

***k*-coverage Problems and Solutions in Wireless Sensor Networks: A Survey**

Nilufa Yeasmin
Stamford University Bangladesh

ABSTRACT

Wireless Sensor Networks (WSNs) consist of sensor nodes those are capable of sensing environmental data such as temperature, humidity, pressure, sound etc. and transmit those data to a basestation through the network. Achieving sufficient coverage in the sensing area is a challenging feature in WSNs. Coverage indicates that how well an area is monitored i.e., whether each location of an area is within the sensing range of sensors. Some WSNs applications require multilevel coverage where each location in an area is covered by more than one sensor node. Moreover, sensor nodes must have to maintain a communication range among themselves. In this paper, a survey has been presented on k -coverage problems in WSNs where each location of an area should be covered by at least k sensors. Here, two categories of k -coverage problems have been identified and their different solution approaches have been discussed.

General Terms:

Wireless Sensor Network Coverage, Geometric Coverage Solution

Keywords:

Sensor nodes, coverage, k -coverage, verification,

1. INTRODUCTION

In recent years, Wireless Sensor Networks (WSNs) have been widely applied in different real-time applications such as e-health monitoring, home automation, forest fire detection, object tracking etc. The applications use sensor nodes to be deployed in different indoor and outdoor locations. The deployed nodes cover a region individually or collectively to provide network services or sensing capabilities. Coverage is one of the most important performance parameter for WSNs which reflect how well a target area is monitored or covered by the deployed sensors. A number of researches have been performed to solve the coverage-related problems. Coverage problems in WSNs can broadly be classified into three major categories: area coverage, point coverage and barrier coverage [10]. In the area coverage problem, the main objective is to monitor or cover all locations of a given area, sometimes referred to as region. However, point coverage problem focuses on covering a set of points (i.e., targets) from all locations of a given area. On the other hand, barrier coverage problem deals with constructing a barrier for intrusion detection in the deployed WSNs region. The goal is to minimize the probability of undetected penetration through

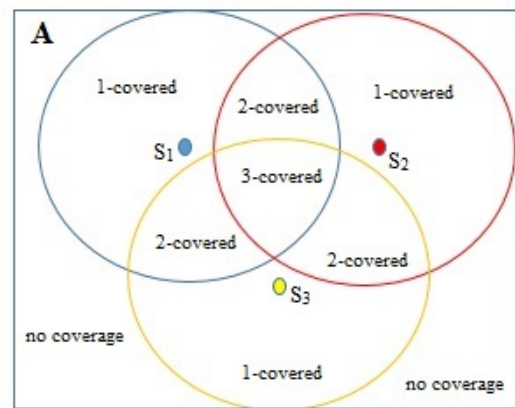


Fig. 1. Coverage Level

the barrier of the WSNs region. In this survey, the focus is on the k -coverage problems in WSNs those falls in the category of area coverage problem.

k -coverage problem concerns with covering a region by k sensors where $k \geq 1$. For example, if sensing range of 2 (two) sensors overlap, i.e., a region is within the sensing range of 2 sensors, the overlapped region is called 2-covered. Fig. 1 shows no-coverage regions, 1-covered, 2-covered and 3-covered regions by 3 (three) disc sensing range sensors, S_1 , S_2 and S_3 for a given rectangular area A . Similarly, a region is called k -covered if it is within the sensing range of at least k different sensors where $k \geq 1$. Different applications of WSNs like fire detection, radiation detection, and intrusion detection may require k -coverage in real-time. It helps to design a fault-tolerance robust network. Moreover, redundant k sensors can ensure more reliable sensor data received from the environment. Some WSNs applications used for location determination also need k -coverage for improved accuracy.

