International Journal of Engineering Research-Online
A Peer Reviewed International Journal
Articles available online http://www.ijoer.in

Vol.3., Issue.2, 2015

RESEARCH ARTICLE

ISSN: 2321-7758

AN OPTIMAL ALGORITHM FOR FULL COVERAGE IN WIRELESS SENSOR NETWORKS

K.KAVITHA[1], T.THAMARAI MANALAN[2]

[1]M.E Applied electronics, Sri Eshwar college of engineering, Coimbatore

[2]Assistant Professor, Sri Eshwar college of engineering, Coimbatore

Article Received: 31/03/2015 | Article Revised on:08/04/2015 | Article Accepted on:11/04/2015

ABSTRACT

Network coverage is vital for wireless sensor networks (WSN) to detect a specified area and to offer a good quality of service. Network coverage means every point surrounded by the area must be enclosed by at least one sensor node. The occurrence of hole is inevitable in the specified area due to the inner nature of WSN, random deployment, environmental factors, and external attacks. For maintaining the coverage quality of the given WSN, we suggest a low complexity distributed and localized algorithm (HEAL) to locate and heal the holes. This algorithm allows a local healing where only the nodes located at a right distance from the hole will be concerned in the healing process. Performance results through ns 2 simulator illustrate that HEAL can deal holes of various forms and sizes and provides a cost-effective and an exact solution for hole discovery and healing. In this paper, we would also like to make sure coverage and network connectivity in a specified area consisting energy hole. We advance the energy hole detection scheme to handle the network with energy hole.

Keywords-Hole detection, hole healing, Distributed and Localized algorithms, energy hole

1. INTRODUCTION

Wireless Sensor Network consists of a large number of sensor nodes with sensing, processing and transmission capabilities. These most recent decades, wireless sensor technology leads several sectors such as environmental monitoring, military surveillance, medical diagnosis, building automation, industrial automation tasks, etc. The sensing range of each node is used to compute parameters of its environment and a communication range to sustain connectivity with other nodes. Many analyses in WSNs are attracted within the deployment of sensors that make sure both network coverage and connectivity. Random deployment is easy, however it causes many issues. Definitely, we can realize several disconnected areas and few regions will be closely covered where as others are poorly covered.

The effectiveness of a sensor network depends on the coverage of the region of interest. Designing sensor networks face many challenges in measurability, self-organization and energy efficiency. A typically used statement in learning sensor networks is that sensors are uniformly closely spread in the area. Though, during a reality, this assumption does not generally hold. Although nodes are distributed uniformly at random, still there are areas with node density a lot of lesser than others. Additional factors such as environment variation and sensor energy consumption can also make non-uniform sensor distributions. Almost, sensor networks in general have coverage holes, i.e. regions without sufficient effective sensors as shown in Fig. 1.

To defeat this difficulty we recommend a method, called hole detection and healing (HEAL). HEAL is a low complexity, distributed and localized algorithm. It can deal with holes of various forms and sizes without considering sensors density and distribution.

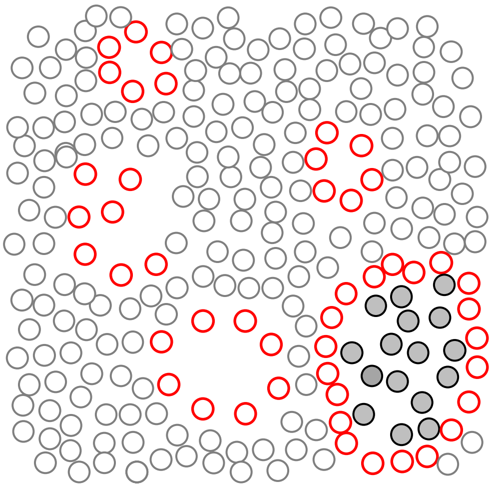

Fig.1 Empty circles correspond to sensors. Large hole is formed with failed black nodes in a network with sensors randomly placed. Small holes are formed with low sensor deployment

2. Related work

2.1. Locating Holes and Healing

Once the sensor network is deployed, the coverage holes may emerge due to low sensor deployment or dead sensor nodes. The failed sensors in the region may screen the connectivity and the coverage of the network reducing the dependability of the network. Thus the network wants to be maintained from time to time. Many research based on hole location and healing has been made. Few of them are as below: Fang et al. [1] found the BoundHole algorithm with the right-hand rule to discover sensors on the border of the holes. This BoundHole algorithm is easy and local. Every sensor stores data regarding its 1-hop neighbors. The processing at every node depends on the 1-hop neighbor and data passed with the packet that traverses the hole border in the BoundHole method. Thus the algorithm is distributed and scales well to huge networks.

