International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2025/29 Volume 12, Issue 4, July – August (2025)

RESEARCH ARTICLE

Coverage Optimization in Wireless Sensor Networks by Detecting and Healing Uncovered Area Through Iterative Sensor Deployment: A Computational Geometry Based Approach

Sharmila Devi
Department of Computer Science and Engineering, Guru Jambheshwar University of Science and Technology, Hisar, India.
shar02chahal@gmail.com

Anju Sangwan
Department of Computer Science and Engineering, Guru Jambheshwar University of Science and Technology, Hisar, India.
anju.sangwan@yahoo.com

Received: 19 April 2025 / Revised: 13 July 2025 / Accepted: 28 July 2025 / Published: 30 August 2025

Abstract – Achievement of optimal area coverage is a significant challenge in Wireless Sensor Networks (WSNs) influenced by factors such as the limited range of sensors, energy constraints and the dynamic nature of environments. Random deployment strategies for sensor nodes often result in coverage holes or redundant nodes. This study primarily focuses on identifying uncovered areas (coverage holes) within a specific region and devising a strategy to ensure maximum coverage with minimal overlap among newly deployed sensor nodes. The proposed work introduces an iterative approach, based on Computational Geometry to enhance coverage by detecting and healing uncovered area. The positions of new sensor nodes are determined by evaluating line intersections in uncovered areas, specifically those passing through the midpoints of uncovered arcs. This methodology ensures the effective placement of new nodes while minimizing redundancy. The process is repeated till the time no further progress in coverage can be attained. Simulation results demonstrate that this method effectively identifies uncovered areas and significantly improves coverage, making it a practical solution for optimizing two-dimensional area coverage in applications such as wireless sensor networks and geometric problem-solving.

Index Terms – Wireless Sensor Networks (WSNs), Computational Geometry, Intersection Points, Coverage Optimization, Coverage Holes, Coverage Region.

1. INTRODUCTION

Wireless Sensor Networks (WSNs) have become a transformative technology, supporting a wide array of applications such as environmental surveillance, emergency response, healthcare services and industrial process control. These systems are composed of sensor nodes distributed across a region, which operate cooperatively to gather and relay data. Random deployment typically involves scattering sensor nodes across the target area from aircraft, drones, or manually in large or inaccessible regions cause uneven placement of sensors. This will cause issues like coverage holes and redundant sensor nodes that do not contribute effectively to monitoring. If a coverage hole remains unaddressed for an extended period, then its size will expand as the sensor nodes at the boundary of the hole are burdened with increased responsibilities. Coverage issue also originates from factors such as the limited sensing range of nodes and energy constraints as discussed in [1], [2], [3].

Addressing this issue is very important to make a WSNs more reliable and effective. Enhancing overall network coverage by focusing on the detection and resolution of coverage holes with minimizing node redundancy, constitutes a primary area of research in WSNs. This is becoming significant due to the growing utilization of WSNs across a wide spectrum of area monitoring applications[4], [5]. Some applications, such as military area exploration, have no scope for coverage hole tolerance. To enhance coverage in random deployment scenarios, strategies utilizing Computational Geometry and optimization techniques increase the reliability and practicality of WSNs for real-world applications. Various studies [6], [7], [8] utilizing Delaunay Triangulation for sensor node deployment minimizes coverage holes and ensures an energy-efficient network.

RESEARCH ARTICLE

1.1. Identification of Coverage Holes

Depending on the nature of the data utilized, sensor networks typically employ three main approaches to detect coverage gaps: geographical, topological, and statistical [9].

- Geographical Method

This technique relies on knowing the precise positions of sensor nodes, which can be obtained through hardware tools like scanners or by sharing node data across the network. It assumes that node locations are pre-determined and uses geometric constructs such as Voronoi diagrams, Delaunay triangulation etc. to pinpoint and outline coverage gaps.

- Topological Method

This method depends only on the network's connectivity structure. It does not require any assumptions about node positions or additional location data, making it more flexible in certain scenarios.

- Statistical Method

This method relies on the statistical distribution of sensor nodes, commonly represented through probabilistic models. While it does not need specific node data, it only requires a high density of sensors to be effective in detecting coverage holes.

1.2. Healing of Coverage Holes

Coverage hole healing techniques are designed to eliminate or reduce gaps in network coverage. Such holes can result from node malfunctions, energy depletion, environmental disruptions, or the inherent randomness in node placement. Three main approaches are used to minimize or remove coverage holes [10] , [11]:

- Computational Geometry-Based Approach

Voronoi diagrams and Delaunay triangulation are commonly used to identify regions with no coverage. Once a coverage hole is detected, existing or new sensor nodes are repositioned to fill the coverage holes. These methods are highly precise and effective in structured environments.

- Topology-Based Approaches

Topology-based approaches focus on the connectivity structure of the network rather than the physical positions of the nodes. These methods analyse how nodes are linked to one another to infer the presence of coverage holes.

Techniques such as boundary node detection and graph-based algorithms are used to identify areas with weak or missing connectivity. Once a hole is detected, the network can respond by activating redundant nodes or reconfiguring the roles of existing ones to restore coverage.

- Virtual Force-Based Approaches

Virtual force-based methods draw inspiration from physical laws, modeling sensor nodes as particles subjected to both attractive and repulsive forces. These virtual forces guide the movement of mobile nodes toward uncovered areas in the network.

The estimation of total uncovered area also plays a significant role for selection of appropriate hole recovery algorithm which optimizes the performance, reliability and energy efficiency of WSNs [13].

1.3. Motivation

This paper aims to present a straightforward and efficient method for detecting and recovering uncovered regions, while minimizing sensing overlap among newly deployed sensor nodes. An iterative algorithm has been developed that applies computational geometry to improve coverage. Proposed approach iteratively identifies and fills uncovered regions, ensuring coverage improvement in each iteration. The use of intersection points between overlapping circular regions is used to create arcs on the boundary of sensing regions and then utilised to analyse uncovered areas adds a level of precision that is typically lacking in other methods. Determining new sensor node's positions in uncovered region at the intersection points of lines which are passing from the centres of uncovered arcs. To avoid overlapping in new sensors, only limited number of locations are chosen based on their distance to deploy new nodes. This iterative process continues until the algorithm can no longer achieve further coverage improvements, ultimately leading to a significant increase in overall coverage. The inspiration behind this research is to provide a scalable and efficient solution that can be applied in various real-world scenarios, improving the reliability and cost-effectiveness of monitoring and surveillance systems.

1.4. Contributions

This research presents several notable contributions, listed below:

- Iterative coverage optimization: Proposing an innovative, iterative approach based on Computational Geometry to systematically enhance area coverage while minimizing overlapping.
- Intersection-based analysis: A novel mechanism is implemented to detect circular region intersections, classify arcs and identify "uncovered arcs" using computational geometric calculations. This ensures precise identification of regions that require additional coverage.