Coverage problems have been addressed in different research fields. The Art Gallery problem [11] and the Disc Covering problem [16] of computational geometry are related to the coverage problem. The Art Gallery Problem refers to the minimum number of observers required to monitor a polygon area and it assumes that an observer can watch all the points within its line of sight. The Disc Covering problem asks for the smallest radius of n identical discs which can be arranged to cover a unit disc. However, solutions

of these problems are not directly applicable to WSNs applications due to the nature of sensor nodes. For example, sensor nodes have limited sensing range and battery life whereas the observers of the Art Gallery problem have infinite visibility unless any obstacle appears. Unlike the observers, all the sensor nodes need to communicate with each other within its limited communication range in WSNs. Sometime, the distributed nature of WSNs also restricts to adapt all solutions those are solved in centralized manner.

This survey focuses on two categories of k -coverage problems and their solutions for WSNs. There are some surveys [10][14][7] based on the different coverage problems in WSNs. However, in this article, specifically the k -coverage problems and solutions for WSNs have been presented, which is first to the best of my knowledge. Two following categories of k -coverage problems have been identified:

- (1) **k -coverage Verification:** In this category, k -coverage problem is formulated as a decision problem where a given area needs to be verified whether it is k -covered or not.
- (2) **Selecting Subset from Deployed Nodes for k -coverage:** The problem selects minimum subset from already deployed sensor nodes to activate from sleeping mode in order to maintain k -coverage in the given area.

Different solution approaches for each category have been discussed in the rest of the article.

2. K-COVERAGE VERIFICATION

Problem Definition: Given an area deployed with n sensors, determine whether the area is k -covered, i.e., every location in the area is within the sensing range of at least k sensors where $k \geq 1$. This is a decision problem [6].

Wang *et al.* [15] have proposed Coverage Configuration Protocol (CCP) to provide different degrees of coverage in the area required by applications. It has been proved that a set of sensors covering a convex region are connected as long as their communication range is no less than twice of the sensing range. This property is used to derive the relationship between the coverage degree and connectivity among sensor nodes. Wang *et al.* have claimed that k -coverage could be ensured for a convex region A , if 1) there are intersection points between sensors or between sensors and A 's boundary; 2) all the intersection points between sensors are at least k -covered; and 3) all intersection points between any sensor and A 's boundary are at least k -covered. The CCP allows to verify the k -coverage by only checking the intersection points. The protocol further extends the verification problem to activate a sensor from sleeping mode based on k -coverage eligibility criteria. The criteria states that a sensor is eligible to be active if all the intersection points inside its sensing circle are at least k -covered. The complexity of the eligibility algorithm is $O(n^3)$ where n is the number of neighboring nodes within a sensor's sensing radius.

In [6], Huang and Tseng have provided a polynomial time algorithm in terms of the number of sensors to solve the k -coverage verification problem. Instead of checking every location in the area, the authors have proposed to check only the perimeter of each sensor to determine its coverage. Each sensor first determines its segments in the perimeter that are covered by neighboring node. This is done by calculating the angle, $\alpha = \arccos(\frac{d(s_i, s_j)}{2r})$, where s_j is the neighboring sensor of s_i and r is the sensing range of s_i . The arch of s_i falling in the angle $[\pi - \alpha, \pi + \alpha]$ is represented as segments perimeter-covered by s_j . The start (left) and end (right)