Yong et al. [4] found DSSA, the centralized movement-assisted virtual forces-based algorithm to heal the hole. Yangly et al. [2] found grid-quorum based movement, in which the network is partitioned into many tiny grid cells, and the number of sensors in each cell is considered as the load of the cell. In [5], authors handle the point coverage problem with novel evaluation metric. Table. 1 summarizes the main disadvantages of above mentioned proposed algorithms.

TABLE I : Comparison of existing system to locate hole and healing

Existing system	Main Drawbacks
[1]	- High message complexity
[4]	- Centralized approach
[2]	- It may generate huge message overhead when the network is very dense due to the increased number of rounds of scans - At the end of the clustering process, if two adjacent clusters are empty the scan process will not be correct
[5]	- Considers only the point coverage problem

3. Proposed system

In our HEAL algorithm we introduce joint system to locate and repair the holes. Our hole location method deal with holes of different forms and sizes. We crack to find a limited number of sensors neighboring the hole, only those sensors are included in the process of moving and healing the hole.

Before entering into HEAL algorithm, we create the subsequent assumptions:

1. A high density mobile WSN is deployed in an obstacle free region.
2. Node deployment can be deterministic or random.

3. All the deployed nodes are homogeneous (processing power, communication & energy consumption are same).
4. Location information of the every sensor is known by means of some localization method with respect to specified region.
5. Every sensor is identified the boundary range information of the specified region.
6. $R_C \geq 2R_S$ is assumed to make sure a common standard.

3.1. Locating Hole

Fang et al. [1] defined the stuck nodes where data packets can possibly get stuck in greedy multi-hop forwarding. The existence of stuck nodes indicates the existence of holes. Here the source node broadcasts a data packet to its 1-hop neighbors in the path towards the sink node. This method is continued until the packet delivers to the sink or it is stuck at a sensor whose all 1-hop neighbors are isolated from the pathway of sink. A node where a packet may get stuck is called stuck node [3]. But in contrast, if there is no sink node existing for the deployed nodes then the chance of finding a stuck node becomes not practical.

We suggest an algorithm QUADRANT rule to defeat this problem. In this method, each node can capable of discover itself either it is a stuck node or not, irrespective of the existence of a sink node. In QUADRANT rule, we can assure that each individual node is capable to communicate in 360° depends on its communication range. The rule specifies that a node is a stuck node if where there no one 1-hop neighbor within the range of angle spanned by itself which is less than π/4. To realize a hole, every node executes the QUADRANT rule [6] to prove whether the node itself is a stuck node or not.

QUADRANT Rule: The communication range (R_c) of every node is partitioned into four quadrants of 90° each and verifies for the incidence of at least one 1-hop neighbor within a quadrant as shown in Fig. 2. If just this condition is fulfilled for all the four quadrants the node is not a stuck node besides it is called itself as a stuck node.

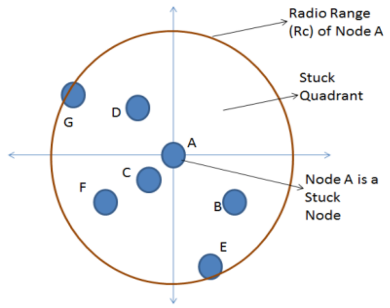

Fig. 2 Locating stuck node

3.2. Computation of Hole center

After stuck nodes are discovered by QUADRANT rule, they will verify its location information next to the existing boundary range of specified region. Thus the stuck nodes discriminate themselves from the boundary nodes of the specified region and can engage in hole boundary determination. Once a node discovers itself as a stuck node b_i, (boundary node for hole) it generates a Hole Discovery (HD) packet [7], consists its ID (afterward hole will use it as its hole ID) and sends it to the next stuck node b_{i+1}, here next stuck node is selected based on Right Hand Rule. Then b_{i+1} includes its location information with the received HD packet and resends it to the next one b_{i+2}. This process is continued up to the HD packet is travelled in the region of the hole and at last reached the originator node b_i.

In the HD packet 4 Boolean variables are included to discover the network boundary, X_{max}, Y_{max}, X_{min}, Y_{min}. A stuck node, which receives a HD packet, compares the coordinates defined in the packet with its coordinates, and if it finds that it has a higher or a lower value than one of these values compared to all its neighbors, it sets the corresponding Boolean variable to 1.