RESEARCH ARTICLE

- Adaptive sensor nodes placement: Additional sensor nodes are tactically deployed at the intersections of lines connecting the midpoints of uncovered arcs.
- Simulations results Visualization: The methodology includes clear visualization of sensor nodes area coverage, uncovered areas, and newly added sensor nodes at each iteration. This aids in understanding and analysing the iterative improvements in coverage.

These contributions form the backbone of a systematic and efficient approach to tackle sensor's coverage optimization problems, making this research applicable to various fields requiring spatial resource allocation.

The remainder of this paper is structured as follows: Section 2 presents a review of related work. Section 3 outlines the system model and the algorithms employed for identifying and addressing coverage holes in WSNs. Section 4 details the simulation environment and analyzes the proposed approach. Finally, Section 5 concludes the paper with key observations and insights.

2. RELATED WORK

Detection and remediation of coverage holes have been central topics of investigation in WSNs, given their direct impact on overall network performance. Identifying coverage holes is essential to ensure complete sensing and data reliability across a region of interest. A variety of algorithms and frameworks have been developed to detect and address coverage holes, employing approaches that span from static node evaluation to mobility-driven techniques. While extensive literature exists in the broader domain of WSNs which presents distinct challenges due to their unique deployment environments and constrained communication conditions. This section reviews several prior studies focused on the detection and mitigation of coverage holes. Authors in [14] proposed a decentralized algorithm for detecting and healing coverage holes, which operates without relying on coordinate information. They used the concept of boundary critical points for coverage holes detection that does not depend on the locations of sensor nodes. For coverage holes healing the concepts of perpendicular bisectors and boundary critical points are used to provide improved coverage. For heterogeneous WSNs a distributed scheme is used in [15] to detect coverage holes. This scheme utilized the concepts of nodes mobility and divided the process into three stages: initialization, detection of coverage holes and recovery. During initialization stage, sensor nodes determine information about their neighbor nodes and in second phase, each node determines whether it lies on the boundary of a coverage hole or not. The recovery phase further subdivided into two modes: Pro-active mode (in case hole is because energy depletion of sensor nodes) and Re-active mode (in case hole is because random deployment). A coordinate-independent approach based on Delaunay triangulation is used in [16] to achieve full coverage. The method enables distributed detection of coverage holes and identification of optimal paths for their remediation. In this approach, coverage holes healing process did not require accurate node location. It is integrated with a cooperative movement mechanism that prevents the formation of new holes during node movements involved in the healing process. A decentralized method consisting of two distinct phases is used in [17] where each node independently evaluates its status to determine if it is a stuck node, thereby assisting in the identification of coverage holes. Once these gaps are recognized, alternative data transmission paths are constructed to route along the holes.

The problem addressed in [18] introduces an approach which is designed to detect boundaries of coverage holes within a wireless network. This approach is based on communication graph, which is capable of accurately identifying nodes near hole boundaries. Approach defined in [19] is also focuses on detection and repairing of coverage holes in WSNs. Given method dynamically analyzes the network topology, identified boundary nodes of empty region, and used clustering strategies to address coverage holes. Coverage holes are repaired by activating passive nodes and designating them as cluster nodes within the empty areas. Proposed approach also improves connectivity and communication within the network. A novel approach is introduced in [20] to address coverage holes in urban cellular networks. It utilized unmanned aerial vehicles equipped with reinforcement learning algorithms to detect coverage holes efficiently. Unmanned aerial vehicles relied on stored building maps, their current locations and measured signal strengths to dynamically update their understanding of the signal environment. This method reduces the need for traditional drive tests and human labor while ensuring effective detection of coverage holes. Authors in [21] introduced a method to detect and repair sensing holes caused by damaged or energy-depleted nodes in WSNs. It utilizes a network-centric hybrid mobility scenario to address these issues efficiently. Proposed method is compared with other approaches: This includes the Enhanced Virtual Forces Algorithm (with boundary forces), the Coverage-Aware Sensor Automation Protocol, and the Sensor Self-Organizing Algorithm. By setting node density thresholds for different scenarios, the method demonstrated superior recovery efficiency, reduced computational costs, and minimized energy depletion compared to other techniques. The proposed approach prolongs network lifetime by offloading processing and mobility overhead to the external network. Authors in [22] defined a method which is focuses on addressing coverage holes in WSNs which are caused due to node failures, energy depletion, or environmental obstacles. They introduced novel strategies leveraging advanced deep learning techniques for analyzing and addressing coverage holes.

RESEARCH ARTICLE

Additionally, an enhanced Social Group Optimization (SGO) algorithm was employed to quantify coverage hole areas or uncovered areas within the region of interest, while a decentralized game-theoretic optimization method was used to detect the boundaries of these holes. This comprehensive approach enabled efficient resource allocation for effective coverage restoration. Given technique achieves reductions in hole detection time and required fewer patching sensors. A bio-inspired approach is used in [23] to address the issue of coverage holes in WSNs. The approach employs a swarm intelligence-based algorithm for detecting and healing coverage holes, drawing inspiration from the behavior of platelets in the blood clotting process. These agents relied on native and comparative information to navigate toward the holes and accordingly deploy themselves at optimal positions along the hole boundaries. Proposed method demonstrated superior scalability and flexibility, efficiently manages battery levels and adapts to various hole sizes. Experimental results highlighted its robustness against agent failures and perception corruption.

A distributed hole detection and repair algorithm is introduced in [24]. This algorithm dynamically detects coverage hole and calculates its position and size. It utilized already deployed sensor nodes and avoid the need for additional sensor nodes. These nodes coordinate their movements to restore the void area without disrupting existing coverage or connectivity. Authors in [25] proposed a strategy to optimize the topology of WSNs for enhanced coverage. They introduced a technique centered on the wolf pack algorithm, which improves the network's performance. Authors in [26] proposed a chord-based hole detection method capable of identifying both closed and open coverage holes within the region of interest. This method effectively pinpointed the boundaries of coverage holes, enhancing the network's ability to monitor its environment. This approach minimized sensing coverage overlap while ensuring complete coverage of network using the fewest possible sensor nodes. In [27], authors presented a distributed Voronoi based cooperation method in which nodes cooperates with each other to detect a coverage hole. Sensor nodes generate localized Voronoi diagrams as part of the coverage evaluation process in this method. Mobility of sensor nodes is also restricted to evade the generation of novel coverage holes. Healing of coverage hole is the responsibility of the cooperating nodes. Algorithm based on virtual edge-based hole detection is proposed by authors in [28] to sense coverage holes. They used Voronoi Diagram, a standard geometric tool to detect coverage holes. Border nodes were discovered with the help of distance calculation between nodes. If the calculated distance exceeds the detecting range, the corresponding node is marked as a boundary node. These boundary nodes are then used as input to identify the shape of the coverage hole. The algorithm proposed effectively determines the outline of coverage hole but it is only applicable to small areas. Use of Delaunay Triangulation in [29] is suggested to discover and patch-up coverage holes. Authors considered the hybrid network of sensors, including both static and dynamic sensor nodes. By estimating the coverage holes of static sensors, they calculated the best position for assisted sensors to heal the coverage holes. A simpler way is provided to heal coverage holes and give optimal solution for hole detection and healing problem. Authors in [30] introduced a collaborative coverage enhancing algorithm aimed at maximizing area coverage. They used Voronoi Diagrams to identify coverage holes by utilizing static nodes. The given area is partitioned into several cells, which are subsequently divided into multiple triangles. Then moveable nodes are used to recuperate these coverage holes. This method accurately determines the number of holes in a specific region. Authors in [31] introduced a method for detecting coverage holes in different triangles by using static nodes. Heal coverage hole algorithm is proposed to recover coverage holes by using mobile nodes. Delaunay Triangulation is constructed by static nodes after which optimal position within each triangle is determined. To address coverage holes and devise a deployment rule for the assisted nodes, range of all nodes is calculated. In paper [32], coverage holes discovery algorithm is presented which is based on Voronoi Diagram. Given approach identified the presence of coverage holes by analyzing the edges of the Voronoi Diagram. Computation is performed in distributed manner because Voronoi Diagram is used on the sink node and other computations takes place on other common nodes.