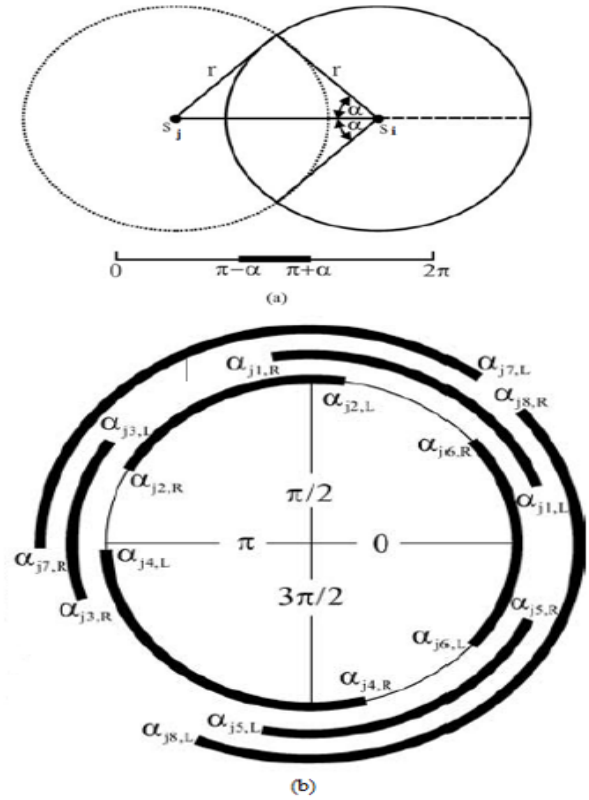


Fig. 2. Determining: (a) the segment of s_i 's perimeter covered by s_j , and (b) the perimeter-coverage of s_i 's perimeter. [6]

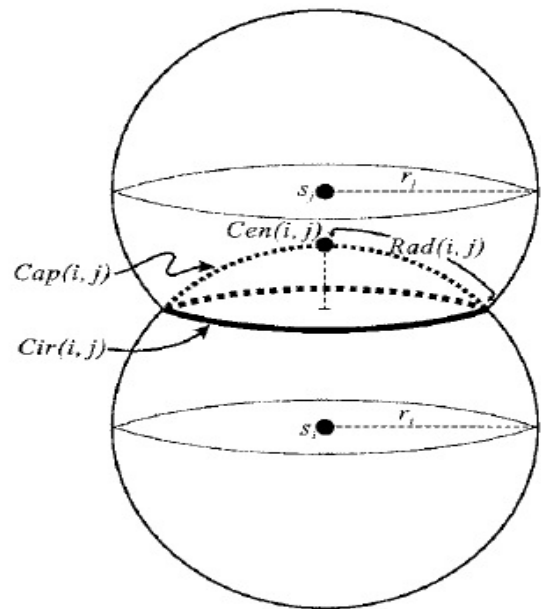


Fig. 3. Illustration of terminology [8]

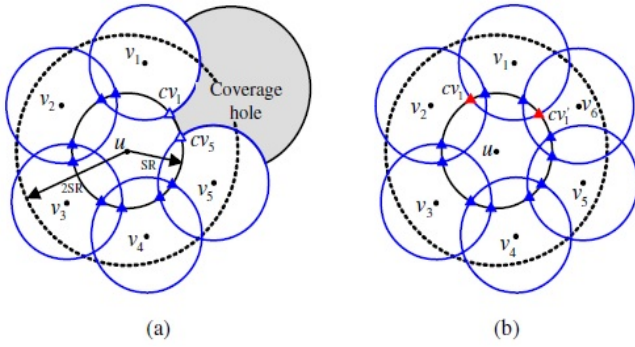


Fig. 4. Intersection Coverage Checking [19]

points of each segment, denoted by $[\alpha_{j,L}, \alpha_{j,R}]$, are placed on a line segment $[0, 2\pi]$. Then all the points are sorted in an ascending order into a list L . Finally, the perimeter coverage of s_i is determined by traversing the line segment $[0, 2\pi]$ while visiting each element in the sorted list L from left to right. The perimeter coverage scenario is illustrated in Fig. 2. It has been proved that as long as the perimeters of sensors are k -covered, the whole region is also k -covered. The authors have considered both the Unit-disc Coverage (k -UC) model and Non-unit-disc Coverage (k -NC) model. The cost of the algorithm for k -UC problem is $O(nd \log d)$ where d is the maximum number of neighbor sensors of a given sensor node. The k -NC problem can also be solved with same complexity as k -UC problem except that the neighbors of a sensor need to be re-defined. This complexity is lower than the approach presented in [15] where the coverage level has been determined by checking the coverage of every intersection points in the sensor's sensing range that results in $O(n^3)$ complexity [6].

