Wireless Pers Commun (2018) 101:75–86
https://doi.org/10.1007/s11277-018-5668-7

A Greedy Approach for Coverage Hole Detection and Restoration in Wireless Sensor Networks

Manoj Verma[1] · Sanjay Sharma[1]

Published online: 11 May 2018

Abstract Wireless sensor network contains several small sensor nodes that are designed to work autonomously. Coverage preservation is an underlying requirement to efficiently deliver certain services in WSNs. During network operation, some sensor nodes die because of several reasons like energy exhaustion, link failure, node failure etc. We refer it as coverage hole problem of WSNs. In this paper, a new decentralized, node based, localized algorithm called Coverage Hole Detection and Restoration is proposed for detection as well as restoration of coverage holes. Our proposed algorithm is expected to outperform existing algorithms on the parameters of energy and time consumption for convex and non-convex holes.

Keywords Wireless sensor networks · Coverage hole problem · Coverage hole detection · Coverage hole restoration

1 Introduction

Wireless Sensor Networks (WSNs) consists of tiny battery operated sensor nodes that are designed to deliver data for applications like environment monitoring, intrusion detection, surveillance systems etc. [1]. These sensor nodes are deployed randomly or manually in a specific target region to collect, process and send the sensed data to remotely situated base station. Some critical applications like battlefield surveillance systems and intrusion detection need fully covered target region to make it work efficiently. During network functioning, some sensor nodes die due to energy depletion that may result in partial

✉ Manoj Verma
Manojverma162@gmail.com

Sanjay Sharma
drssharma66@gmail.com

[1] Department of Mathematics and Computer Applications, Maulana Azad National Institute of Technology, Bhopal, India

breakage of network coverage. We refer it as coverage hole problem of WSNs [2] [3]. These coverage holes affect the overall performance of WSNs. Therefore, an accurate detection and an efficient restoration technique for these coverage holes is highly required.

Several coverage preserving algorithms have been proposed [2–5] to maximize coverage lifetime of WSNs. Some Coverage aware scheduling algorithms are proposed in [2, 3] to balance energy consumption and intend to prolong coverage lifetime. In [4, 5], some coverage and energy aware clustering algorithms are proposed in which a sensor node with completely overlapped coverage area is selected as cluster head to route the data. Early deaths of these cluster heads do not create any coverage hole in the network as their sensing area is completely covered by their neighbor sensor nodes. However, after going through many transmission rounds some more sensors gets down those results in inescapable coverage holes. Some more coverage preserving algorithms are reported in [6–8] but they lack in detection of coverage holes hence missing restoration techniques. Some coverage hole detection techniques proposed by some researchers are reported in [9–16] but they do not provide any restoration strategy.

In this work, we mainly focus on to devise a new algorithm to detect and restore coverage holes in WSNs. We proposed a solution on the basis of Crucial Intersection Points (CIPs) which is a new concept. Our proposed solution can be divided into two parts: (1) A Crucial Intersection Points (CIPs) based localized algorithm for coverage hole detection and (2) A perpendicular bisection method based location calculation algorithm for coverage hole restoration.

The remaining part of this paper is organized as follows: Sect. 2 reviews the related work on coverage hole description, detection and restoration. Section 3 provides a description of problem formulation and terminologies used in this work. In Sect. 4 we propose our coverage hole detection algorithm. Section 5 describes the proposed coverage hole restoration algorithm. Performance is evaluated in Sect. 4 and finally Sect. 5 concludes the paper.