Some drawbacks in existing coverage hole detection and healing approaches:

- Centralized detection methods often become inefficient or fail in large-scale networks, due to excessive communication overhead.
- Certain algorithms struggle to accurately identify the exact boundaries of coverage holes.
- Healing algorithms may reposition nodes without accounting for their remaining energy, resulting in new coverage holes soon after deployment.
- Some healing strategies deploy additional sensor nodes without optimal placement, leading to unnecessary overlap and resource wastage.

By using the concept of intersection points of overlapping sensor nodes, the proposed approach finds the exact uncovered area boundaries. New sensor nodes are placed only on the specific intersection points in uncovered regions, which prevents overlapping of newly placed sensor nodes. Strategically placing nodes in precisely identified uncovered areas results in lower deployment cost and better resource management.

RESEARCH ARTICLE

3. PROPOSED COVERAGE HOLES DETECTION AND HEALING APPROACH

3.1. System Model

This section outlines several assumptions to provide clarity on the proposed methodology. All sensor nodes S_n= {S_1, S_2, S_3, …… S_n} are deployed randomly in two-dimensional area. These nodes are homogeneous in nature i.e. with circular sensing region of radius r. The locations in respect of all nodes are well-known with the help of existing GPS (Global Positioning System) technology. No two sensor nodes are deployed at the same location. All sensor nodes are stationary. Euclidean Distance ($d_{i,j}$) between two sensor nodes S_i (x_i, y_i) and S_j (x_j, y_j) is described in equation (1):

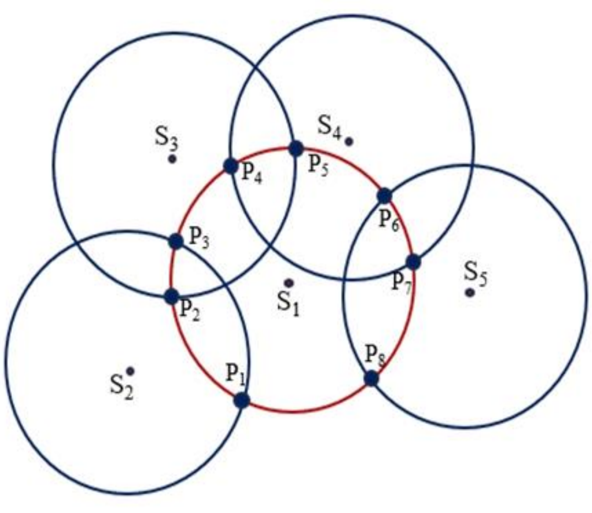
$$d_{i,j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \qquad (1)$$

The proposed method is structured into two distinct phases: coverage hole detection and healing of uncovered area.

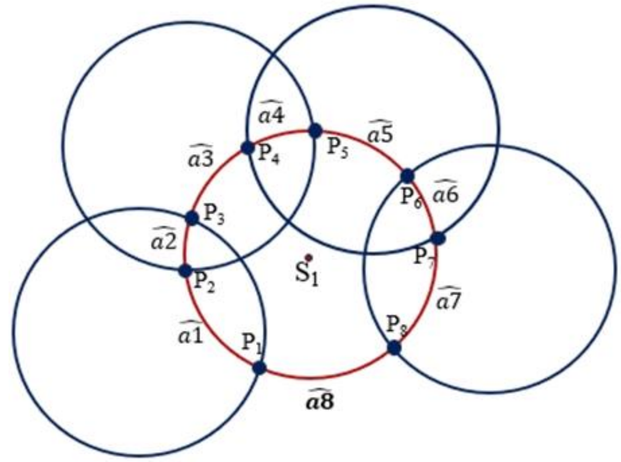
3.2. Coverage Hole Detection

Definition: A circular coverage region of a sensor node, with radius r, is a part of hole if it contains at least one uncovered arc.

Initially, a very less number of sensor nodes are deployed to avoid redundancy. To identify uncovered regions, the uncovered arc of each sensor node is determined. To achieve this, it is pertinent to recognize the neighboring nodes of each node within the grid. If a sensor node is secluded i.e., it has no neighboring nodes, it is not considered for detecting coverage holes. Based on ($d_{i,j}$<2r) presented in [33], each neighboring node must intersect with each other on two locations i.e. corresponding two intersection points.



(a)



(b)

Figure 1 Presenting Intersection Points in (a) and Corresponding Arcs in (b) of Node S_1 with its Neighboring Nodes

Figure 1(a) and 1(b) present intersection points and corresponding arcs on node S_1, where node S_1 is reference node and S_2, S_3, S_4, S_5 are its neighbouring nodes. Node S_1 is intersected by these neighbouring nodes at points P_1, P_2, P_3, P_4, P_5, P_6, P_7 and P_8. Intersection points coordinates can be calculated by considering sensor node's coverage region as a spherical area with radius r. To understand the concepts of intersection points, two circular regions $S_1(x_1, y_1)$ and $S_2(x_2, y_2)$ with radius r1 and r2 are considered. These circles are intersecting at points P and Q as shown in Figure 2.

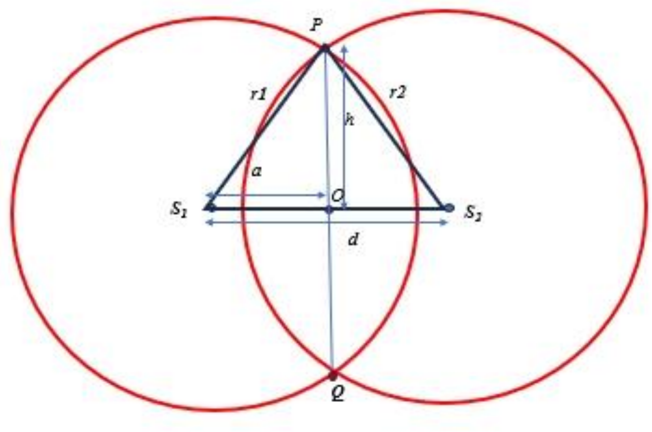


Figure 2 Representing Points of Intersection P and Q of Nodes S1 and S2

Equations (2), (3) represent the equations of two circles:

$$(x - x_1)^2 + (y - y_1)^2 = {r_1}^2 \qquad (2)$$

International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2025/29 Volume 12, Issue 4, July – August (2025)

RESEARCH ARTICLE

$$(x-x_2)^2+(y-y_2)^2={r_2}^2 \quad (3)$$

The distance (d) between two circles S_1 and S_2 is

$$d=\sqrt{(x_2-x_1)^2+(y_2-y_1)^2} \quad (4)$$

Two circles can intersect at two points only when d<2r where d is represented in equation (4). After joining two intersecting points P and Q, it is forming common chord PQ. PQ is the perpendicular bisector of line S_1S_2, which intersects at point O (x_o, y_o).