The solution presented for k -coverage verification in [6] is applicable for 2-Dimensional area. However, based on the similar concept of perimeter coverage as [6], Huang *et al.* have also proposed a polynomial-time algorithm for solving the k -coverage decision problem in 3-Dimensional area [8]. The authors have formulated the problem as α -Ball-Coverage, where α is any natural number and sensor's sensing region is modeled as ball. Initially, each sensor S_i calculates its spherical cap, $Cap(i, j)$ and circle, $Cir(i, j)$ for each neighboring sensor S_j . $Cap(i, j)$ represents the intersection of sphere S_i and ball B_j whereas $Cir(i, j)$ denotes the intersection of spheres S_i and S_j . These terminologies have been illustrated in Fig. 3. Then, for each neighbor $s_k \neq s_j$, the angle of $Cir(i, j)$ that is circle-covered by $Cap(i, k)$ is calculated which is denoted by $[\theta_{k,L}^j, \theta_{k,R}^j]$. For all the angles, $\theta_{k,L}^j$ and $\theta_{k,R}^j$ points are placed on a line segment $[0, 2\pi]$ which is then sorted in ascending order into a list L_j . Then, the line segment $[0, 2\pi]$ is traversed by sequentially visiting each point in the list L_j in order to determine the circle coverage of $Cir(i, j)$, denoted by cc_j . Finally, the minimum circle coverage for all neighboring circles on S_i is returned as the sphere coverage of s_i . The complexity of the algorithm is $O(d^2 \log d)$ where d is the maximum neighboring sensor of a sensor. Similar to [6], it has been proved that as long as the spheres of sensors are k -covered, the whole region is also k -covered [8]. Both the approaches presented in [6] and [8] consider that all the sensors have same sensing radius and know their locations along with their neighbor sensor's location

The approach presented in [15], [6], and [8] requires that exact locations of all deployed sensors are known. In contrast, Zhang

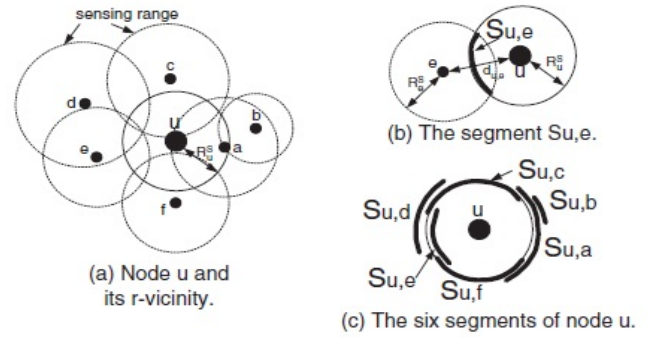


Fig. 5. Sensing Border Concept [3]

and Wang have proposed a distributed algorithm where each sensor node determines whether they are on the coverage boundary by using localized distance [19]. For k -coverage verification of an area, at first, it is determined whether the intersection points in the sensing border of each node is k -covered. Based on intersection coverage, the algorithm detects boundary node of a coverage hole for the verification of 1-coverage. Boundary node is the node for which there exists at least one intersection point on its sensing border which is not covered by any other node's sensing range. For example, in Fig. 4(a), node u is a boundary node of 1-coverage-hole whereas u is not a boundary node of 1-coverage-hole in Fig. 4(b). Finally, the 1-coverage verification is extended to k -coverage verification that detects the boundary node of a k -coverage hole. The total running time of the algorithm is $O(n^3)$.