2 Related Work

Many coverage preserving algorithms have been proposed in [2–8] to maximize the coverage lifetime of WSNs. They are mainly coverage aware clustering and scheduling algorithms. In [2], a dynamic event detection technique called on-demand k-coverage is proposed in which static sensor nodes can adjust their sensing range during transmission time. They analyzed probability of event detection and lifetime of the scheme and conclude that increased network lifetime is achieved. A residual energy based sensor scheduling algorithm is proposed in [3]. Sensors can be put into sleep mode, activated or selected as a CH in this scheme. Main objective of this algorithm is to increase the network lifetime with fully covered target area. Algorithm in [4] is a coverage aware clustering protocol called as CACP. In this algorithm a new coverage aware cost metric is used for cluster head selection and a layered self activation scheme for active nodes selection is proposed. Objective of maximized network lifetime despite death of some sensor nodes is achieved. Another coverage and energy aware algorithm is reported in [5] called as ECDC. In this, they considered residual energy and coverage aware cost metric both for cluster head selection. Other coverage preserving algorithms are reported in [6–8]. All the above algorithms are designed with a common objective of maximized coverage lifetime. But, one common drawback of all these is that they do not provide any strategy to detect coverage hole formation in the network and nothing proposed for its restoration.

Various coverage hole detection algorithms are proposed in [9–16]. Algorithm [9] is a computational geometry based hole detection protocol for self organized WSNs. It can be applied for all kind of irregular polygons. In [10], two types of simplicial complexes are adopted to spot the coverage holes. Limitation of this algorithm is that it can detect only non triangular holes. In [11], a triangular oriented diagram is proposed for detection and calculation of coverage hole size. For healing purpose, they used circumcircle or in circle type to determine target localization for mobile sensors. But this algorithm does not provide full restoration with optimal number of nodes. Similar problem is tackled in [12] by using Delaunay triangulation based method. They used the properties of empty circle for detection of coverage hole. A boundary nodes clustering algorithm is also provided in it to identify different kind of coverage holes. In [13], a homology based approach for coverage hole detection is proposed. The relationship between cech complex and rips complex is analyzed in terms of coverage hole under different ratios between sensing radius and communication radius of a node. Simulation shows that this algorithm cannot localize all kind of coverage holes. In [14], a distributed protocol for coverage hole detection is proposed. They simply calculated the intersection points of nodes and filtered them into covered and non-covered. Then these intersection points are used to detect the coverage holes in network. In [15], an algorithm is presented for the same problem by using simple distance calculation and minimizing the number of transmitted messages. Each node determines whether it is a boundary node or not by verifying its intersection points. Similar problem is tackled in [16], in which a solution based on hop count measures is presented. The most alluring feature of this protocol is that it uses only basic data traffic information to detect coverage holes in the network. But the major shortcoming is that It is not suitable for large scaled randomly deployed wireless sensor networks. All the above discussed techniques contains one common drawback that they do not propose any strategy to overcome this coverage hole problem.

Several other algorithms for coverage hole restoration are also proposed in [17–24]. In [17], they proposed an algorithm for coverage hole detection and patching by using simplicial complexes. They used perpendicular bisection method to patch the coverage holes. It can work even if only partial coordinate information of sensor node is available. But the Number of nodes required for patching are not optimal because of redundant deployment of patching nodes. Similar problem is tackled in [18, 19] using mobile sensor nodes. But these solutions are not suitable for purely static wireless sensor network. In [20], an algorithm based on intersection points between sensors is proposed. They proposed a hole patching strategy based on perpendicular bisector of line between those intersection points. But they ignored the scenario of non-convex holes in the network and the required nodes for restoration are also not optimal. In [21, 22] a rearrangement method based algorithm are proposed but they are not suitable for static wireless sensor networks. In [23], they proposed a solution by increasing sensing range of sensor nodes. They divided whole target region into covered and uncovered cells. A sensor node with higher residual energy has been given priority to recover by increasing its sensing range. But this algorithm will not work where the nodes are deployed at its maximum sensing range. In [24], a tree-based method is proposed to detect the existing coverage holes. After detection of hole they also proposed a method to dissect a large hole into several smaller holes and then determine the optimal patch position for each sub-coverage hole. Drawback of this algorithm is that they Ignored the effects of small coverage holes on the overall performance of a network and also Ignored the scenario of non-convex holes. All these techniques do not present an optimum solution for coverage hole detection as well as restoration.