So,

$$O=\left(\frac{x_1+x_2}{2},\frac{y_1+y_2}{2}\right) \quad (5)$$

Distance between S_1 and O is a

$$a=\frac{d}{2} \quad (6)$$

PS_1O is a right-angle triangle, so height h is

$$h=\sqrt{{r_1}^2-a^2} \quad (7)$$

A perpendicular line is drawn from point O as defined in equation (5) in the direction orthogonal to S_1S_2. The values of a and h are determined using equations (6) and (7), respectively. If S_1S_2 has direction $((x_2-x_1),(y_2-y_1))$ then perpendicular directions are $(-(y_2-y_1),(x_2-x_1))$.

$$P=\left(O_x+h\frac{-(y_2-y_1)}{d},O_y+h\frac{-(x_2-x_1)}{d}\right) \quad (8)$$

$$Q=\left(O_x-h\frac{-(y_2-y_1)}{d},O_y-h\frac{-(x_2-x_1)}{d}\right) \quad (9)$$

Equations (8) and (9) are utilized to accurately determine the points of intersection, labeled P and Q within the uncovered regions. In this approach, the coordinates of all intersection points are determined for every intersected node. These intersection points on the circle's periphery, result in the formation of arcs.

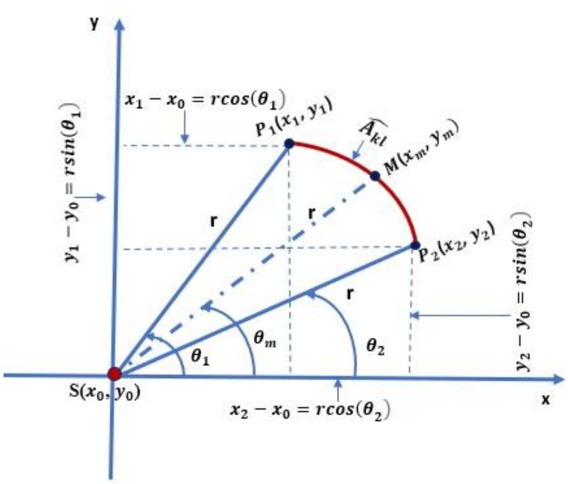


Figure 3 Representing Arc's ($\widehat{A_{kl}}$)Mid-Point M Coordinates

The rectangular coordinates of P_1 with respect to S are (x_1-x_0) and (y_1-y_0) which are related to its polar coordinates (r,θ_1). So,

$$x_1-x_0=rcos(\theta_1) \quad (10)$$

$$y_1-y_0=rsin(\theta_1) \quad (11)$$

Then,

$$\theta_1=tan^{-1}\left(\frac{y_1-y_0}{x_1-x_0}\right) \quad (12)$$

Also,

$$x_2-x_0=rcos(\theta_2) \quad (13)$$

$$y_2-y_0=rsin(\theta_2) \quad (14)$$

Then

$$\theta_2=tan^{-1}\left(\frac{y_2-y_0}{x_2-x_0}\right) \quad (15)$$

Mid-point angle is

$$\theta_m=\frac{\theta_1+\theta_2}{2} \quad (16)$$

Mid-point coordinates

$$x_m=x_0+rcos(\theta_m) \quad (17)$$

$$y_m=y_0+rsin(\theta_m) \quad (18)$$

Equations (10)-(18) are used to determine the mid-point's coordinates of arc $\widehat{A_{kl}}$, which is used to determine whether the arc is covered or not. An arc is considered uncovered if its mid-point is not covered by any neighbouring node. In this way uncovered arcs for each node in the given network are identified.

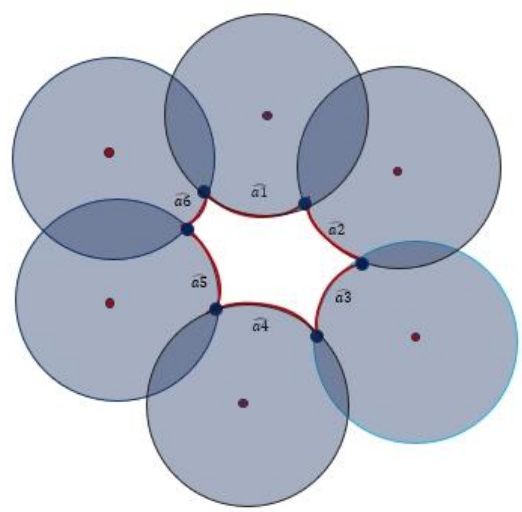


Figure 4 Representing a Boundeded Coverage Hole

These uncovered arcs forms coverage holes in the network. If two uncovered arcs are sharing the same intersection points, it

RESEARCH ARTICLE

will make boundary of same coverage holes. If the first and last arc of a coverage hole is sharing the same intersection point then that hole is a bounded hole.

Arcs $\widehat{a1}$, $\widehat{a2}$, $\widehat{a3}$, $\widehat{a4}$, $\widehat{a4}$, $\widehat{a5}$ and $\widehat{a6}$ form the boundary of the coverage hole, as illustrated in Figure 4. This hole is covered hole, as arc $\widehat{a1}$ and arc $\widehat{a6}$ sharing common intersection point. This allows us to determine any coverage hole that exists at any given time. Algorithm 1 discusses the entire process of detection of uncovered arcs.