Bejerano has proposed a k -coverage scheme that also uses the distance of a node from its neighbor instead of using exact location [3] [4]. However, the assumption is similar to [15], [6] and [8], that is the k -coverage of every internal node can be verified by checking its sensing range border. Moreover, the Transmission Radius (TR) should be twice greater than the Sensing Radius (SR) to ensure connectivity. At first, a node, u , runs a segment sequence algorithm to detect 1-coverage-hole as in [19]. A cyclic segment sequence of node u , denoted as Q_u , is a sequence of segments with the property that every first and last segments overlap with exactly two other segments in Q_u . For example, in Fig. 5, the cyclic segment sequence of node u is $Q_u = \{S_{u,a}, S_{u,c}, S_{u,e}, S_{u,f}\}$. After the calculation of Q_u , the scheme computes r -map coordinates that represents the location of neighbors of node u located in its r -vicinity ($N_u(r)$). r -map coordinate of node v is represented as $\langle d_{u,v}, \theta_{u,v} \rangle$ where $d_{u,v}$ is the Euclidean distance of node v from originating node u and $\theta_{u,v}$ is the angle of node v relative to an arbitrary polar-axis. Then, from the r -map coordinate, the scheme calculates the angle between the start and end point of each segment in the cyclic segment relative to node u and sort them in increasing order of angle. Finally, the algorithm traverse the sensing border of node u and verifies whether every point is at least k -covered. The overall running time of the k -coverage verification scheme is $O(n^3)$, where n is the number of nodes in $N_u(r)$ [3].

So and Ye have designed a framework for the k -coverage decision problem by using k -order Voronoi diagram [13]. The approach presented in [6] and [8] have worst-case running time $O(n^2 \log n)$ and $O(n^3 \log n)$ respectively. Moreover, these complexities are independent of k [13]. On the other hand, the approach used in [13] provides a $O(n \log n + nk^2)$ solution that only depends on the number of sensors and k parameter. The intention of this approach is to

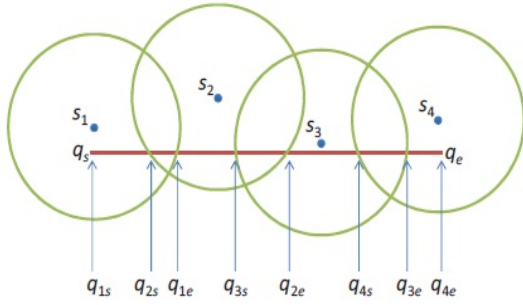


Fig. 6. An illustration of 2-coverage of a line segment $l = q_s q_e$ [9]

find a small set of points in the area S , so that the k -coverage decision problem could be answered by only examining those points efficiently. The sensing range is considered as ball $B(a, r)$ where a is the center and r is radius. At first, k -th order Voronoi diagram, $V_k(l)$ is drawn based on the given center of the balls. Then, only three set of points are checked to determine whether they are k -covered. The point sets are 1) the vertices of $V_k(l)$ for given S , 2) the intersections between the sides of S and $V_k(l)$, and 3) the corners of S . Moreover, the framework can also determine the 1-coverage in R^2 and R^3 respectively in $O(n \log n)$ and $O(n^3)$ time for different sensing range. In addition to the decision problem of k -coverage, the proposed framework can also determine the maximum value of k such that every point in S is k -covered. The complexity of determining k is $O(n^3)$ and $O(n^4)$ respectively for R^2 and R^3 .