The proposed algorithm (CHDR) has following additional benefits:

1. A localized coverage hole detection mechanism in which each node tries to detect the coverage hole in network and after that sends the information to base station. Hence, it is scalable and distributable.
2. It can detect and restore both convex and non-convex holes efficiently.

3 Network Model and Terminologies

The Network model and some definitions of related terminologies used in this paper are presented in this section.

3.1 Network Model

The network model we assumed for the proposed algorithm is as follows: A set of sensor nodes deployed randomly over a 2-dimensional target region R and they are fixed after deployment. Every sensor node knows geographic location of itself and its neighbors via GPS or any other location information system. We consider that the sensing range (R_s) and communication rang (R_c) of each node is equal ($R_s = R_c$) and are immutable after deployment. If any area r in target region R is not covered by at least one sensor node then this unattended area will be called as coverage hole. We consider binary communication model, which means sensor is detectable inside the circle of communication range R_c or undetectable when outside. Sensing model is also considered binary, which means the area can be sensed inside the circle of sensing range R_s, or cannot be sensed when outside.

3.2 Terminologies

Following terminologies are used during designing of this algorithm:

Definition 1 (Target region): We defined a target region T_r that is required to be sensed by sensor nodes. If all the points inside T_r are covered by sensor nodes it means the region has no coverage holes. For simplification we used rectangular target region in this paper.

Definition 2 (Neighbor nodes): Nodes that comes under the sensing range of a node S are called as neighbor nodes of S denoted as $N_n(S)$.

$$\{S_i \in N_n(S) | |S_i S| < 2R_s|\}$$

Definition 3 (Intersection points): A point P is called as an intersection point if

(1) It lies on the boundary of two sensor nodes S_i and S_j, i.e. $|PS_i| = |PS_j| = Rs$

OR

(2) It lies on the boundary of a sensor node S_i and target region T_r

(a) *Intersection points between two sensors* (Fig. 1):-

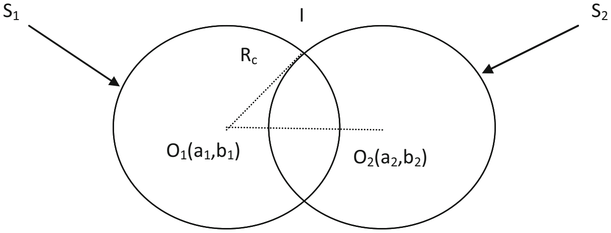

Fig. 1 Intersecting sensor nodes S_1 and S_2

$$\mathbf{O_1I} = \mathbf{R_s}$$

$$\mathbf{O_1O_2} = \sqrt{(a_2 - a_1)^2 + (b_2 - b_1)^2}$$

Intersection points exists only if $O_1O_2 < 2R_s$ and $O_1O_2 \neq 0$

Equation of sensor node S_1 and S_2 will be:

$$(\mathbf{X} - \mathbf{a_1})^2 + (\mathbf{Y} - \mathbf{b_1})^2 = \mathbf{R_s^2}$$

$$\text{and} \quad (\mathbf{X} - \mathbf{a_2})^2 + (\mathbf{Y} - \mathbf{b_2})^2 = \mathbf{R_s^2}$$

Using these equations we will find out the intersection points between two sensors.

(b) *Intersection points between sensor nodes and boundary line of the region:-*

Line's equation using two points $(X_{1,}\ Y_1)$ and (X_2, Y_2) is –

$$\mathbf{Y} = \mathbf{mX} + \mathbf{b}$$

where $\mathbf{m} = \frac{(Y_2 - Y_1)}{(X_2 - X_1)}$ and **b** is the intersection point over Y-axis.

Sensing circle's equation will be –

$$(\mathbf{X} - \mathbf{a})^2 + (\mathbf{Y} - \mathbf{b})^2 = \mathbf{R_s^2}$$

where (a, b) is coordinates of sensor node and R_s is sensing radius.

Using these equations we will find out the intersection points between sensor and target region's boundary line.