Input:

1. Sensor node's location coordinates $S_i (x_i, y_i)$
2. Sensing Radius R_s
3. Number of initially deployed sensing nodes N

Steps:

1. Initially NB_i =Null // NB_i is the set of Neighboring Nodes
2. For each i, i ϵ N (where N is the total number of nodes)
3. For each $S_i (x_i, y_i)$

{

For each $S_j (x_j, y_j)$ except i=j, j ϵ N

3.1 {

$d_{i,j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$ // $d_{i,j}$ signifies the Euclidean distance separating nodes S_i and

If $d_{i,j} < 2R_s$

{

Add S_j to NB_i

$NB_i = \{S_j\}$, j ϵ N

}

}

3.2 Count=0

For each NB_i

{

for each S_j ϵ NB_i

{

Find intersection points of S_i and S_j using equation (8) and (9)

Add intersection points P_i and P_j to set I_i

Count=Count+2

}

}

3.3 For each P_k ϵ I_i Where (k=1,2,3……Count)

For each (k=1; k≤ Count; k++)

{

l=(k<Count)? k+1:1

{

Find A_{kl} //find arcs on sensor S_i generated due to intersection points

Add A_{kl} to set A_i // A_i is the set of arcs on S_i

Find mid-point on arc A_{kl}

Mid $A_{kl} = \frac{(P_{k1} + P_{k2})}{2}, \frac{(P_{l1} + P_{l2})}{2}$

Add mid A_{kl} to the set M_i

$M_i = \{\text{mid } A_{kl}\}$ where i ϵ arc A_{kl}

}

}

3.4 For each mid-point $M_{kl} (k_i, l_i)$

{

For each (i=1; i≤ Count; i++)

{

For (j=1; j≤ NB; j++)

{

Find $d = \sqrt{(x_j - k_i)^2 + (y_j - l_i)^2}$

If d > R_s then

Arc A_{kl} is a part of coverage hole

Add arc A_{kl} to the set of uncovered arcs

Uncov= $\{A_{kl}\}$

Else

Arc A_{kl} is not included in coverage hole

}

}

}

}

Output:

Set of uncovered arcs

Algorithm 1 Coverage Holes Detection Algorithm

RESEARCH ARTICLE

3.3. Healing of Uncovered Area

After identifying uncovered arcs, this phase outlines a strategic method for deploying new sensor nodes to enhance coverage. Lines are drawn from the centre and midpoints of a sensor node where uncovered arcs are present. The locations of all intersection points within the uncovered area are determined. These intersection points are used to identify precise locations for deploying new sensor nodes. New sensor nodes are strategically deployed to ensure they do not fall within the sensing range of other newly added nodes. As shown in Figure 5, a single sensor node is placed to cover three intersection points. This is because, once a node is deployed at one of these points, the other two fall within its coverage area. This approach minimizes sensor overlap, thereby maximizing overall area coverage while reducing redundancy.

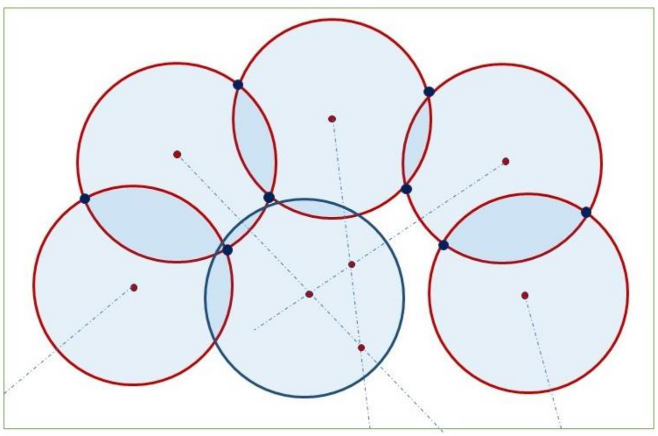


Figure 5 New Sensor Node (Represented with Blue Circle) Deployment on the Line Intersection Point

Input:

1. Sensor node's location coordinates S_i (x_i, y_i)
2. Sensing radius r
3. Total nodes: Number of sensing nodes initially deployed(N)
4. Uncovered arcs: Set of uncovered arcs
5. Midpoints: Set of mid-points corresponding to uncovered arcs

Steps:

1. While Uncovered arcs is not empty do
2. for each arc in Uncovered arcs do
3. Compute line equation using:

$$y - y_i = \frac{y_j - y_i}{x_j - x_i} x - x_i$$

4. Store all intersection points within uncovered area
5. end for
6. for each intersection point in uncovered area do
7. if intersection point is within sensing region of a newly deployed node, then
8. skip deployment at this point
9. else
10. Deploy sensor node at this intersection point
11. Total nodes = Total nodes + 1
12. end if
13. end for
14. Update Uncovered arcs based on newly deployed sensor nodes
15. End else

Output: Number of new sensor nodes deployed

Algorithm 2 Healing of Coverage Holes (Uncovered Area)

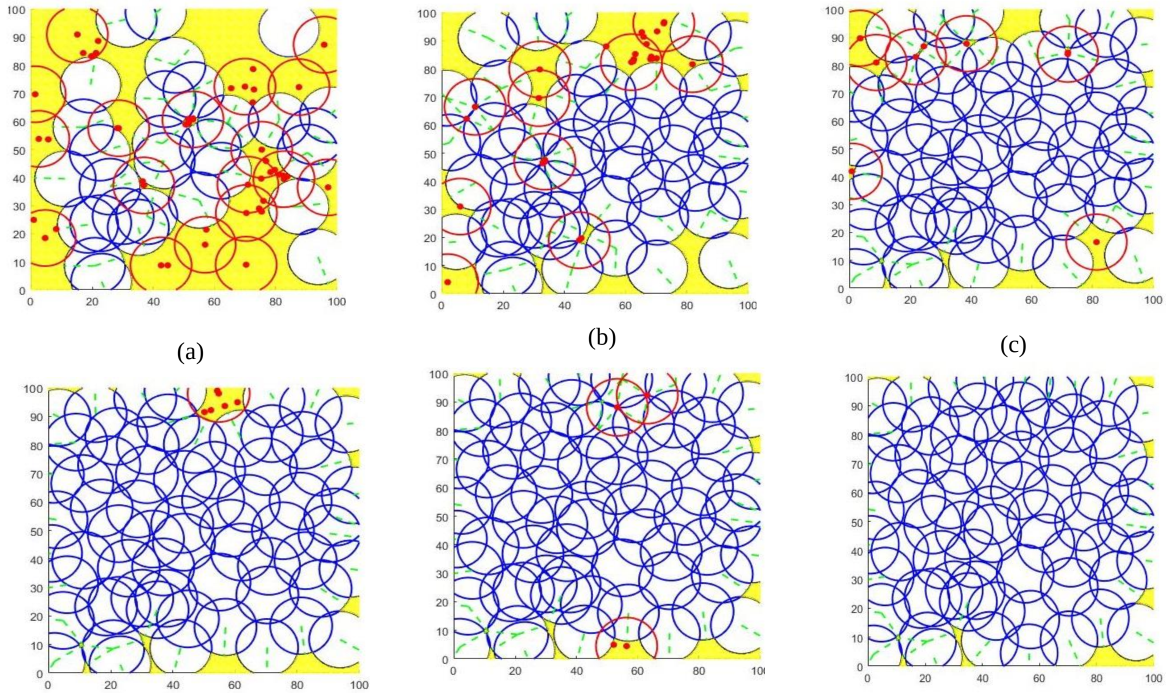
Algorithm 2 represents the process of new sensor node's deployment with minimum overlapping in uncovered area. Uncovered arcs are then analyzed to determine optimal positions for placing additional sensor nodes. New sensor nodes are positioned only when the intersection points of lines drawn through the midpoints of uncovered arcs lie within the uncovered regions. To prevent overlapping, no new sensor nodes are placed if the intersection points lie within the coverage of already deployed nodes. This process is repeated iteratively until the uncovered area no longer requires additional sensor nodes. Proposed iterative approach provides efficient and precise method to position sensor nodes in a specific network to maximize coverage while minimizing redundancy. By identifying uncovered arcs, calculating intersection points, and strategically deploying sensor nodes based on coverage criteria, the algorithm ensures optimal use of resources.

