www.s2is.org

International Journal on Smart Sensing and Intelligent Systems

ISSN 1178-5608

Self Organizing Sensor Network to Enhance Event Coverage

[+]S. Indu, ^Venu Kesam, ^Santanu Chaudhury, [+]Asok Bhattacharyya

[+]Delhi Technological University, Delhi

[^]Indian Institute of Technology, Delhi

Emails: s.indu@rediffmail.com

Submitted: October 16, 2010 ***Accepted: January 10, 2010*** ***Published: March 1, 2011***

Abstract- We are proposing a self deploying mobile sensor network which is empowered with event based relocation of redundant sensors for enhancing the quality of event sensing. Energy efficient Cell quorum based protocol is used for communication between event location and redundant sensors. Computationally light cascaded reorganization of sensor is suggested for relocation of sensors. The proposed method provides good coverage as it dynamically relocates the sensors either for avoiding coverage hole or for improving the quality of event sensing.

Index terms: Cell Quorum based protocol, redundant sensors, event detection, and relocation

I. INTRODUCTION

Sensor deployment has received considerable attention recently. When the environment is unknown or hostile such as remote harsh fields, disaster areas and toxic urban regions, sensor deployment cannot be performed manually. To scatter sensors by aircraft is one possible solution. However, using this technique, the actual landing position cannot be controlled due to the existence of wind and obstacles such as trees and buildings. Consequently, the coverage may be inferior to the application requirements no matter how many sensors are dropped [1], [2], [3]. Moreover, in many cases, such as during in-building toxic-leaks detection, chemical sensors must be placed inside a building from the entrance of the building. In such cases, it is necessary to make use of mobile sensors, which can move to the correct places to provide the required coverage. A self configuring Deployment algorithm with sensor relocation on event detection for a Mobile sensor networks is proposed. A Mobile sensor network can be considered as a collection of distributed sensor nodes, which are capable of sensing, moving, and communicating within its allowable range. Sensor networks inherently are different from standard communication networks as their aim is to monitor phenomena over space and time. The movement assist the nodes to self deploy and to self repair. Sensor networks are event-based systems that rely on the collective effort of densely deployed sensor nodes which continuously observe physical phenomenon. The main objective of a sensor network is to reliably detect / estimate event features from the collective information provided by sensor nodes. The energy consumption and processing constraints of small wireless sensor nodes can be reduced by collective sensing notion through networked deployment. Hence the collaborative nature of sensor nodes not only brings significant advantages over traditional sensing in accuracy of sensing, larger coverage area, and extraction of localized features but also the spatiotemporal correlation drastically enhance the overall network performance. For the deployment of Mobile sensor network, location of each node should be determined dynamically. For calculating the location of nodes relative to the field, a spatial addressing scheme for Mobile sensor Network is developed using hexagonal cell based management model. A hexagon represents a better coverage model for an Omni directional sensor node with a circular field, which also maximizes the coverage with minimum overlapping. In this model the coverage space is logically partitioned into several disjoint and equal sized hexagonal cell regions. Each cell is then given a cell-ID and the same can be calculated from the coordinates of sensor node. Cell quorum based protocol is suggested for communication between distributed sensors at event location and redundant sensors. As the sensors can compute their spatial address dynamically while moving, the optimal path for relocation can be easily determined. Uniform coverage, energy efficient protocol and cascaded movement of sensors with minimum moves make the proposed method efficient and reliable.

II. RELATED WORK

Many research works are being directed towards overcoming the common issues of sensor networks like self organizing, energy efficiency, event detection and coverage etc. and a considerable amount of research is being carried out on sensor deployment issue itself. A centralized virtual force based mobile sensor deployment algorithm (VFA) [4] [5] proposed by Y. Zou later modified by Jiming, combines the idea of potential field and disk packing [6]. This algorithm uses a powerful cluster head, which will communicate with all the other sensors, collect sensor position information, and calculate forces and utilizes these forces to position each sensor. Also this method assumes that the sensor node knows its initial location.

Zak Butler and et. al. in their work ”Event Based Motion Control” [7], discussed enhancement of coverage of mobile sensor network by replacing damaged sensors. They discussed different methods which were computationally complex and may not even converge for some probable cases. Their main assumption was complete initial coverage which may not be possible for all practical cases such as herd monitoring, environmental monitoring etc.. Further in their work ”controlling mobile sensors for monitoring events with coverage constraints” [8], they used veroni diagram to find out coverage holes, which is computationally intense. The algorithm proposed by Bin Zhang [9], randomly partitions space into sufficiently small neighborhoods at each iteration. Within each neighborhood are distribution process directed by a cluster head is enacted. As this is cluster based method it has high message complexity. Yee Ming Chen et.al [10] has proposed a similar work using fire tracker as an agent for event identification. It is assumed that sensor knows it’s relative location with respect to neighboring nodes. They also suggest movement of sensors in coalation with other sensors to enhance the effectiveness of extinguishing the fire, but the method used to move along with other sensors is not clear. The challenging issues of self organization is selective relocation of sensors, energy efficiency, event detection etc. Zihui Ge and et.al. proposed match making in their work ”Matchmaker” [11] they developed a pub/sub strategy so that nodes can publish their facilities / their needs on a common platform and can be shared effectively. Guiling Wang and et. al. used rectangular grid quorum based sensor relocation [12] where they identified redundant sensors for replacing coverage holes, and they applied pub=sub strategy for effective sensor relocation. They have used a rectangular grid based sensor deployment. As these nodes do not calculate their relative position in the field, routing the cascaded motion became computationally complex. Further the work is experimented by J.Teng [13] Wai-Leaong Yeow and et.ai in their work ”A novel target moving model and energy efficient target tracking in sensor network” [14], they prioritized the sensing area based on probability of presence of target and used for periodical sensing of the areas to achieve energy efficient sensing. N.A vasanthi and et. al. [15] have developed an energy efficient target tracking algorithm where they assumed that target will always enter from the boundary and travel towards the center and accordingly sleep schedule is planned for obtaining energy efficiency. Harouun rababaah and et. al in their work ”Gaurdm Duty Alarming Technique” [16] have assigned predefined frequency of monitoring for each node depending on sensor density at that location for energy efficient target tracking. Rahul et. al [17] have developed a fuzzy logic model for sensor deployment. They have considered remaining energy in each node as a parameter. But the nodes do not know their relative position in the field and possibility of coverage hole is quiet high. In the work “LF Indoor Location and Identification System” Antti Roppen et.al. [18]developed a location identification system using floor based antennas. In our work we used simple triangulation method for location identification which is simple and cheaper. In their work Jiann-Liang Chen and et. al, “Adaptive Routing Protocol for Reliable Sensor Network Applications” [20] developed redundancy node and dual routing protocol for Adhoc networks. We used cell quorum based protocol as it suits for communication among nodes among a row and column.