Definition 4 (Crucial Intersection Points(CIP)): An intersection point P is said to be Crucial Intersection Point (CIP) if it is not covered by any of the deployed sensor nodes.

Definition 5 (Coverage hole): If an area r of target region R is not covered by at least one sensor node then this unattended area is called as coverage hole. Coverage holes can be classified into two types—close coverage hole and open coverage hole. Close coverage hole is completely surrounded by sensor nodes whereas in case of open coverage hole at least one of its edge is surrounded by boundary line of target region R (Fig. 2).

In the above Fig. 2, it is shown that sensor nodes $\{S_1, S_2, S_3, \ldots\ldots, S_{17}\}$ are randomly deployed over a target region R. Neighbor nodes of S_1 are S_2 and S_5 because |S1, S2| and

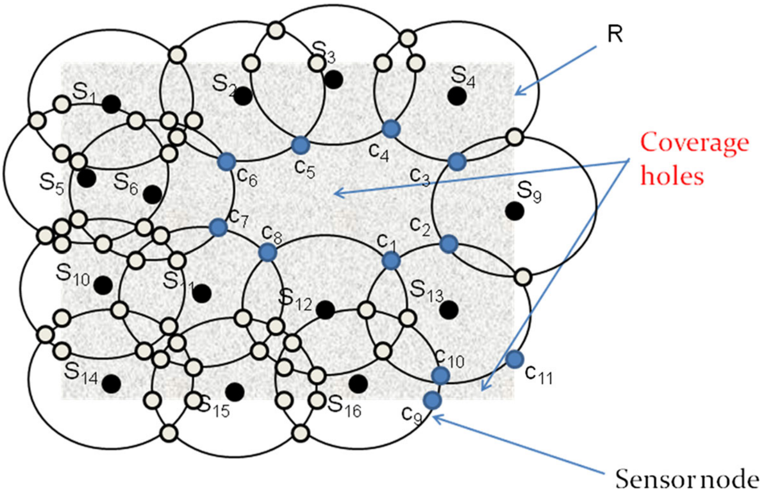

Fig. 2 WSN with coverage holes

|S1, S5| are less than $2R_s$ where R_s is sensing range of a node. Intersection points $\{c_1,c_2,c_3,\ldots\ldots.,c_8\}$ and $\{c_9,c_{10},c_{11}\}$ are the Crucial Intersection Points (CIPs) because they are uncovered by any other node other than their corresponding intersecting sensor nodes. Intersection points $\{c_1,c_2,c_3,\ldots\ldots.,c_8\}$ form a closed coverage hole and $\{c_9,c_{10},c_{11}\}$ form an open coverage hole.

4 Proposed Strategy

In this section, we present our strategy for coverage hole detection and restoration using Crucial Intersection Points (CIPs). Our first strategy is for coverage hole detection using CIPs and after that in second strategy we find out the position for new patching node using perpendicular bisection method of a line.

4.1 Coverage Hole Detection

For detection of coverage holes in network we used crucial intersection points based method to detect the exact region which is not covered by any node other than the intersecting ones. First of all, each sensor node find out the crucial intersection points between neighbor nodes and itself. After that, on the basis of these intersection points it will find out the uncovered regions of the network called coverage holes. Detailed procedure is given below (Fig. 3).

Pseudo code for Coverage hole detection:-

```
1. Initialize Sensor node list (S) to a predefined integer value.
2. Initialize Crucial Intersection Point (CIP) list (C) to NULL.
3. Select a sensor node $S_i$ from Sensor node list (S) and obtain a set of adjacent sensing neighbors $N_n(S_i)$.
4. Create a list of node's sensing neighbors and sort them in clockwise.
5. Create a list of intersection points ($P_i$) with its sensing neighbors and the target region.
6. Check all intersection points ($P_i$) for Crucial Intersection Point and remove the non-crucial intersection points.
7. If Crucial Intersection Point found
       If CIP list is not NULL
           If the starting crucial intersection point($C_i$) is revisited or boundary of the
                   monitoring region is touched.
                   Connect each consecutive Crucial Intersection Point($C_i$) along the
                   boundary line of the nodes having $C_i$ and EXIT.
           Else
                   Add the Crucial Intersection point to CIP list(C) and
                   Select Crucial neighbor sensor and
                   Go to step(2).
           Else
                   Add the Crucial Intersection point to CIP list(C) and
                   Select crucial neighbor sensor and
                   Go to step(2).
8. Continue this procedure until the Sensor node list (S) is NULL.
```