4. SIMULATION AND ANALYSIS

To validate the proposed method, simulations have been performed on MATLAB R2018a to identify uncovered area and enhance coverage by strategically adding new sensor nodes as showcased in Figure 6. Table 1 outlines the parameters utilized for the simulation. The iterative enhancement approach is employed to maximize the area coverage, with the process terminating once no additional sensor nodes could be placed within the target area. The impact of varying simulation parameters like the number of sensor nodes, sensing radius, and coverage proportion (%) is analyzed in terms of area coverage. Boundary values are excluded from consideration, which may leave certain areas near the boundaries uncovered.

Table 1 Simulation Parameters

Number of sensor nodes initially	30
Radius	10m
Area	100m*100m
Number of iterations	6
Dimension	2D



(a) (b) (c)

(d) (e) (f)

Figure 6 Optimized Coverage Enhancement with Proposed Algorithm

The effect of area coverage after each iteration is depicted in Figure 7. Initially 30 random senor nodes of radius 10m are deployed in an area 100m*100m.

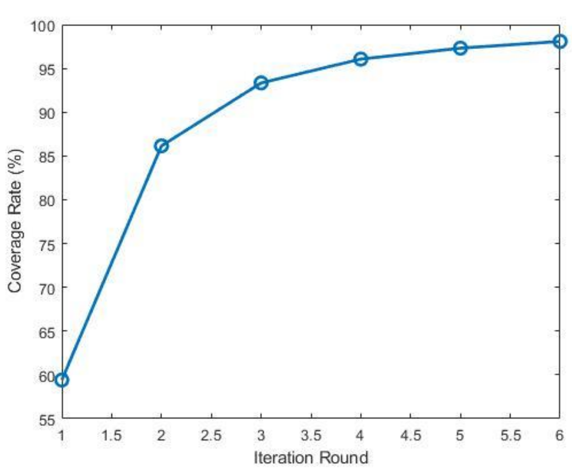


Figure 6 Coverage Rate Increases after Every Iteration

Table 2 Coverage Rate After Each Iteration

Iteration Number	Total Area Coverage (%)
1	59.41
2	86.15
3	93.34
4	96.06
5	97.31
6	98.08

In the first iteration for 30 sensor nodes only 59.41% area is covered. After applying proposed approach, it is increased to 86.15% in the next iteration which is indicating significant improvement. New sensor nodes are added with minimum overlapping at every stage of the process. At 3rd iteration coverage rate is 93.34% and after 6th iteration it is 98.08% as shown in Table 2.

International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2025/29 Volume 12, Issue 4, July – August (2025)

RESEARCH ARTICLE

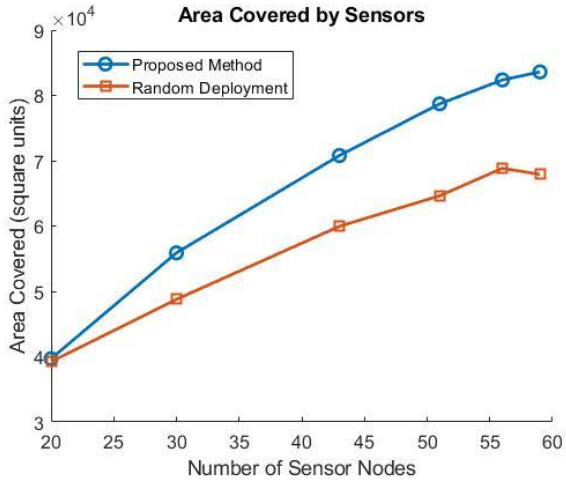


Figure 7 Area Coverage(Proposed Method/Random Deployment) for Different Number of Sensor Nodes

Table 3 Total Area Coverage Effect of Different Sensor Nodes in Random Node Deployment and the Proposed Method

Number of Sensor Nodes	Total Area Covered by Random Deployment (m^2)	Total Area Covered by Proposed Method (m^2)
20	39224.52 m^2	39682.80 m^2
30	48772.62 m^2	55885.77 m^2
43	59948.55 m^2	70794.72 m^2
51	64628.46 m^2	78704.10 m^2
56	68838.93 m^2	82364.49 m^2
59	67903.20 m^2	83591.10 m^2

From Figure 8, proposed algorithm significantly improves area coverage as compare to random nodes deployment in WSNs. Initially the area coverage is similar for 20 random sensor nodes but when number of sensor nodes increases to cover maximum area there is significant difference between total area covered by random deployment and area covered by using proposed algorithm also also shown in Table 3.

For a count of 51 sensor nodes area covered in random deployment method is 64628.46 m^2, but using proposed algorithm total area covered is 78704.10 m^2. By deploying 59 sensor nodes area covered is 83591.10 m^2 while in case of random deployment it is 67903.20 m^2, which is showing great difference.

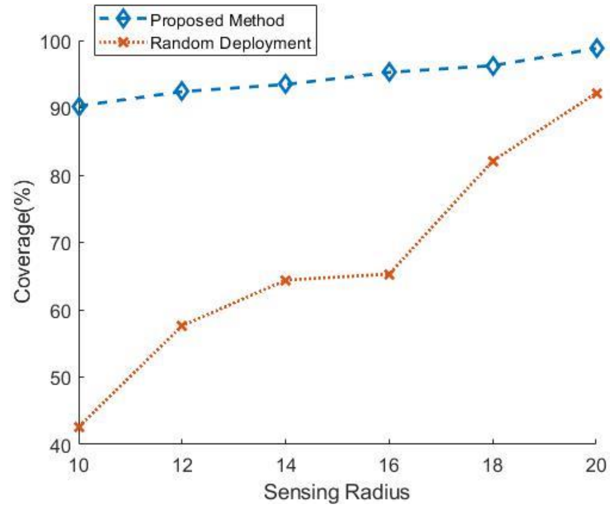


Figure 8 Comparison of Coverage Rate for Varying Radius

Table 4 Comparison of Coverage Rate

Sensing Radius(m)	Coverage (%) by Random Deployment	Coverage (%) by Proposed Method
10m	42.60	90.26
12m	57.62	92.40
14m	64.40	93.46
16m	65.30	95.26
18m	82.10	96.26
20m	92.20	98.80

The graph values as shown in Figure 9 represents the area coverage (%) achieved through proposed method and random deployment method for various sensing radii. At smaller sensing radii 10m and 12m proposed method achieves 90.26% and 92.40% coverage, showing high efficiency even with a limited sensing radius. While in case of random method it only covers 42.60% and 57.62%, significantly lagging the proposed method. This indicates that random placement struggles to optimize coverage for sensor with smaller sensing radius. At moderate sensing radii 14m and 16m proposed method improves coverage consistently, reaching 93.46% and 95.26%. Where random deployment method coverage increases to 64.40% and 65.30%, but the growth rate slows compared to the earlier radii. At larger sensing radii 18m and 20m proposed method achieved optimal coverage i.e. 96.26% and 98.80%. While random method's coverage improves to 82.10% and 92.20% respectively, the gap with the proposed method persists as shown in Table 4. The proposed method outperforms the random approach across all sensing radii, with higher and more consistent coverage. All these results prove its effectiveness in utilizing the available sensing radius.