Juang *et al.* have addressed the k -coverage verification problem in different way [9]. Here, instead of verifying the total given area, a query line segment is checked to determine whether all sub-segments of the line segment are k -covered i.e., each point locations in the sub-segment is within the sensing range of k sensors. According to Fig. 6, the sub-segments $q_{2s}q_{1e}$, $q_{3s}q_{2e}$, $q_{4s}q_{3e}$ are 2-covered whereas the sub-segments $q_{1s}q_{2s}$, $q_{1e}q_{3s}$, $q_{2e}q_{4s}$, $q_{3e}q_{4e}$ are 1-covered. The intersecting points are called CE-points. For all segments s , q_s and q_e are denoted as in-CE points and out-CE points respectively. Three methods based on R-tree index [5] have been proposed as the solutions where R-tree index of the area S is constructed in advance. The first method called the Basic method, initially stores all the CE points sequentially by traversing meta-R-tree and then the coverage number of the first sub-segment is computed. While traversing the CE points sequentially, if an in-CE-point is encountered, the coverage number is increased by 1; otherwise, the coverage number is decreased by 1 if an out-CE-point of a sensor is encountered. To improve the calculation of finding CE points in Basic method, the second approach, the Projection method uses vertical inscribed squares (VI-square) kept in the leaf nodes of the R-tree. For a sensor, $s(x, y)$ with sensing radius r , the VI-square is an internal square of $r(s)$, where the coordinates of its lower-left and the upper-right points are $(x - \frac{r}{\sqrt{2}}, y - \frac{r}{\sqrt{2}})$ and $(x + \frac{r}{\sqrt{2}}, y + \frac{r}{\sqrt{2}})$ respectively [9]. As a sub-segment of a line segment is covered by a VI-square, it can be used to provide the lower bound of the coverage numbers of sub-segments. Instead of finding CE points, this method computes square covering end points (SCE-points) which is more lightweight. The remaining parts of finding coverage number is same as the Basic method. Though the Projection method provides a lower bound, it can not ensure the actual coverage of some sub-segments [9]. Computation for both the Basic and Projection method depends on the number of

sensors around the given line segment. However, the third method called the tessellation method partitions the given line segment into same length equal to $\frac{r}{2}$ instead of generating the intersection points. Then, for each sub-segment, its corresponding Regular Pruning Window (RPW) [9] is generated and the meta-R-tree is traced to find the number of sensors. If the number of sensors in RPW are larger than or equal to k , the sub-segment is verified as k -covered; otherwise the basic method is applied to the sub-segment. From the experiment, it has been found that the tessellation method provides better performance in terms of running time, compared to the other two methods [9].

3. SELECTING SUBSET FROM DEPLOYED NODES FOR K -COVERAGE

Problem Description: For a desired coverage degree, $k \geq 1$, select a minimal subset of sensors from already deployed n sensors so that every location of the area is within the sensing range of at least k different sensors. This k -coverage problem has been proved to be NP-hard by reduction to the minimum dominating set problem [17].

In [15], the proposed CCP protocol allows each sensor node to run the K_s -eligibility algorithm in order to determine whether the node should be active from sleep mode to maintain k -coverage in the area. However, the algorithm does not return a total size of the subset for activated nodes. Zhou *et al.* [20] have proposed a centralized and a decentralized greedy algorithm to find the minimum size subset of k -covered sensors which delivers a solution of near-optimal factor of $O(\lg n)$. This algorithm takes a set of sensors and a query region as its input and provides the required subset as an output to maintain k -coverage.

Yang *et al.* [18] have represented the sensor subset selection problem as k -(Connected) Coverage Set (k -CCS/ k -CS) problem. The problem has been defined in terms of linear programming (LP) and an approximation algorithm has been proposed as a solution. Furthermore, two more non-global solutions for (k -CCS/ k -CS) problem have also been designed. The first solution uses a regular clustering approach to select backbone nodes to form the subsets. The second solution uses a pruning algorithm based on 2-hop neighborhood information. In this algorithm, a node u is k -covered by a subset C of its neighbors if and only if three conditions are satisfied: 1) The subset C is connected by nodes with higher priorities than u , 2) Any neighbor of u is a neighbor of at least k nodes from C , 3) Each node in C has higher priority than u .

In addition to select a subset for k -coverage, Sheu *et al.* [12] have proposed a k -coverage preserving protocol to achieve energy efficiency while maintaining the required area is k -covered. The authors have modeled this problem as a minimum set k -coverage problem and solved it by using a heuristic greedy algorithm. The k -coverage preserving protocol deals with the probabilistic k -coverage requirement model. In the probabilistic k -coverage protocol, any point within a region can be detected by more than k sensors with no lower than a confidence probability. Given a confidence probability P_c , the purpose is to find the lower bound of the cumulative detection probability at any point in R , and it should not be lower than the threshold P_c . Thus the lower-bound of detection probability $P_d(s_i)$ for probabilistic coverage degree k' is given by:

$$P_c \leq 1 - (1 - P_d(s_i))^{k'}$$

The simulation shows that the k -coverage preserving protocol is close to the performance of lower bound [12].

