Coverage-driven Self-deployment for Cluster Based Mobile Sensor Networks

Xiaoling Wu, Lei Shu, Min Meng, Jinsung Cho*, and Sungyoung Lee
Department of Computer Engineering, Kyung Hee University, Korea
{xiaoling, sl8132, mengmin, sylee}@oslab.khu.ac.kr, chojs@khu.ac.kr

Abstract

Coverage, energy efficiency and connectivity are important metrics for sensor network deployment. In this paper, we propose a relay shift based approach to eliminate coverage hole due to the initial random dropping or the existence of faulty sensors. The sensors in our model are assumed to have only limited mobility. Firstly, the optimal number of clusters are determined by energy function and the optimal cluster head positions are determined by Fuzzy C-mean (FCM). We use proposed Relay Shift Based Algorithm (A-RSBA) for movement assisted sensor deployment. A* algorithm is applied to find a shortest path from a redundant sensor node to a virtual node point in an uncovered area, and each sensor moves along this path by relay shift based on the principle that evenly distributed sensors can provide better coverage. Simulation results show that our approach can improve coverage within a short time and limited movement distance as well as ensuring connectivity and energy efficiency.*

1. Introduction

Wireless sensor networks are expected to be widely utilized in the future since they can greatly enhance our capability for monitoring and controlling the physical environment [1]. Due to the inevitable relationship with the physical world, the proper deployment of sensor nodes is very important for the successful completion of the sensing tasks.

Sensor deployment has received considerable attentions recently. Some of the research work [2, 3] assume that the environment is sufficiently known and under control. However, when the environment is unknown or hostile, sensor deployment cannot be performed manually. To drop 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 not be able to satisfy the application requirements. Some researchers suggest simply deploying large amount of static sensors to increase coverage; however it often ends up harming the performance of the network [4]. Moreover, in many cases, such as in-building toxic-leaks detection [5], chemical sensors must be placed inside a building from the entrance of the building. In such cases, it is necessary to take advantage of mobile sensors, which can move to the appropriate places to provide the required coverage.

To address this issue, a class of work has recently appeared where mobility of sensors is utilized to achieve desired deployment [6~10]. Typically in such works, the sensors detect lack of desired deployment objectives. The sensors then estimate new locations, and move step by step to the final positions. For example, in [8], the authors present the virtual force algorithm (VFA) as a new approach for sensor deployment to improve the sensor field coverage after an initial random placement of sensor nodes. The cluster head (CH) executes the VFA algorithm to find new locations for sensors to enhance the overall coverage. They also considered unavoidable uncertainty existing in the precomputed sensor node locations. This uncertainty-aware deployment algorithm provides high coverage with a minimum number of sensor nodes. While the above works are quite novel in their approaches, the mobility of the sensors in their models is assumed unlimited.

In fact, the mobility of sensors is limited in most cases. To this extent, a class of Intelligent Mobile Land Mine Units (IMLM) [11] to be deployed in battlefields have been developed by DARPA. The IMLM units are employed to detect breaches, and move with limited mobility to repair them. This mobility system is based on a hopping mechanism that is actuated by a single-cylinder combustion process. The hop distance is dependent on the amount of fuel and the propeller dynamics. Some other techniques can also provide

* Corresponding author

Proceedings of The Sixth IEEE International Conference on Computer and Information Technology (CIT'06)
0-7695-2687-X/06 $20.00 © 2006 IEEE

such kind of mobility, for instance, sensors supplied by spring actuation etc. This type of model normally trades off mobility with energy consumption [11]. Moreover, in many applications, the latter goals outweigh the necessity for advanced mobility, making such mobility models quite practical in the future. [11] is one of the very few papers which deal with the mobility limited deployment optimization. The mobility in the sensors they consider is restricted to a flip. However coverage is the only considered objective in their paper and their approach is not feasible in network partition case.

In this paper, we design and evaluate our proposed Relay Shift Based Algorithm (A*-RSBA) for mobility limited sensor self-deployment. In our model, sensors can move only a hop at a time to a new location, i.e., the moving distance is bounded by a certain value (we use transmission range which makes sense in terms of connectivity). A certain number of mobility limited sensors are initially deployed in the sensor network. The sensor nodes are clustered; the optimal number of cluster heads is chosen based on energy function and positions are determined by FCM. The initial deployment may not cover all regions in the network. Regions that are not covered by any sensors are coverage holes. In this framework, our problem is to determine an optimal movement plan for the sensors in order to maximize the network coverage and simultaneously minimize the total number of sensor movements, so that the energy consumption is reduced at the same time. We use A* algorithm to find a shortest path from a redundant sensor to the virtual node point in a coverage hole, and design relay shift based sensor deployment protocol based on the principle of moving sensors from densely deployed areas to sparsely deployed areas.

The rest of the paper is organized as follows. Section 2 introduces the energy efficient clustering method. In section 3, we present the proposed Relay Shift Based Algorithm (A*-RSBA) for mobile nodes self-deployment. Section 4 evaluates the performance of the proposed method and compares with related work. Based on the simulation results, we justify our design and discuss future work in Section 5.

2. Energy-efficient clustering

2.1 Determination of optimal number of CHs

Since centralized computation is not suitable for WSNs especially when the network is large scale, we execute our proposed movement assisted algorithm in a clustered topology. We use a WSN model with nodes homogeneous in their initial amount of energy and assume that all nodes are distributed randomly over the sensor field. We particularly present in this section how the optimal number of clusters can be computed.

This clustering is optimal in the sense that energy consumption is well distributed over all sensors and the total energy consumption is minimum. Such optimal clustering highly depends on the energy model used. For the purpose of this study we use similar energy model and analysis as proposed in [12].