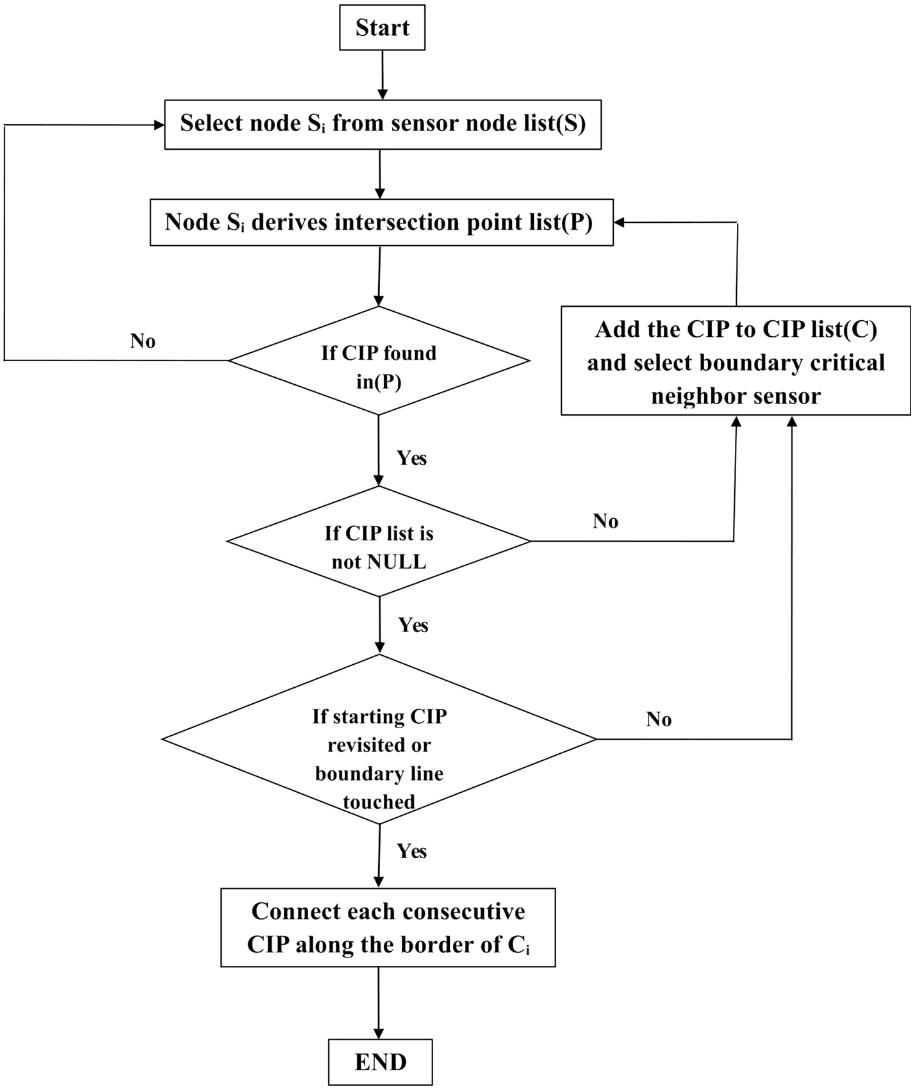
Start
Select node Sᵢ from sensor node list(S)
Node Sᵢ derives intersection point list(P)
If CIP found in(P)
No
Yes
Add the CIP to CIP list(C) and select boundary critical neighbor sensor
If CIP list is not NULL
No
Yes
If starting CIP revisited or boundary line touched
No
Yes
Connect each consecutive CIP along the border of Cᵢ
END

Fig. 3 Flow chart of coverage hole detection algorithm

4.2 Coverage Hole Restoration

After detection of coverage hole in network we have to restore those holes by deploying new nodes over optimum locations. To detect the location of new node we used bisection method for straight line. The line is drawn between two crucial intersection points which are situated at maximum stored distance. The detailed procedure is given below (Fig. 4).