Once the HD packet reaches the initiator node (here node b_i), b_i gets the information of the all stuck nodes $\{b_0,b_1,....b_N\}$ from the acknowledged HD packet. Then it discovers two border nodes b_m and b_n so as to the space between them is the greatest distance than the distance between any two nodes in the set of border nodes. The calculation of distance between nodes b_m and b_n is,

$$Distance\ (b_m,b_n) = Max\{Distance(b_j,b_k)\ /\ b_jb_k \in \{b_0,b_1,....b_N\}\} \quad (1)$$

Then initiator node computes the hole center, it is the mid-point v of nodes b_m and b_n based on the formula,

$$x_v = (x_{bm} + x_{bn}) / 2 \quad (2)$$

$$y_v = (y_{bm} + y_{bn}) / 2 \quad (3)$$

The sensor which has the least Hole-ID removes the HD packet and elects itself as Hole Manager (HM). HM node will be responsible for the broadcasting of hole-healing message.

3.3. Hole Healing – HHA Determination

In hole healing process virtual force such as attractive and repulsive forces are involved. The attractive force is applied from the hole center and attracts the nodes towards it. In the same way, the repulsive force exists between nodes reduce the overlapping in between. HHA (Hole Healing Area) is a region in that the forces will be efficient. HHA performs healing process where only the nodes placed at a right distance from the hole will be involved in the process. The HM node that has the information about the size of the hole and border nodes computes the HHA and directs nodes on their movement. HHA will also decide the number of nodes that must be relocated to ensure a local repair of the hole. To locate the radius of the circle that describe HHA, an iterative approach is used with the formula,

$$R = r * (1 + \beta) \in R^{+} \quad (4)$$

Where r is the hole radius, β is a positive constant, which depends nodes density and the sensing range Rs.

For β = 0 in (4), R = r. The area defined by this circle (HHA-0) is equal to πr^2. The number of nodes required to heal HHA-0 is computed by,

$$\frac{\pi r^2}{\pi R_s^{\,2}} = \frac{r^2}{R_s^{\,2}} \quad (5)$$

If the number of nodes obtained by the HM node is fewer than the required number to cover the hole, the healing process may form a new hole. To keep away from this situation the HM node finds a new HHA circle by rising β, and continues this process until it finds an adequate amount of nodes to recover the hole. After the HHA determination, the HM node forwards a movement packet to the nodes contains information regarding the hole to repair. The nodes that get this packet will be coming into the relocation segment.

3.4. Hole Healing – Node Relocation

After HHA determination, the HM node informs regarding the healing process to the nodes involved in it. Nodes that accept forces from the hole center, shift towards it as shown in Fig.3.

The model of virtual forces used:

Attractive Force: The hole center v forces an attractive force on all node in the HHA and situated at a distance larger than d_{th}^{a} from v. A node p in the HHA receives an attractive force $\overrightarrow{F_a}(p,v)$ from v is known by,

$$\overrightarrow{F_a}(p,v) = \begin{cases} \frac{-k_a}{d(p,v) l_a} * e^{\frac{r}{d(p,v)}} \overrightarrow{u_a}, & d(p,v) > d_{th}^{a} \\ 0, & d(p,v) \le d_{th}^{a} \end{cases} \quad (6)$$

Where $\overrightarrow{u_a}$ is the unit vector oriented from the point p to the hole center v, d(p,v) is the Euclidean distance between the node p and the hole center located at v(x,y). l_a is a distance coefficient, k_a is a coefficient that defines the intensity of the attractive force and r is the hole radius.

The exponential factor reins the movement of nodes in the HHA, so that the nodes nearby the hole center will move longer distance than those on the boundary to stop the making of new holes during the current healing process, where the distance is inversely proportional to the force.

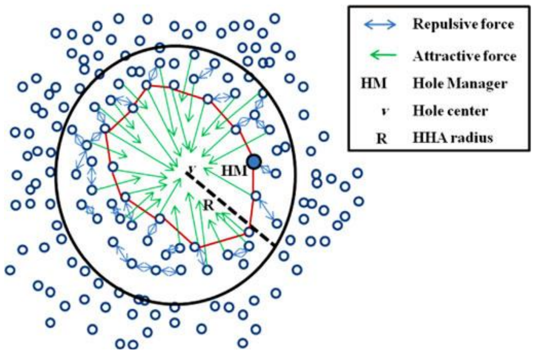

Fig. 3 Principle of the healing process

Repulsive Force: A repulsive force exists among nodes within the range given as $0 < \overrightarrow{F_r} < d_{th}^{r}$ is utilized to minimize the overlapped coverage between nodes, here $\overrightarrow{F_r}$ denotes repulsive force. If the distance between two nodes p and q is less than d_{th}^{r}, then $\overrightarrow{F_r}(p,q)$ increases as the distance d (p,q) decreases and is given by,

$$\overrightarrow{F_r}(p,q) = \begin{cases} \frac{-k_r}{d(p,q) l_r} \overrightarrow{u_r}, & 0 < d(p,q) < d_{th}^{r} \\ 0, & d(p,v) \ge d_{th}^{r} \end{cases} \quad (7)$$