Most of the existing works assume that the mobile sensors know its relative position in the field. In the proposed work, we used spreading algorithm for sensor deployment, as it not only assures complete coverage but also helps the node to calculate its relative position in the field dynamically. The energy efficient Cell quorum based protocol for communication, between redundant sensors and event location, and cascaded movement of sensor nodes make the sensor network efficient and reliable. Hence we are presenting a self organizing mobile sensor network which assures complete coverage and good quality of sensing.

III. PROBLEM STATEMENT

The basic issues of mobile sensor networks are localization, coverage and energy efficiency. The global locations of each sensor node can be easily obtained by using GPS enabled nodes, but no literature is available to calculate its relative position in the field. Proper deployment algorithm can provide good coverage, but mobile sensor networks are more susceptible to coverage holes as well as sparsely covered event location. The time taken for healing a coverage hole using iterative algorithm is very high and hence is not suited for event based relocation. We developed a spreading algorithm which computes relative positions of sensor nodes dynamically, identify event location and redundant sensors, fast and automatically heals coverage holes by doing minimum moves. Cell Quorum based communication protocol is used for communication between redundant sensors and sensor node at event location. Optimal relocation of redundant sensors is accomplished by doing minimum moves as the sensor nodes are aware of their relative location in the field. The energy efficient protocol and minimum movement of sensor nodes helps to improve the life of the network

A. Spreading Algorithm

The spreading algorithm provides complete coverage [20]. A brief description is given in this paper for proper understanding. This algorithm needs the area of the sensing field as reference for localization of sensor nodes. This can be given as the range of Latitude and Longitude in case of an open field or as coordinates in case of a closed field. The coverage space is logically partitioned into several disjoint and equal-sized hexagonal cellular regions. The size of the hexagon depends on the coverage capacity of sensor used. Each cell is then assigned a unique Cell-id (Cx, Cy) relative to the field as shown in fig. 1. For the given (x,y) co-ordinates ((x,y) coordinates of the sensor node) , the corresponding cell-id (Cx, Cy) of the node is calculated by comparing the node coordinates with the coordinates of the central point of each cell. The coordinates of each node can be calculated either from satellite (GPS) or from 3 Beacons placed locally as shown in fig. 2.

The nodes are made ’ON’ randomly. Once a node is ’ON’ it will broadcast it’s cell ID and coordinates, then it will listen from other nodes. If it is not hearing from any other node of the same cell, it will elect itself as the Cell Head otherwise it will register itself as a slave node. Once a head node is elected then all the other nodes in that cell will become slaves and will listen the messages from cell heads of the neighboring cells. If it is not hearing any message from the neighboring cell head, that cell will be considered as an unoccupied cell. Once adjacent unoccupied cells are identified the head node will move redundant sensors from its own cell to the adjacent un occupied cells. If there are no redundant cells then the head node itself will move to one of the adjacent un occupied cell. These steps will be repeated till all the cells are covered. The hexagonal representation of cells helps us to cover the given field with minimum number of nodes with minimal overlapping area for Omni directional sensing pattern. Minimum message flooding due to local communication of sensors and energy efficient cascaded movement of sensors are the specialty of the spreading algorithm.

B. Event based Relocation

The deployment algorithm will provide 100 percent coverage if the number of nodes is greater than or equal to the number of cells. If the number of nodes is greater than the number of cells, the occupancy of some cells will be more than one. The cell head of these nodes record the number of redundant sensors present around it in the same cell. As the sensors are moving, the cell head may also be changing continuously. Hence it is necessary to update the no. of redundant sensors in a cell periodically. These redundant sensors can be effectively utilized to enhance the quality of event sensing by relocating them to the event location. If the strength of sensed quantity of any sensor is more than a threshold value, it will record it as an event and communicate the same to the cell head. The cell head will look for redundant sensors using cell quorum based communication protocol. The cell head of event location and cell head of redundant sensors collectively make the cascading schedule. Proper selection of cascading schedule makes the movement energy efficient. As the space is equally divided and each sensor is moving only one cell distance, the schedule with minimum no. of nodes will be optimal.