Pseudo code for Coverage hole restoration:-

```
1. Initialize Euclidian distances set($D_{ij}$) to zero.
2. Input a point $C_i$ from CIP list (C).
3. Select the next adjacent point $C_j$ in clockwise.
4. Calculate the Euclidian distance from $C_i$ to $C_j$.
5. If Euclidian distance between $|C_i, C_j|$ is less than or equal to $2R_s$
       Store $|C_i, C_j|$ to set $D_{ij}$
       Add one to j
       Go to step 2
   Else
       Select maximum value from stored distances set($D_{ij}$) and
       find two points $N_1$ & $N_2$ on the both side of the bisector of the corresponding $b_ib_j$ such that $N_1C_i = N_1C_j = R_s$ & $N_2C_i = N_2C_j= R_s$
6. If $N_1$ is already covered by any sensor node
       If $N_2$ is already covered by any sensor node
           Then deploy new node $S_n$ on Midpoint of $b_ib_j$
       Else
           Deploy new node on $S_n$ on $N_2$
   Else
       Deploy new node $S_n$ on $N_1$

7. Select node $S_n$ and run Coverage Hole Detection strategy from step 4–8.
```

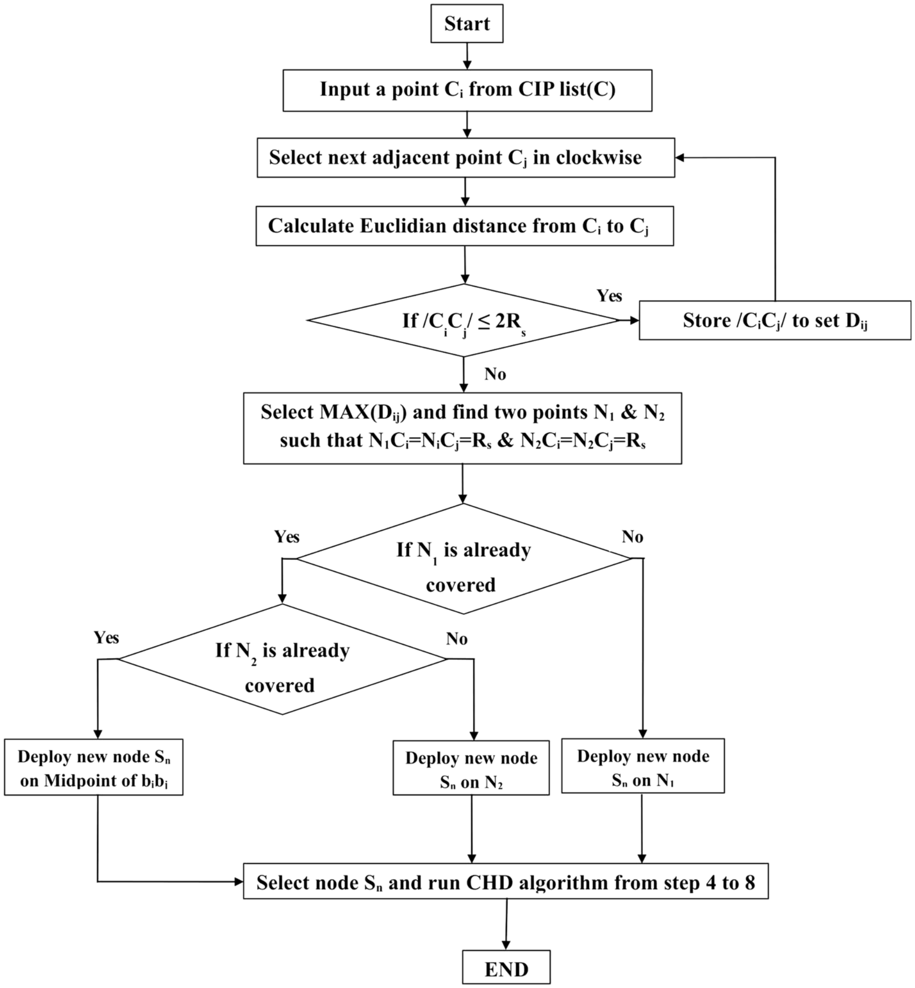
Start
Input a point C_i from CIP list(C)
Select next adjacent point C_j in clockwise
Calculate Euclidian distance from C_i to C_j
If $/C_iC_j/ \leq 2R_s$
Yes
Store $/C_iC_j/$ to set D_{ij}
No
Select MAX(D_{ij}) and find two points N_1 & N_2 such that $N_1C_i=N_iC_j=R_s$ & $N_2C_i=N_2C_j=R_s$
Yes
If N_1 is already covered
No
Yes
If N_2 is already covered
No
Deploy new node S_n on Midpoint of b_ib_i
Deploy new node S_n on N_2
Deploy new node S_n on N_1
Select node S_n and run CHD algorithm from step 4 to 8
END