Where $\overrightarrow{u_r}$ is the unit vector oriented from node q to p, k_r is a factor that defines the intensity of the repulsive force and l_r is a distance coefficient such as $l_r > l_a$.

Movement Equation: The final position of a node p is determined by the resultant force of the sum of all repulsive forces and the attractive force applied by the hole center. This force is denoted as $\overrightarrow{F_p}$,

$$\overrightarrow{F_p} = \sum_{q \in NS, p \neq q} \overrightarrow{F_r}(p,q) + \overrightarrow{F_a}(p,v) \qquad (8)$$

If $\overrightarrow{F_p}$ = 0 then the node p remains in its original position. Otherwise, p moves one time step in the direction forced by $\overrightarrow{F_p}$. The final position of p is given by,

$$\overrightarrow{P_p}(t+\Delta t) = \frac{\overrightarrow{F_p}}{\|\overrightarrow{F_p}\|} * \mathrm{V} + \overrightarrow{P_p}(t) \qquad (9)$$

Where V is the node velocity and $\overrightarrow{P_p}(t)$ is its position at instant t.

4. Energy Hole Problem

The sensor nodes within a certain neighborhood contribute in the processing of related events. The sensor nodes spend related energy costs on the same sensing job. We can deduce that the distribution of energy consumption for sensor nodes which ruin their residual energy is continuous. The neighboring sensor nodes which may exhaust their residual energy are called energy hole. Energy hole may fail sensor network and cause large coverage hole if the problem is not appropriately handled. When sensor nodes within energy hole run out of their energy and becomes coverage hole, the sensor network may fail to notice more jobs and come across risky injuries for surveillance application. In order to determine energy hole problem in sensor networks, a energy hole detection algorithm is planned to sense energy hole. When the region of energy hole is determined, mobile sensors can be used to heal energy hole.

While healing the coverage hole if we include the nodes which are having insufficient energy there is a possibility of formation of new hole called energy hole. That may also reduce the network coverage and the network connectivity. Therefore energy hole problem should also be considered while healing the coverage hole. The HM node broadcasts Energy Hole Information Collection (EHIC) packet to every node which are going to participate in the healing process in HHA region to know the residual energy information. After knowing energy information of nodes, the HM node again forwards moving packet to inform about healing process to only the nodes which are having sufficient energy.

5. Experimental Results

5.1. Verification of Heal

For verification, we have implemented HEAL in ns2 simulator [8]. The HEAL parameters are considered as in Table II.

TABLE II: HEAL Parameters

	Sub-scenario A	**Sub-scenario B**
Max (X,Y) (m,n)	(200,200)	(200,200)
Deployment	Deterministic	Deterministic/ Random
$R_s(m)$,$R_c(m)$	12, 24	12, 24
Simulation time (s)	500	500
Max speed (m/s)	10	10
Number of nodes	50	50
Routing protocol	DSDV	DSDV

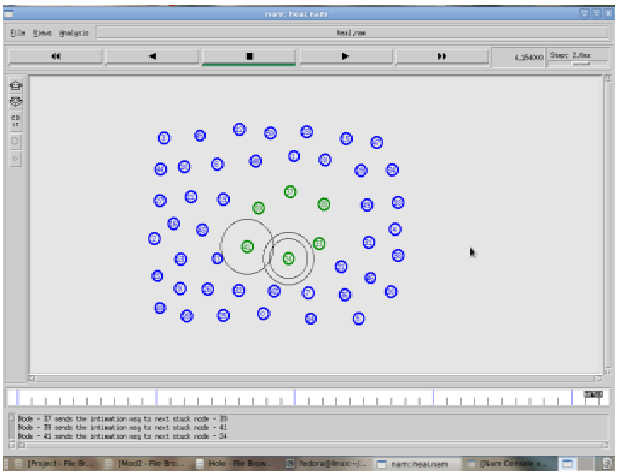

Fig. 4 Original network before healing process with coverage hole surrounded by green colour nodes