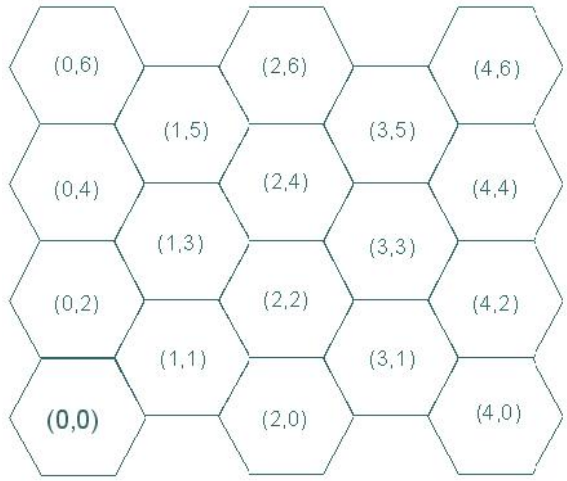

Fig-1 Hexagonal Space Tessellations

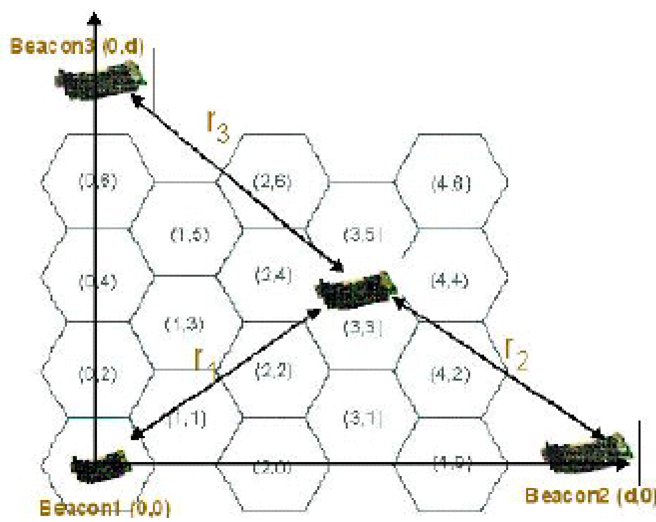

Fig-2 Localization

IV SELF ORGANIZING SENSOR NETWORK

A. Spreading Algorithm

Let N be the total number of cells in the given area whose side is given by ”S” O_i be the occupancy at a iteration i with a given probability of electing head be P_{head}

$$O_i = \sum_{k=1}^{N} C_i^k \quad (1)$$

$$C_i^k = \begin{cases} 1 \; if \; cell \; is \; occupied \\ 0 \; if \; cell \; is \; not \; occupied \end{cases} \qquad (2)$$

O_i is the initial occupancy

Now it can be shown that the following inequality satisfies at any time instant

$$O_i \leq O_j \; \forall_i \leq j \qquad (3)$$

The occupancy may decrease on due to the following reasons

- All the adjacent cell of a void cell is occupied with a single sensor node and if more than one head node send node to the void cell, the occupancy reduces
- If all the adjacent cells of a void cell is occupied with normal node, the algorithm will skip step 5-9 and there will not be any change in occupancy and the occupancy reduces

The reduction in occupancy due to the first reason is avoided by notifying the victim node to all adjacent nodes. The reduction in occupancy due to second reason will be solved by itself which is explained as given below

Given probability of the node being head is P_{head} . In the cell the probability of the node to in normal state is P_{normal}

Hence

$$P_{head} + P_{normal} = 1 \qquad (4)$$

$$P_{normal} = 1 - P_{head} \qquad (5)$$

After k iterations the probability of the node being the head at least once is given by

$$P_{at\;least\;once\;head} = 1 - P_{normal\;K} \qquad (6)$$

$$P_{at\;least\;once\;head} = 1 - (1 - P_{head}) \qquad (7)$$

Hence for large value of k the P_{head} will become 1 and hence at least one node will become head, then that head node will go through the steps 5-9 which will fill the empty cell hence increasing the occupancy.

If the number of nodes is less than number of cells, the occupancy O_i will remain constant and it will not satisfy (2) the algorithm will not converge. The nodes will always be in motion, providing dynamic coverage.

a. Algorithm:

H_i^k represents the head node of a cell k at i th iteration and P is the set of current node locations. A_i^k is the adjacent cell information of the head cell H_i^k and V_i^k are the victim nodes that are targeted to the empty neighboring cells of the cell C_k by the head node H_i^k

1. Partition the sensing field into small sub cells C_k having a regular pattern (hexagonal);
2. i=0;
3. While termination conditions are not satisfied do;
4. Select a cell head H_i^k from the nodes which are moved to a cell C_k;
5. Each node n within a cell sends position P(n) to the cell head;
6. Each cell head H_i^k learn the neighboring cells information and constructs the adjacency list A_i^k
7. Each cell head H_i^k selects the victim nodes V_i^k that are to be sent to the neighboring cells $C_j = C_{nbrs}$
8. Assign each victim node a new position;
9. Notify the adjacent cells with positions of the victim nodes;
10. All the Victim nodes will move to the new cells;
11. i = i + 1

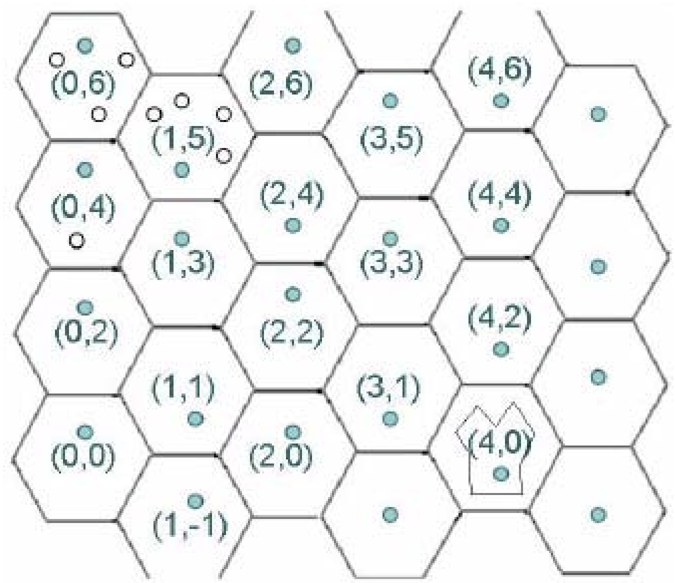

Fig. 3. Relocation System Model

B. Redundant Sensor Location

Fig.3 illustrates the sensor relocation problem of a sensor network with hexagonal cells. The black nodes are used to represent cell heads. Each cell is indexed by a tuple, whose first number is used to represent the column and the second number is used to represent the row. Cells (0, 6), (1, 5), and (0, 4) have redundant sensors. When there is an event at cell (4, 0), its cell head first needs to locate the available redundant sensors and should move these sensors to the event location for improving the quality of sensing. The deployment protocol itself can be used for sensor relocation by moving neighboring cells to the event location. But this will disturb the whole system and takes time to stabilize, which in turn wastes lot of energy. To avoid this we have to move the redundant sensors to the event location without disturbing the stability of the system. Which can be achieved by using Cell quorum based communication protocol and subsequent diagonal movement of redundant sensors. G.Wang and et. al [12] devised the same logic for a self organizing sensor network to heal coverage holes, as the nodes do not know their relative positions, formation of cascading schedule was computationally intense in that case.

