond, "Connectivity-based distributed coverage hole detection in wireless sensor networks," in *IEEE Global Telecommunications Conference (GLOBECOM '11)*, pp. 1–6, 2011.
- [18] S. Zeadally, N. Jabeur, and I. M. Khan, "Hop-based approach for holes and boundary detection in wireless sensor networks," *IET Wireless Sensor Systems*, vol. 2, no. 4, pp. 328–337, 2012.
- [19] W.-C. Chu and K.-F. Su, "Location free boundary recognition in mobile wireless sensor networks with a distributed approach," *Science Direct Computer Network*, vol. 70, pp. 96–112, 2014.
- [20] D. Dong, Y. Liu, and X. Liao, "Fine-grained boundary recognition in wireless ad hoc and sensor networks by topological methods," in *Proceedings of the 10th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pp. 135–144, 2009.
- [21] S. Fekete, A. Kroller, D. Pfisterer, S. Fischer, and C. Buschmann, "Neighborhood-based topology recognition in sensor networks," in *Proceedings of the Algorithmic Aspects of Wireless Sensor Networks (ALGOSENSORS '04)*, vol. 3121 of *Lecture Notes in Computer Science*, pp. 123–136, Springer, 2004.
- [22] P. Corke, R. Peterson, and D. Rus, "Finding holes in sensor networks," in *Proceedings of the IEEE Workshop on Omniscent Space: Robot Control Architecture Geared toward Adapting to Dynamic Environments (ICRA '07)*, April 2007.
- [23] Q. Fang, J. Gao, and L. J. Guibas, "Locating and bypassing holes in sensor networks," *Mobile Networks and Applications*, vol. 11, no. 2, pp. 187–200, 2006.
- [24] M. R. Senouci, A. Mellouk, and K. Assnoute, "Localized movement-assisted sensordeployment algorithm for holedetection and healing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 5, pp. 1267–1277, 2014.
- [25] R. Ghrist and A. Muhammad, "Coverage and hole-detection in sensor networks via homology," in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN '05)*, pp. 254–260, Los Angeles, Calif, USA, April 2005.
- [26] W. Guiling, G. Cao, and T. La Porta, "Movement-assisted sensor deployment," in *Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies*, IEEE, March 2004.
- [27] Y. Wang, J. Gao, and J. S. B. Mitchell, "Boundary recognition in sensor networks by topological methods," in *Proceedings of the 12th Annual International Conference on Mobile Computing and Networking (MOBICOM '06)*, pp. 122–133, ACM, New York, NY, USA, September 2006.
- [28] G. Bldg and S. Funke, "Topological hole detection in wireless sensor networks and its applications," in *Proceedings of the Joint Workshop on Foundations of Mobile Computing (DIALM-POMC '05)*, pp. 44–53, 2005.
- [29] S. Babaie and S. S. Pirahesh, "Hole detection for increasing coverage in wireless sensor network using triangular structure,"

*International Journal of Computer Science Issues*, vol. 9, no. 1, pp. 213–218, 2012.

- [30] A. Kröllner, S. P. Fekete, D. Pfisterer, and S. Fischer, “Deterministic boundary recognition and topology extraction for large sensor networks,” in *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms*, pp. 1000–1009, ACM, January 2006.
- [31] X. Li and D. K. Hunter, “Distributed coordinate-free hole recovery,” in *Proceedings of the IEEE International Conference on Communications Workshops (ICC '08)*, pp. 189–194, IEEE, Beijing, China, May 2008.
- [32] E. Zhao, J. Yao, H. Wang, and Y. Lv, “A coverage hole detection method and improvement scheme in WSNs,” in *Proceedings of the International Conference on Electric Information and Control Engineering (ICEICE '11)*, pp. 985–988, April 2011.
- [33] G. Wang, G. Cao, and T. La Porta, “A bidding protocol for deploying mobile sensors,” in *Proceedings of the 11th IEEE International Conference on Network Protocol (ICNP '03)*, pp. 315–324, November 2003.
- [34] X. Li, D. K. Hunter, and K. Yang, “Distributed coordinate-free hole detection and recovery,” in *Proceedings of the 2006 IEEE GLOBECOM*, San Francisco, Calif, USA, November–December 2006.
- [35] J. Kanno, J. G. Buchart, R. R. Selmic, and V. Phoha, “Detecting coverage holes in wireless sensor networks,” in *Proceedings of the 17th Mediterranean Conference on Control and Automation (MED '09)*, pp. 452–457, Thessaloniki, Greece, June 2009.



**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