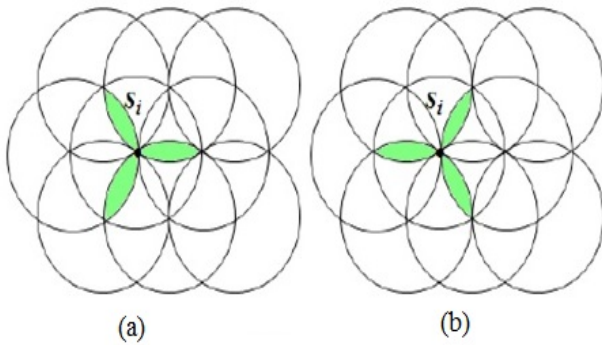


Fig. 7. (a) First and (b) Second three-lens flowers of s_i [2]

While previous researchers studied the problems in homogeneous sensor networks, Ammari et al.[2] have focused on the problem of connected k -coverage in heterogeneous sensor networks. Specifically, random sensor set deployment and pseudo-randomly distributed sensor set deployment has been used for the approach. The authors have shown that it is possible to design distributed protocols for heterogeneous sensor networks. In the protocol, a sensor, s_i randomly picks one of the three-lens flowers as shown in Fig. 7 and checks whether its sensing range is k -covered. However, it has been proved impossible to design efficient centralized protocols due to random deployment and sensor heterogeneity. Thus, a pseudo-random deployment method have been proposed. In this method, sensors are deployed in different layers in a circular field with respect to the sink according to their strengths.

Ammari *et al.* have further studied the connected k -coverage protocols and have proposed four configuration protocols in [1]. The first protocol, called the centralized randomized connected k -coverage, selects a minimum sensor set from a field through the sink node to achieve k -coverage and connectivity. Initially, sensors located in the three lenses of a given slice are selected as the previous work in [2]. At each selection, it is checked whether k sensors have been selected to k -cover the slice, then the degree of coverage for the other adjacent slice is updated. The second and third protocols are called T-clustered randomized connected k -coverage (T-CRACCK) and D-clustered randomized connected k -coverage (D-CRACCK). For the third approach, in each round, the sink node selects a subset of sensors, called cluster-heads, which are responsible for selecting a subset of neighboring sensors to k -cover its underlying cluster and maintain connection to each other. Precisely, each cluster-head exploits the overlap between the slices of its cluster as well as the overlap between its slices and the adjacent cluster-heads to select a minimum number of its sensing neighbors. Both T-CRACCK and D-CRACCK consider different degree of network clustering. In the fourth protocol, called distributed randomized connected k -coverage (DIRACCK), all sensors are required to coordinate among themselves to k -cover a field in each round.

4. CONCLUSION

In this survey, different solutions and their assumptions for two types of k -coverage problem in WSNs have been presented. There exist both centralized and decentralized coverage solutions for the problems. From the survey, it has been found that researches have focused more on solving the k -coverage problem in 2D WSNs than in 3D WSNs. Moreover, homogeneous sensor nodes and disk sensing model have mostly been used in different solutions. A few solu-

tion approaches consider the probabilistic sensing model of WSNs. However, no approaches have been found which consider different obstacles and interference in the given area while maintaining k -coverage. This is a short survey that provides the broad picture of different solution approaches for k -coverage problems. In future, this survey could be more detailed and extended by involving different sensor scheduling scheme and routing protocols those ensure k -coverage in WSNs.