b. *Cell quorum based communication protocol*;

Cell Quorum based communication protocol is used to advertise the location of redundant sensors and request for more sensors at the event location by the respective cell heads. By organizing cells as quorums, each advertisement and each request can be sent to a quorum of cells. Due to the intersection property of quorums, there must be a cell which is the intersection of the advertisement and the request. The cell head will be able to match the request to the advertisement. A simple publisher quorum and subscriber quorum can be constructed by choosing all the nodes along the same row and column. For example, as shown in Fig.3., suppose cell (1,5) has redundant sensors, it sends the advertisement to cells in a row ((1,5), (3,5), (5,5), (7,5), (9,5)) and a column ((1,3), (1,1), (1,-1)).When cell (4,0) is looking for redundant sensors, it only needs to send a request cells ((4,0), (3,1), (2,2), (1,3)) and ((3,-1), (4,0), (5,1)). The intersection node (1, 3) will be able to match the request to the advertisement. Suppose N is the number of cells in the network. By using this quorum based system, the message overhead can be reduced from N to $\sqrt{N}$. The message overhead is very low compared to flooding. We can further reduce the message overhead by observing the specialty of our problem. This can be even extended to multiple event detection.

C. Sensor movement;

Having obtained the location of the redundant sensors, we need to determine how to move the redundant sensors to the target location (destination). It is very important that the movement should be completed within the time frame of the event i.e. before the event dies out. We should look for cascaded movement instead of direct movement as it will reduce power consumption as well as time frame.

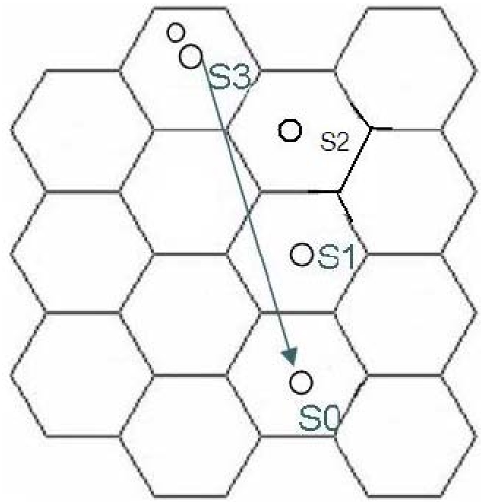

Fig.4. Direct movement of sensors

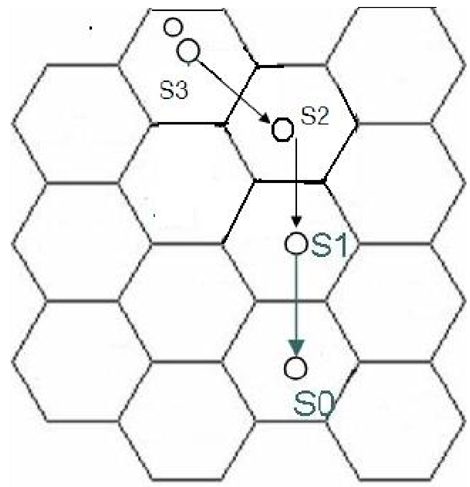

Fig.5. Cascaded movement of sensors

c. Cascaded reorganization of sensors

As shown in fig.4, instead of letting the redundant sensor S_3 move directly to the destination. S_1 and S_2 are chosen as cascading schedule as shown in fig.5. As a result, S_3 moves to replace S_2, which in turn moves to replace S_1 and S_1 moves to the destination. Since the sensors can first exchange communication messages (logical move) and form a cascading schedule to (physically) move at the same time, the relocation time is much shorter. S_0 is the cell head of event location. Apart from this, as the movement is for shorter distances, energy consumed by the individual sensors will be minimum.

The cascaded movement of sensors is very fast, as each sensor is moving to the nearby cell simultaneously. Hence the relocation can be achieved within the affordable time delay. The spreading algorithm assures complete coverage of the sensor network. By any chance, if a sensor dies out, the neighboring cell head will run relocation algorithm and replaces the damaged sensor. As each sensor is moving for a shorter duration, the method will be energy efficient. This algorithm will be very effective, if the number of sensors is more than number of cells. If number of sensors is less than number of cells, sensor relocation may not be effective; still the deployment algorithm will give dynamic coverage.

The set of cascading nodes for relocation and their departure time together is defined by a cascading schedule. For example, in fig.8, $S_3(t_3)$, $S_2(t_2)$, $S_1(t_1)$, S_0 is a cascading schedule, which can be used to increase the coverage of the event. The cascading schedule should ensure that the number of movements should be minimum and it should happen well within the required time of event. Each movement should also consider the restriction of recovery delay of individual node. i e. movement should satisfy both conditions. As each sensor knows its cell ID and its coordinates relative to the field, the number of movements can be minimized by moving along the intermediate diagonal cells (by incrementing or decrementing both coordinates by 1 depending on the event location and redundant sensor location) till the node reaches the corresponding column/row of the event location and there after moving along the cells of the column/row of the event location. If the event and redundant sensors lie on same row or column, the schedule should be made along the corresponding row/column. This method not only minimizes the movement but also avoids collision of sensors. If there are redundant sensors in more than one cell, the nearest one is identified from its location. The sensor at the event location initiates the formation of cascading schedule by broadcasting the allowable time delay, its cell ID and cell ID of redundant sensor. The neighboring cell head will respond it if it is on the diagonal path or in the same row/ column of the redundant sensor.

d. Algorithm for cascaded movement.