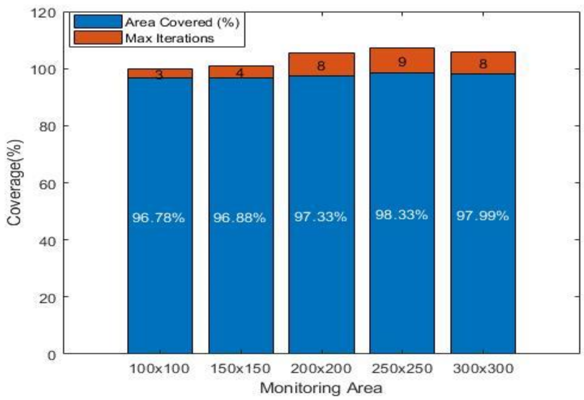


Figure 9 Representing Coverage Optimization in Minimum Iterations in Case of Changing Monitoring Area Size(m×m)

Table 5 Representing Area Coverage After Each Iteration

Monitoring Area(m×m)	Maximum Area Coverage (%)	Number of Iteration
100m×100m	96.78	3
150m×150m	96.88	4
200m×200m	97.33	8
250m×250m	98.33	9
300m×300m	97.99	8

To evaluate area coverage effectiveness at different scales, the study considered five monitoring regions with dimensions of 100 meters × 100 meters, 150 meters × 150 meters, 200 meters × 200 meters, 250 meters × 250 meters and 300 meters × 300 meters. Within each area, initially 30 random sensor nodes of radius 20 m are deployed, then proposed coverage optimization strategy is applicable to organize new sensor nodes in the uncovered area. After each deployment, the study collects data on maximum iteration required to cover maximum area coverage (%). In smaller monitoring area of 100 meters × 100 meters, only 3 iterations are required to achieve 96.78% area coverage. As the size of target area increases the number of iterations upturns as shown in Figure 10. In case of 300 meters × 300 meters area there are 8 iterations to cover 97.99 % area. Because of random deployment of initial sensor nodes, sometimes overall iterations tend to be more in smaller size target area. In case of 250 meters × 250 meters area to achieve 98.33% there is one more iteration as compare to iterations required in 300 meters × 300 meters area as highlighted in Table 5.

5. CONCLUSION

This research provides an effective solution to address the challenge of achieving optimal area coverage in WSNs, which is often hindered by coverage holes and redundant node deployments. By leveraging Computational Geometry, the proposed iterative approach ensures a systematic identification of uncovered areas (coverage holes) and strategic placement of new sensor nodes. Proposed approach utilizes the intersection points of overlapping sensor nodes to precisely determine the boundaries of uncovered areas. New sensor nodes are deployed solely at the intersection points within uncovered regions, effectively eliminating overlap and reducing a limitation often seen in existing approaches. This targeted placement strategy reduces deployment costs and ensures more efficient resource management. Simulation results validate the effectiveness of this approach, demonstrating substantial improvement in network coverage while keeping resource utilization efficient. This study not only addresses the coverage holes in random deployment strategies but also introduces a scalable and precise algorithm for enhancing area coverage in dynamic environments.

REFERENCES

[1] X. Yu, M. Xu, L. Cheng, and N. Hu, “A novel coverage holes detection and holes recovery algorithm in wireless sensor networks,” in the 27th Chinese Control and Decision Conference (2015 CCDC), IEEE, 2015, pp. 3640–3644. Accessed: Jan. 23, 2025. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7162556/.

[2] J. Feng and H. Chen, “Healing coverage holes for big data collection in large-scale wireless sensor networks,” Mob. Netw. Appl., vol. 24, no. 6, pp. 1975–1984, 2019.

[3] S. M. Koriem and M. A. Bayoumi, “Detecting and measuring holes in wireless sensor network,” J. King Saud Univ.-Comput. Inf. Sci., vol. 32, no. 8, pp. 909–916, 2020, doi: 10.1016/j.jksuci.2018.08.001.

[4] Das, S., & Debbarma, M. K. (2018). A survey on coverage problems in wireless sensor network based on monitored region. In Advances in Data and Information Sciences: Proceedings of ICDIS 2017, Volume 2 (pp. 349-359). Singapore: Springer Singapore.

[5] P. Sharma and R. P. Singh, “Coverage hole identification & healing in wireless underground sensor networks,” Meas. Sens., vol. 24, p. 100540, 2022, doi: 10.1016/j.measen.2022.100540.

[6] C.-H. Wu, K.-C. Lee, and Y.-C. Chung, “A Delaunay triangulation based method for wireless sensor network deployment,” Comput. Commun., vol. 30, no. 14–15, pp. 2744–2752, 2007, doi: 10.1016/j.comcom.2007.05.017.

[7] F. Jing, W. Qiong, and H. JunFeng, “Optimal deployment of wireless mesh sensor networks based on Delaunay triangulations,” in 2010 International Conference on Information, Networking and Automation (ICINA), IEEE, 2010, pp. V1-370. Accessed: Mar. 22, 2025, doi:10.1109/icina.2010.5636371.

[8] R. Nespolo, L. A. Neves, G. Roberto, M. Ribeiro, M. Z. Do Nascimento, and A. Cansian, “Method based on triangulation for sensor deployment on 3D surfaces,” in 2016 29th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), Ieee, 2016, pp. 136–143, doi:10.1109/sibgrapi.2016.027.

[9] A. Soundarya and V. Santhi, “An efficient algorithm for coverage hole detection and healing in wireless sensor networks,” in 2017 1st International conference on electronics, materials engineering and nano-technology (IEMENTech), IEEE, 2017, pp. 1–5, doi:10.1109/iementech.2017.8076995.