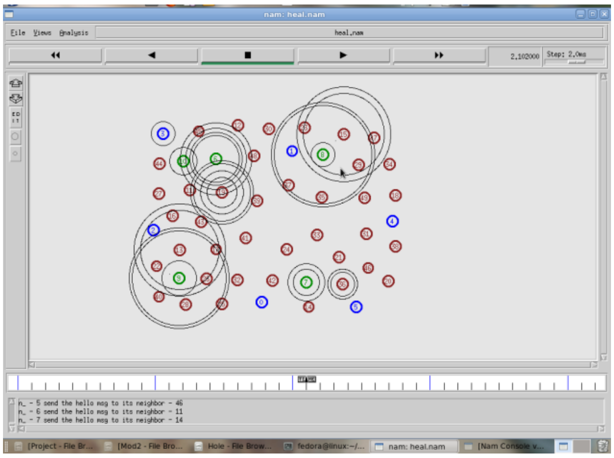

Fig. 5 Residual energy calculation

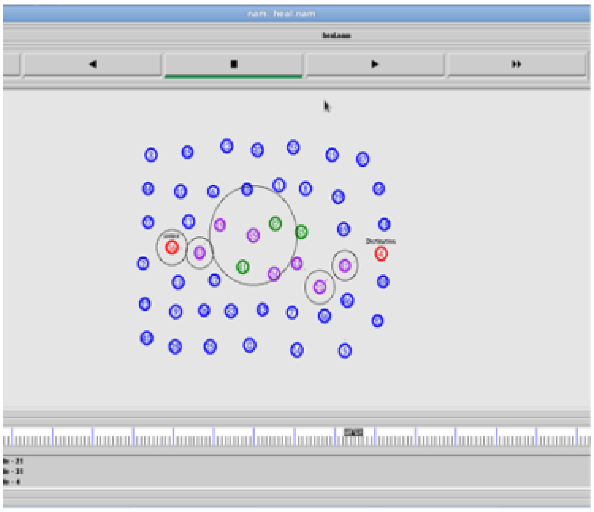
Fig. 6 Repaired network after healing process

6. Conclusion

In this paper we have proposed and implemented an algorithm HEAL, for detecting and healing coverage hole in WSN. Compared to the existing schemes, HEAL is a very low complexity algorithm and can handle holes of various forms and sizes irrespective of the nodes distribution and density. By exploiting the virtual force concept, our approach relocates only the sufficient nodes within the shortest distance and at the lowest cost. The experimental results shows that HEAL provide a cost-effective and an accurate solution for coverage improvement in WSNs. The modified HEAL algorithm can also handle the energy hole problem to further improve the network coverage.

REFERENCES

[1] Q. Fang, J. Gao, and L.J. Guibas, “Locating and Bypassing Holes in Sensor Networks,” Mobile Networks and Applications, vol. 11, no. 2, pp. 187-200, 2006.

[2] S. Yangy, M. Liz, and J. Wu, “Scan-Based Movement-Assisted Sensor Deployment Methods in Wireless Sensor Networks,” IEEE Trans. Parallel and Distributed Systems, vol. 18, no. 8, pp. 1108-1121, Aug. 2007.

[3] K. Mougou, S. Mahfoudh, P. Minet, A. Louiti ”Redeployment of Randomly Deployed Wireless Mobile Sensor Nodes” IEEE VTC2012, Quebec, Canada, September 2012.

[4] Z. Yong and W. Li, “A Sensor Deployment Algorithm for Mobile Wireless Sensor Networks,” Proc. 21st Ann. Int’l Conf. Chinese Control and Decision Conf. (CCDC ’09), pp. 4642-4647, 17-19 June, 2009.

[5] X. Li, H. Frey, N. Santoro, and I. Stojmenovic, “Strictly Localized Sensor Self-Deployment for Optimal Focused Coverage,” IEEE Trans. Mobile Computing, vol. 10, no. 11, pp. 1520-1533, Nov. 2011.

[6] Shirsat Amit and Bhargava Bhara, “Local geometric algorithm for hole boundary detection in sensor networks,” Security and Communication Networks, 2011, 4(9), pp. 1003–1012.

[7] Mustapha Reda Senouci, Abdelhamid Mellouk and Khalid Assnoune, “Localized Movement-Assisted Sensor Deployment Algorithm for Hole Detection and Healing,” IEEE Transactions on parallel and distributed systems, vol. 25, no. 5, may 2014.

[8] “NS2: Collaboration Between Researchers at UC Berkeley, LBL, USC/ISI, and Xerox PARC, The ns Manual,”http://www.isi.edu/nsnam/ns/ns-documentation, Feb. 2006.