E_i represents the head node of an Event cell and R_m j is the head cell of one of the 'm' cells having redundant sensors. The event cell head will execute cascaded movement algorithm.

1. Calculate the Eucleadean distance between the redundant cell head and event cell head, and select the redundant cell head with minimum distance;
2. Compare the cell-ID of cell having redundant sensors and event cell;
3. Decide movement path;
4. Execute the movement;

V. SIMULATION RESULTS

A. Spreading Algorithm

Initially we verified the spreading algorithm by simulating the same and later on in continuation we verified event detection and sensor relocation. We have developed a java based simulator for simulating freely moving mobile nodes. The simulator can also be used to view the topology generated while executing the algorithm. The simulator assumes no packet collisions. It also assumes that there are no packet errors during transmission and reception. In other words, we assume a perfect wireless channel. Fig.9 shows the GUI panel for the simulator in which it provides two select buttons which can be used for selecting different initial distribution of the nodes. By default it will have the random distribution of sensor nodes. It provides a time slider at the bottom of the panel which can be used to slide the time epochs, we can drag the time slider to a point so that it will show the corresponding node distribution at the time epoch. The simulator runs for 2500 time epochs. At every time epoch we will get the previous state of the node and the node will executes the algorithm based on the previous state, then the current state will be saved again.

For simulation, 140 nodes in an area of 1025 X 850 sq.units are used. The side of the hexagon is varied from 70 units to 100 units. For each Case 3 different initial configurations are considered. Simulation results are shown in fig.6. for

- Uniform Distribution: All the cells are occupied with a single node
- Random Distribution: Nodes are distributed in a random fashion
- Single Distribution: All nodes are placed in a single cell
- The probability of P_{head} is changed from 0.5 to 0.7

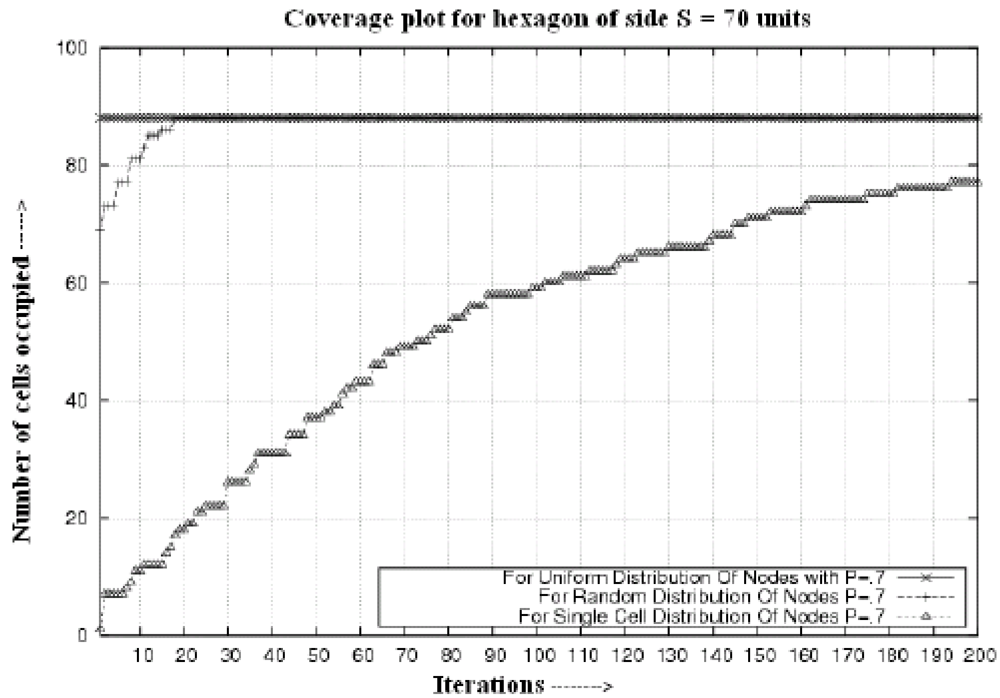

Fig. 6. Coverage plot

B. Event detection and Sensor relocation

Simulations were carried out to find out average battery remaining with variable number of nodes for comparing with Naive protocol. We compared the power consumed by the sensor nodes and