[10] C. So-In, T. G. Nguyen, and N. G. Nguyen, “An efficient coverage hole-healing algorithm for area-coverage improvements in mobile sensor networks,” Peer--Peer Netw. Appl., vol. 12, no. 3, pp. 541–552, May 2019, doi: 10.1007/s12083-018-0675-8.
[11] R. Beghdad and A. Lamraoui, “Boundary and holes recognition in wireless sensor networks,” J. Innov. Digit. Ecosyst., vol. 3, no. 1, pp. 1–14, 2016, doi: 10.1016/j.jides.2016.04.001.
[12] S. Das and M. Kanti DebBarma, “Computational geometry based coverage hole-detection and hole-area estimation in wireless sensor network,” J. High Speed Netw., vol. 24, no. 4, pp. 281–296, 2018, doi:10.3233/jhs-180595.
[13] H. Zhang and J. C. Hou, “Maintaining sensing coverage and connectivity in large sensor networks.,” Ad Hoc Sens Wirel Netw., vol. 1, no. 1–2, pp. 89–124, 2005.
[14] Z. Kang, H. Yu, and Q. Xiong, “Detection and recovery of coverage holes in wireless sensor networks,” J. Netw., vol. 8, no. 4, p. 822, 2013.
[15] A. M. Khedr, W. Osamy, and A. Salim, “Distributed coverage hole detection and recovery scheme for heterogeneous wireless sensor networks,” Comput. Commun., vol. 124, pp. 61–75, 2018, doi: 10.1016/j.comcom.2018.04.002.
[16] C. Qiu and H. Shen, “A delaunay-based coordinate-free mechanism for full coverage in wireless sensor networks,” IEEE Trans. Parallel Distrib. Syst., vol. 25, no. 4, pp. 828–839, 2013, doi: 10.1109/tpds.2013.134.
[17] Q. Fang, J. Gao, and L. J. Guibas, “Locating and Bypassing Holes in Sensor Networks,” Mob. Netw. Appl., vol. 11, no. 2, pp. 187–200, Apr. 2006, doi: 10.1007/s11036-006-4471-y.
[18] S. Funke and C. Klein, “Hole detection or: ‘how much geometry hides in connectivity?’” in Proceedings of the twenty-second annual symposium on Computational geometry, Sedona Arizona USA: ACM, Jun. 2006, pp. 377–385. doi: 10.1145/1137856.1137911.
[19] X. Feng, X. Zhang, J. Zhang, and A. A. Muhdhar, “A coverage hole detection and repair algorithm in wireless sensor networks,” Clust. Comput., vol. 22, no. S5, pp. 12473–12480, Sep. 2019, doi: 10.1007/s10586-017-1665-y.
[20] M. Rahman, I. Guvenc, D. Ramirez, and C.-W. Wong, “UAV-Assisted Coverage Hole Detection Using Reinforcement Learning in Urban Cellular Networks,” Mar. 09, 2025, arXiv: arXiv:2503.06494. doi: 10.48550/arXiv.2503.06494.
[21] K. Asante, Y. M. Missah, and F. T. M. Asante, “Network-centric optimal hybrid sensing hole recovery and self-healing in IPV6 WSNs,” Int. J. Wirel. Mob. Netw., doi: 10.5121/ijwmn.
[22] R. Chowdhuri and M. K. D. Barma, “Enhancing Network Reliability: Exploring Effective Strategies for Coverage-Hole Analysis and Patching in Wireless Sensor Networks,” Wirel. Pers. Commun., vol. 134, no. 1, pp. 487–517, Jan. 2024, doi: 10.1007/s11277-024-10933-7, doi: 10.1007/s11277-024-10933-7.
[23] G. Simionato and M. G. Cimino, “Swarm intelligence for hole detection and healing in wireless sensor networks,” Comput. Netw., vol. 250, p. 110538, 2024, https://doi.org/10.1016/j.comnet.2024.110538.
[24] B. Khalifa, Z. Al Aghbari, and A. M. Khedr, “A distributed self-healing coverage hole detection and repair scheme for mobile wireless sensor networks,” Sustain. Comput. Inform. Syst., vol. 30, p. 100428, 2021, https://doi.org/10.1016/j.suscom.2020.100428.
[25] S. Wang, H. You, Y. Yue, and L. Cao, “A novel topology optimization of coverage-oriented strategy for wireless sensor networks,” Int. J. Distrib. Sens. Netw., vol. 17, no. 4, p. 155014772199229, Apr. 2021, doi: 10.1177/1550147721992298.
Netw., vol. 26, no. 3, pp. 2223–2239, Apr. 2020, doi: 10.1007/s11276-019-02067-7.
[27] C. Qiu, H. Shen, and K. Chen, “An Energy-Efficient and Distributed Cooperation Mechanism for k-Coverage Hole Detection and Healing in WSNs,” in 2015 IEEE 12th International Conference on Mobile Ad Hoc and Sensor Systems, Dallas, TX, USA: IEEE, Oct. 2015, pp. 73–81. doi: 10.1109/MASS.2015.115.
[28] Y. Zhang, X. Zhang, Z. Wang, and H. Liu, “Virtual edge based coverage hole detection algorithm in wireless sensor networks,” in 2013 IEEE wireless communications and networking conference (WCNC), IEEE, 2013, pp. 1488–1492.
[29] C. Qi, G. Zhang, J. Ren, and D. Li, “Coverage Holes Detecting and Healing in Hybrid Wireless Networks,” in Proceedings of the 10th EAI International Conference on Mobile Multimedia Communications, Chongqing, People’s Republic of China: EAI, 2017. doi: 10.4108/eai.13-7-2017.2271748.
[30] A. Ghosh, “Estimating coverage holes and enhancing coverage in mixed sensor networks,” in 29th annual IEEE international conference on local computer networks, IEEE, 2004, pp. 68–76.
[31] G. Zhang, C. Qi, W. Zhang, J. Ren, and L. Wang, “Estimation and healing of coverage hole in hybrid sensor networks: A simulation approach,” Sustainability, vol. 9, no. 10, p. 1733, 2017, doi:10.3390/su9101733.
[32] G. Dai, H. Lv, L. Chen, B. Zhou, and P. Xu, “A novel coverage holes discovery algorithm based on voronoi diagram in wireless sensor networks,” Int J Hybrid Inf Technol, vol. 9, no. 3, pp. 273–282, 2016, doi:10.14257/ijhit.2016.9.3.25.
[33] T. Gargula, “Circle-circle intersection. A universal method for solving typical surveying problems,” Appl. Geomat., vol. 16, no. 4, pp. 1047–1056, Dec. 2024, doi: 10.1007/s12518-024-00598-z.

Authors

Sharmila Devi received her M. Tech. degree in Computer Science and Engineering from Guru Jambheshwar University of Science & Technology (GJUS&T), Hisar, India. She is pursuing her Ph. D. degree in Department of Computer Science and Engineering, Guru Jambheshwar University of Science & Technology (GJUS&T), Hisar, India. Her areas of research focus on Wireless Sensor Networks.

Dr. Anju Sangwan is currently working as an Assistant professor in the Department of Computer Science and Engineering, Guru Jambheshwar University of Science & Technology (GJUS&T), Hisar, India. She has received the M.Sc (Pure Mathematics) and M. Tech (IT) degrees from the Banasthali University, Rajasthan, India in 2007 and 2009, respectively and Ph. D from Guru Jambheshwar University of Science & Technology, Hisar, India. She has total 21 publications in different journals and international conferences to her credits. Her area of interest is wireless communication and sensor networks.

How to cite this article:

Sharmila Devi, Anju Sangwan, “Coverage Optimization in Wireless Sensor Networks by Detecting and Healing Uncovered Area Through Iterative Sensor Deployment: A Computational Geometry Based Approach”, International Journal of Computer Networks and Applications (IJCNA), 12(4), PP: 475-486, 2025, DOI: 10.22247/ijcna/2025/29.