Fig. 4 Flow chart of coverage hole restoration algorithm

5 Conclusion and Future Work

In this paper, we proposed a solution for detection and restoration of coverage holes in the wireless sensor networks. Our algorithm uses intersection points based approach for detection as well as restoration of coverage holes. It is expected that this algorithm can detect coverage holes accurately. It is also expected that our proposed restoration algorithm will require less number of patching nodes for restoration. This algorithm will outperform over other methods for convex and non convex holes restoration. It can be applied to any kind of polygon shaped area. Further simulation will use our algorithm and compare with the existing available approaches. In future, we will work on to reduce the complexity of this algorithm.

References

1. Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). Wireless sensor networks: A survey. *Computer Networks, 38*(4), 393–422.
2. Alam, K. M., Kamruzzaman, J., Karmakar, G., & Murshed, M. (2014). Dynamic adjustment of sensing range for event coverage in wireless sensor networks. *Journal Network and Computer Applications, 46,* 139–153.
3. Shi, K., Chen, H., & Lin, Y. (2015). Probabilistic coverage based sensor scheduling for target tracking sensor networks. *Information Sciences, 292,* 95–110.
4. Wang, B., Lim, H. B., & Ma, D. (2012). A coverage-aware clustering protocol for wireless sensor networks. *Computer Networks, 56*(5), 1599–1611.
5. Gu, X., Yu, J., Yu, D., Wang, G., & Lv, Y. (2014). ECDC: An energy and coverage-aware distributed clustering protocol for wireless sensor networks. *Computers & Electrical Engineering, 40*(2), 384–398.
6. Torkestani, J. A. (2013). An adaptive energy-efficient area coverage algorithm for wireless sensor networks. *Ad Hoc Networks, 11*(6), 1655–1666.
7. Misra, S., Kumar, M. P., & Obaidat, M. S. (2011). Connectivity preserving localized coverage algorithm for area monitoring using wireless sensor networks. *Computer Communications, 34*(12), 1484–1496.
8. Di, X. (2012). A novel coverage-preserving clustering algorithm for wireless sensor networks. *Physics Procedia, 33,* 1054–1059.
9. Ma, H. C., Sahoo, P. K., & Chen, Y. W. (2011). Computational geometry based distributed coverage hole detection protocol for the wireless sensor networks. *Journal of network and computer applications, 34*(5), 1743–1756.
10. Yan, F., Martins, P., & Decreusefond, L. (2011, December). Connectivity-based distributed coverage hole detection in wireless sensor networks. In *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE* (pp. 1–6). IEEE.
11. Babaie, S., & Pirahesh, S. S. (2012). Hole detection for increasing coverage in wireless sensor network using triangular structure. *arXiv preprint* arXiv:1203.3772.
12. Li, W., & Zhang, W. (2015). Coverage hole and boundary nodes detection in wireless sensor networks. *Journal of network and computer applications, 48,* 35–43.
13. Yan, F., Vergne, A., Martins, P., & Decreusefond, L. (2015). Homology-based distributed coverage hole detection in wireless sensor networks. *IEEE/ACM Transactions on Networking, 23*(6), 1705–1718.