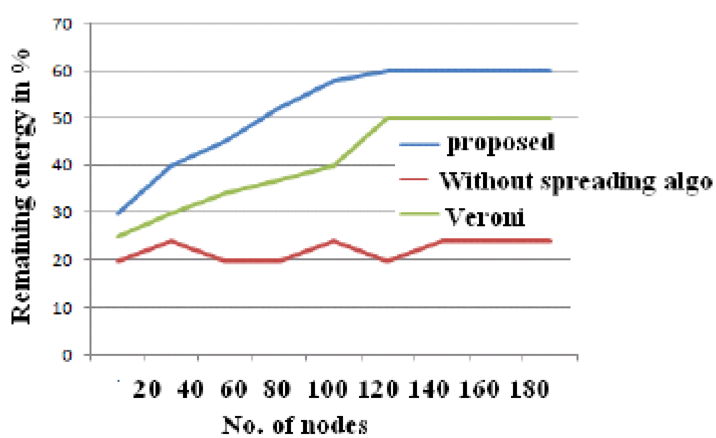

Fig.7. Plot of remaining battery vs number of nodes

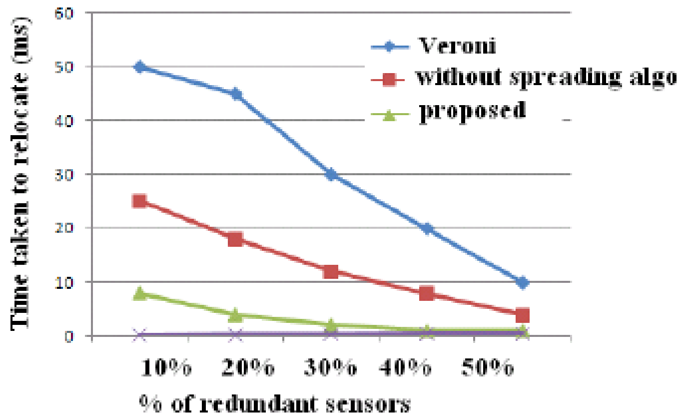

Fig.8. Response time vs number of nodes

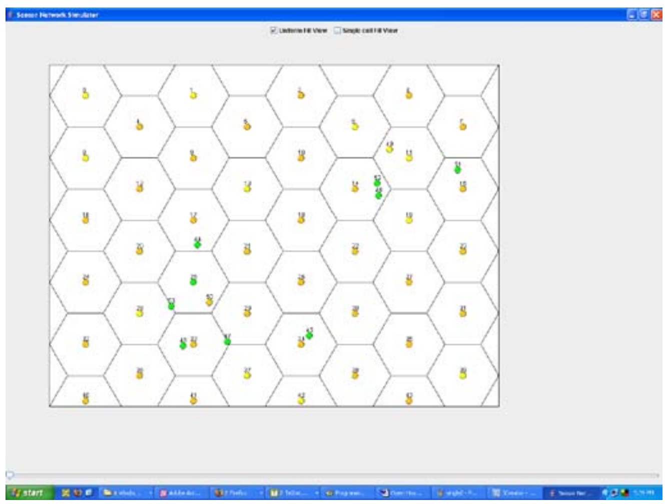

Fig 9(a) Distribution of nodes after running deployment algorithm

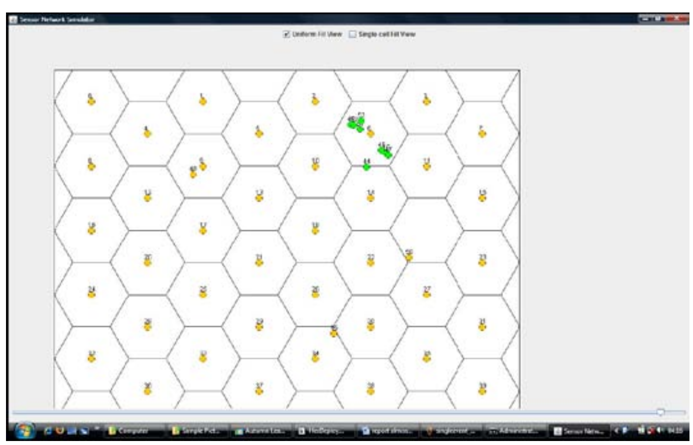

Fig. 9(b). Location of sensor nodes after event based sensor relocation.

response time for relocation with self organizing algorithm (Veroni diagram based) and also with sensor relocation without spreading algorithm for deployment and found that with increase in no. of nodes the average battery remaining is increasing as shown in fig. 7 and fig. 8 . We assumed energy consumption based on distance travelled by the node. Single event detection and sensor relocation is simulated and shown in fig. 9(a) and fig. 9(b). The deployment algorithm gave complete coverage and the redundant sensors moved to the event location.

VI. EXPERIMENTL EVALUATION

For experimental evaluation a sensor node, a range of sensors for monitoring various phenomena and a mobility platform are used. The first two have been purchased from vendors where as the mobility platform was developed in the lab. The cricket motes from Crossbrow are used, as it can be programmed for both sensing computational task as well as location awareness. MPR410 Sensor Nodes with MIB510 programming board as shown in fig: 10 are used for validation. Three separate software were used to implement the system.

The firmware is uploaded on three motes which served as Beacons. This firmware was slightly modified and used as a Listener. Finally a single node was programmed as a packet forwarder for the coverage panel application. The BeeBots are used (as shown in fig. 11) to provide mobility or the nodes. BeeBots have been inspired from CotsBot [21] and have been similarly built by modifying toy car as shown in fig. The BeeBots are configured using TinyOs [22] for interfacing with PIC microcontroller for providing mobility.

Three Beacons and four mobile nodes were used. We developed a coverage panel in Java for displaying, in real time, the actual location of sensor nodes. 3 cricket beacons were used for localization of nodes which are represented as black dots and mobile nodes are represented as red nodes. The distribution of nodes after running deployment algorithm is shown in fig.13. We used nesC to embody the structuring concepts of TinyOS operating system. Three separate software were used to implement the system. Firstly the firmware supplied was uploaded on 3 motes which served as Beacons. This firmware was slightly modified to use as a node which ran our model. Finally a single node was programmed as a packet forwarder for coverage panel. This packet forwarder acts as an interface between sensor nodes and serial port of computer so that we can see the respective positions of all sensors in the field. We programmed the nodes to handle self deployment, localization, event detection and relocation themselves. Fig. 14 and fig.15 gives the intermediate location and final positions of mobile nodes after implementation of DHM based selforganizing algorithm.