5. REFERENCES

- [1] H.M. Ammari and S.K. Das. Centralized and clustered k -coverage protocols for wireless sensor networks. *Computers, IEEE Transactions on*, 61(1):118–133, Jan 2012.
- [2] H.M. Ammari and J. Giudici. On the connected k -coverage problem in heterogeneous sensor nets: The curse of randomness and heterogeneity. In *Distributed Computing Systems, 2009. ICDCS '09. 29th IEEE International Conference on*, pages 265–272, June 2009.
- [3] Yigal Bejerano. Simple and efficient k -coverage verification without location information. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE. IEEE*, 2008.
- [4] Yigal Bejerano. Coverage verification without location information. *Mobile Computing, IEEE Transactions on*, 11(4):631–643, 2012.
- [5] Antonin Guttman. *R-trees: A dynamic index structure for spatial searching*, volume 14. ACM, 1984.
- [6] Chi-Fu Huang and Yu-Chee Tseng. The coverage problem in a wireless sensor network. *Mob. Netw. Appl.*, 10(4):519–528, August 2005.
- [7] Chi-Fu Huang and Yu-Chee Tseng. A survey of solutions to the coverage problems in wireless sensor networks. *Journal of Internet Technology*, 6(1):1–8, 2005.
- [8] Chi-Fu Huang, Yu-Chee Tseng, and Li-Chu Lo. The coverage problem in three-dimensional wireless sensor networks. *Journal of Interconnection Networks*, 8(03):209–227, 2007.
- [9] Kun-Han Juang, En Tzu Wang, Chieh-Feng Chiang, and Arbee LP Chen. Verification of k -coverage on query line segments. In *Proceedings of the 17th International Database Engineering & Applications Symposium*, pages 114–121. ACM, 2013.
- [10] Junbin LIANG, LIU Ming, and KUI Xiaoyan. A survey of coverage problems in wireless sensor networks. *Sensors & Transducers (1726-5479)*, 163(1), 2014.
- [11] Joseph O'Rourke. *Art gallery theorems and algorithms*, volume 1092. Oxford University Press Oxford, 1987.
- [12] Jang-Ping Sheu and Huang-Fu Lin. Probabilistic coverage preserving protocol with energy efficiency in wireless sensor networks. In *Wireless Communications and Networking Conference, 2007.WCNC 2007. IEEE*, pages 2631–2636, March 2007.
- [13] Anthony Man-Cho So and Yinyu Ye. On solving coverage problems in a wireless sensor network using voronoi diagrams. In *Internet and Network Economics*, pages 584–593. Springer, 2005.
- [14] Bang Wang. Coverage problems in sensor networks: A survey. *ACM Computing Surveys (CSUR)*, 43(4):32, 2011.
- [15] Xiaorui Wang, Guoliang Xing, Yuanfang Zhang, Chenyang Lu, Robert Pless, and Christopher Gill. Integrated coverage

- and connectivity configuration in wireless sensor networks. In *Proceedings of the 1st international conference on Embedded networked sensor systems*, pages 28–39. ACM, 2003.
- [16] Robert Williams. *The geometrical foundation of natural structure*. Dover Publications, 1979.
- [17] Shuhui Yang, Fei Dai, Mihaela Cardei, Jie Wu, and Floyd Patterson. On connected multiple point coverage in wireless sensor networks. *International Journal of Wireless Information Networks*, 13(4):289–301, 2006.
- [18] Shuhui Yang, Fei Dai, Mihaela Cardei, Jie Wu, and Floyd Patterson. On connected multiple point coverage in wireless sensor networks. *Journal of Wireless Information Networks*, 2006, 2006.
- [19] Xia Zhang and Cheng Wang. Distributed k-coverage verification algorithm based on localized distance information in wsns. In *Networking, Architecture, and Storage, 2009. NAS 2009. IEEE International Conference on*, pages 196–199. IEEE, 2009.
- [20] Zongheng Zhou, Samir Das, and Himanshu Gupta. Connected k-coverage problem in sensor networks. In *Computer Communications and Networks, 2004. ICCCN 2004. Proceedings. 13th International Conference on*, pages 373–378. IEEE, 2004.