14. Kumar Sahoo, P., Chiang, M. J., & Wu, S. L. (2016). An efficient distributed coverage hole detection protocol for wireless sensor networks. *Sensors, 16*(3), 386.
15. Aliouane, L., & Benchaïba, M. (2016, May). Efficient boundary detection of coverage hole in WSNs. In *Networks, Computers and Communications (ISNCC), 2016 International Symposium on* (pp. 1-6). IEEE.
16. Antil, P., Malik, A., & Kumar, S. (2016). Neighbor Adjacency based Hole Detection Protocol for Wireless Sensor Networks. *Procedia Computer Science, 79,* 866–874.
17. Yao, J., Zhang, G., Kanno, J., & Selmic, R. (2009, May). Decentralized detection and patching of coverage holes in wireless sensor networks. In *Proc. of SPIE* (Vol. 7352, p. 73520 V).
18. Nguyen, D. T., Nguyen, N. P., Thai, M. T., & Helal, A. (2011, July). An optimal algorithm for coverage hole healing in hybrid sensor networks. In *Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International* (pp. 494-499). IEEE.
19. Li, F., Xiong, S., & Wang, L. (2011, December). Recovering coverage holes by using mobile sensors in wireless sensor networks. In *Computational Intelligence and Security (CIS), 2011 Seventh International Conference on* (pp. 746–749). IEEE.
20. Kang, Z., Yu, H., & Xiong, Q. (2013). Detection and recovery of coverage holes in wireless sensor networks. *JNW, 8*(4), 822–828.
21. Abo-Zahhad, M., Ahmed, S. M., Sabor, N., & Sasaki, S. (2015). Rearrangement of mobile wireless sensor nodes for coverage maximization based on immune node deployment algorithm. *Computers & Electrical Engineering, 43,* 76–89.
22. Sahoo, P. K., & Liao, W. C. (2015). HORA: A distributed coverage hole repair algorithm for wireless sensor networks. *IEEE Transactions on Mobile Computing, 14*(7), 1397–1410.
23. Amgoth, T., & Jana, P. K. (2017). Coverage hole detection and restoration algorithm for wireless sensor networks. *Peer-to-Peer Networking and Applications, 10*(1), 66–78.
24. Li, W., & Wu, Y. (2016). Tree-based coverage hole detection and healing method in wireless sensor networks. *Computer Networks, 103,* 33–43.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Manoj Verma received Master of Computer Applications degree from Maulana Azad National Institute of Technology, Bhopal, India in 2013. He is pursuing Ph.D. in Next Generation Networks from the same institute. His research interests include Next generation networks, Wireless Sensor Networks and Routing Protocols.

Dr. Sanjay Sharma is working as a professor in Department of Mathematics and Computer Applications at Maulana Azad National Institute of Technology, Bhopal, India. He is having a teaching experience of 24 years with many reputed international publications. He is also an IPv6 Certified Gold and Silver Network Engineer from IPv6 forum, University Sains Malaysia. He also served as Director and Deputy Director from 2005 to 2009 in All India Council for Technical Education (AICTE), New Delhi, under Ministry of Human Resource Development, Government of India. His research interests include Next generation networks and ipv6, Wireless Sensor Networks and routing protocols, Big Data Analytics and Cloud Computing (SAAS Platform).