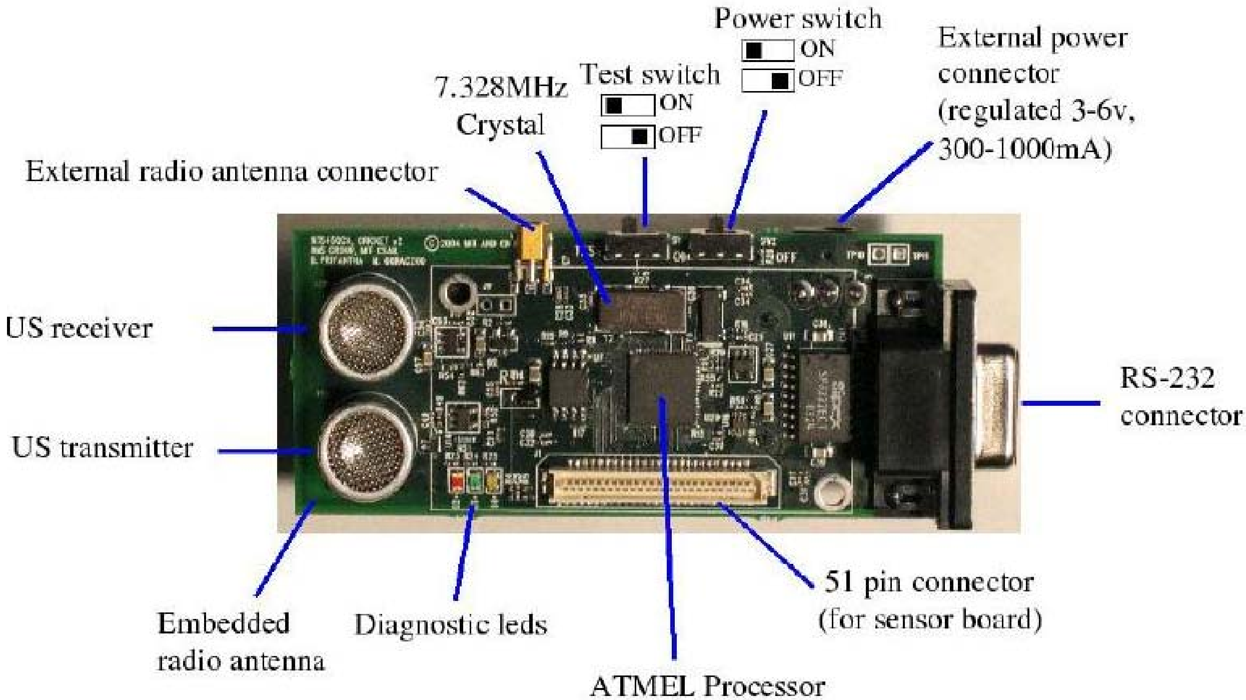

Fig. 10 MPR410 Sensor Node

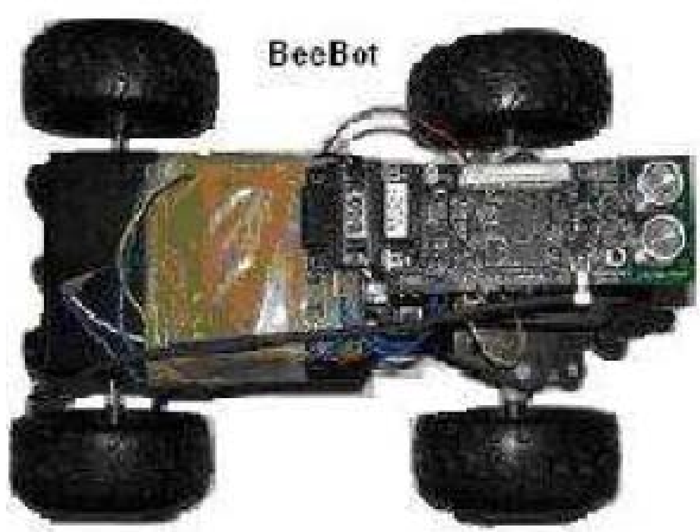

Fig.11 A sample Beebot

Fig.12. Experimental Setup

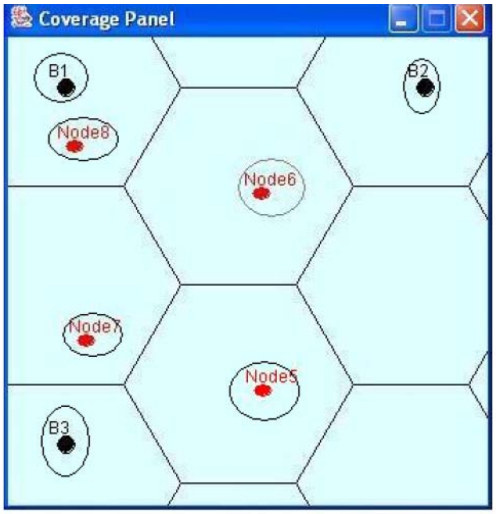

Fig. 13 The locations of the sensors after running deployment algorithm

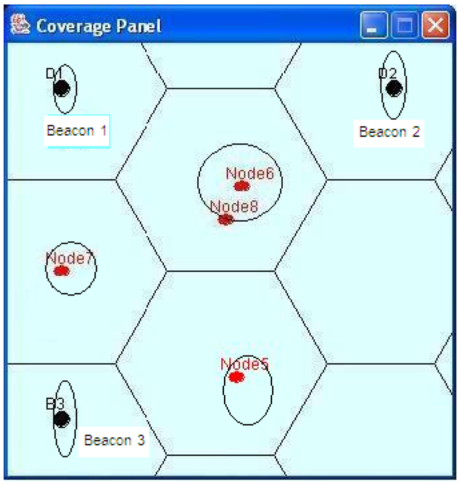

Fig. 14 Intermediate location of sensor nodes

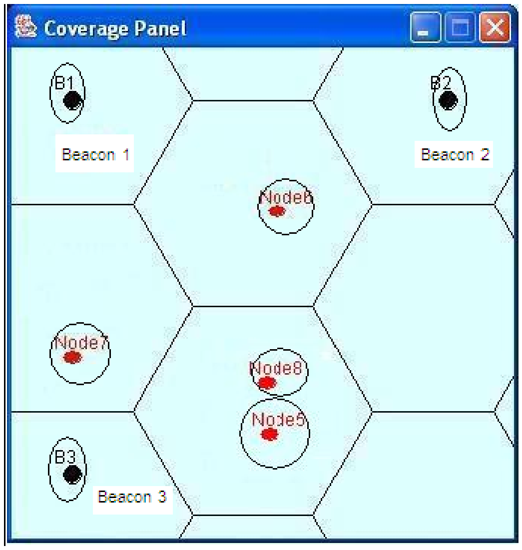

Fig.15. Final location of nodes.

VII. CONCLUSION

We have presented a novel distributed Self organizing sensor network which heals coverage holes and provides accurate information by increasing the coverage at event location using energy efficient relocation of redundant sensors. The simulation results show the correctness of the algorithm. The algorithm is computationally light and scalable. For simulation purpose we have developed a java simulator to display the position of the individual sensor nodes.

We have also verified the simulated results using experimental evaluation. We programmed the motor of remote car and interfaced it with the sensor node for mobility. Then we run the self organizing algorithm for deployment and sensor relocation. The results are self explanatory. This work will be instrumental for sensing disaster fields which are inaccessible for human beings.

REFERENCES

[1]. A. Howard, M. J. Mataric, and G. S. Sukhatme, “An Incremental Self-Deployment Algorithm for Mobile Sensor Networks,” Autonomous Robots Special Issue on Intelligent Embedded Systems 13(2),Sept 2002, pp. 113--126

[2] J. A. Paradiso and T. Starner, “Energy Scavenging for Mobile and Wireless Electronics,” IEEE Pervasive Computing January, pp. 1568 – 1573, 2005.

[3] K. M. Krishna, "'A Surveillance System based on Multiple Mobile Sensors," Proceedings of FLAIRS, Special Track in Sensor Fusion, pp 1-4, 2004.

[4] Y. Zou and K. Chakrabarty, "Sensor Deployment and Target Localization based on Virtual Forces," ACM Transactions on Embedded Computing Systems, Vol. , No. , 20, Pages 1–30.

[5] J. Chen, S. Li, and Y. Sun, "Novel Deployment Schemes for Mobile Sensor Networks," In International Journal SENSORS Volume 7, pp. 2907 – 2919, 2007.

[6] J. H. Reif and H. Wang, "Social potential fields: A distributed behavioral control for autonomous robots," Robotics and Autonomous Systems, vol. 27, pp. 171 – 194, 1999.

[7] Z. Bulter and D. Rus, "Event based motion control for mobile sensor network," IEEE Pervasive Computing**,** Volume 2 Issue 4, October 2003, pp 34-42.

[8] Z. Bulter and D. Rus, "Controlling Mobile Sensors for Monitoring Events with Coverage Constraints," In Proc. of the IEEE International Conference on Robotics, pp. 1568 – 1573, 2003.

[9] B. Zhang and G. Sukatame, "Controlling ensor density using mobility, " Proceedings of 2nd IEEE workshop on Embedded Networked Sensors, pp. 141 – 149, 2005.

[10] Y. M. Chen and S.-H. Chang, "Purposeful Deployment via Self-Organizing Flocking Coalition in Sensor Networks, " International Journal of Computer Science and Applications volume 4 isue 3, pp. 84 – 94, 2007.

[11] J. K. Z. Ge, P. Ji and D. Towsley, "Matchmaker: Signaling for dynamic publish/subscribe applications," Proceedings of the 11th IEEE International Conference on Network Protoc1ols (ICNP'03) 1092-1648/03 $17.00 © 2003 IEEE

[12] G. Wang, G. Cao, and T. L. Porta, "Sensor relocation in mobile sensor network," In Proceedings of 23rd International Conference of IEEE Communications Society, pp. 1– 11, March 2004.

[13] J. Teng, T. Bolbrock, G. Cao, and T. L. Porta, "Sensor Relocation with Mobile Sensors Design Implementation and Evaluation," 1-4244-1455-5/07/2007 IEEE, 2007.

[14] W. L. Yeow, C. K. Tham, and W. C. Wong, "A novel target movement model and energy efficient target tracking in sensor network," 0-7803-8887-9/05/IEEE, 2005.

[15] N.A.Vasnthi, S.Annadurai, and S. Krishna, "Energy saving schedule for target tracking sensor networks to maximise the network life time," 0-7803-9575-1/06/IEEE, 2006.

[16] H. Rababaah and A. Shirkhodaie, “Gaurd duty alarming technique: A novel scheduling approach for target tracking in large scale distributed sensor network,” 1-4244-1160-2/07/IEEE, 2007.

[17] R. Mathur, M. K. Sharma, A. Misra, and D. Baveja, “Energy-Efficient Deployment of Distributed Mobile Sensor Networks Using Fuzzy Logic Systems,” In proceedings of 2009 International Conference on Advances in Computing Control and Telecommunication Technologies, pp. 121 –125, 2009.

[18] Antti Ropponen, Matti Linnavuo, Raimo Sepponen “LF Indoor Location and Identification System” International Journal on Smart Sensing and Intelligent Systems, vol. 2, no. 1, march 2009, pp 94-117.

[19] Jiann-Liang Chen, Yu-Ming Hsu and I-Cheng Chang “Adaptive Routing Protocol for Reliable Sensor Network Applications” International Journal on Smart Sensing and Intelligent Systems, vol. 2, no. 4, December 2009, pp 515-539

[20] S.Indu, Venu Kesam and Asok Bhattacharyya “SELF Deployment OF Mobile SensorNetwork for optimal coverage” International Journal of Engineering Science and Technology Vol. 2(7), 2010, 2968-2975

[21]“The cotsbot project homepage,” Website, http://wwwbsac.eecs.berkeley.edu/projects/cotsbots/index.html.

[22] P. Levis, “Tinyos programming,” Website, February 2006, http://www.tinyos.net/.