According to the radio energy dissipation model, in order to achieve an acceptable Signal-to-Noise Ratio (SNR) in transmitting an l bit message over a distance d, the energy expended by the radio is given by:

$$E_T(l,d) = \begin{cases} lE_{elec} + l\varepsilon_{fs}d^2 & if\ d \le d_0 \\ lE_{elec} + l\varepsilon_{mp}d^4 & if\ d > d_0 \end{cases} \tag{1}$$

where E_{elec} is the energy dissipated per bit to run the transmitter or the receiver circuit, ε_{fs} and ε_{mp} are amplifier constants, and d is the distance between the sender and the receiver. By equating the two expressions at d=d_0, we have $d_0 = \sqrt{\varepsilon_{fs} / \varepsilon_{mp}}$.

To receive l bit message, the radio expends:

$$E_R(l) = lE_{elec} \tag{2}$$

Assume an area $A = M \times M$ square meters over which n nodes are uniformly distributed. For simplicity, assume the sink is located in the center of the field, and that the distance of any node to the sink or its cluster head is $\le d_0$. Thus, the energy dissipated in the cluster head node during a round is:

$$E_{CH}(l) = (\frac{n}{n_c} - 1)lE_{elec} + \frac{n}{n_c}lE_{DA} + lE_{elec} + l\varepsilon_{fs}d_{toBS}^2 \tag{3}$$

where n_c is the number of clusters, E_{DA} is the processing (data aggregation) cost of a bit per report to the sink, and d_{toBS} is the average distance between the cluster head and the sink. The energy used in a non-cluster head node is equal to:

$$E_{nonCH}(l) = lE_{elec} + l\varepsilon_{fs}d_{toCH}^2 \tag{4}$$

where d_{toCH} is the average distance between a cluster member and its cluster head. The expected squared distance from the nodes to the CH is given by:

$$E[d_{toCH}^2] = \frac{M^2}{2\pi n_c} \tag{5}$$

The energy dissipated in a cluster per round is:

$$E_{cluster} = E_{CH} + (\frac{n}{n_c} - 1)E_{nonCH} \approx E_{CH} + \frac{n}{n_c} E_{nonCH} \quad (6)$$

The total energy dissipated in the network is:

$$E_{total} = l(2nE_{elec} + nE_{DA} + \varepsilon_{fs}(n_c d_{toBS}^2 + nd_{toCH}^2)) \quad (7)$$

By differentiating E_{total} with respect to n_c and equating to zero, the optimal number of constructed clusters can be found [13]:

$$n_{c-opt} \approx \sqrt{\frac{n}{2\pi}} \frac{M}{d_{toBS}} = \sqrt{\frac{n}{2\pi}} \frac{2}{0.765} \quad (8)$$

2.2 Optimal clustering

Fuzzy C-mean (FCM) algorithm was introduced by Bezdek and applied later in sensor clustering in [14]. The FCM based algorithm is a data clustering technique wherein each data point belongs to a cluster to some degree that is specified by a membership grade. The algorithm is as follows:

1. *Initialize* $U=[u_{ij}]$ *matrix,* $U^{(0)}$
2. *At k-step: calculate the centers vectors* $C^{(k)}=[c_j]$ *with* $U^{(k)}$
   $$c_j = (\sum_{i=1}^{N} u_{ij}^m \cdot x_i)/(\sum_{i=1}^{N} u_{ij}^m)$$
3. *Update* $U^{(k)}$, $U^{(k+1)}$
   $$u_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}}$$
4. *If* $\| U^{(k+1)} - U^{(k)} \| < \varepsilon$ *then stop; otherwise return to step 2.*

Where u_{ij} is the degree of membership of x_i in the cluster j, x_i is the i^{th} of d-dimensional measured data, c_j is the d-dimension center of the cluster. Once the center of the cluster is identified, if there is no node in that position, the one nearest to the CH location will become a CH. However, we avoid the boundary positioned nodes if there exist several such nodes near to the center of the cluster.

3. Proposed deployment approach: A*-RSBA

Let $G(V, E)$ be the graph defined on V with edges $uv \in E$ iff $uv \leq R$. Here uv is the Euclidean distance between nodes u and v, R is the communication range. We have 4 steps for implementing A*-RSBA:

Step 1: Randomly deploy nodes in the network.

Step 2: Detect coverage holes and redundant sensor nodes. We set two distance threshold value T_1 and T_2. If the longest distance between two nodes A and B along the uncovered area perimeter is larger than T_1, regard it as a coverage hole, and create a virtual node point at the center of straight line AB. If the distance between two neighbors is less than T_2, regard them as redundant nodes. Choose a redundant node nearest to the virtual node point in coverage hole.

Step 3: Use A* algorithm [15] to find a shortest path n_0-n_1-n_2-…-n_{n-1} from a redundant sensor n_0 to the destination n_{n-1} (added virtual node) in a coverage hole. The distance between n_{n-2} to n_{n-1} is bounded by R. A* algorithm is the most popular choice for pathfinding, because it's fairly flexible and can be used in a wide range of contexts. A* was developed to combine heuristic approaches like Best-First-Search (BFS) and formal approaches like Dijsktra's algorithm. A* is like Dijkstra's algorithm in that it can guarantee a shortest path, while BFS cannot; and it is like BFS in that it works as fast as BFS which is faster than Dijsktra's algorithm. Take the advantage of A* algorithm, we can solve our problem more efficiently than our previous work [16] in which Dijsktra's algorithm was applied.

Step 4: Move sensor node n_{n-2} to the virtual node n_{n-1}, move n_{n-3} to n_{n-2} … finally move the redundant sensor n_0 to n_1, and leave the original location of sensor n_0 empty. The nodes coordinates can be updated by equation (9):

$$NetLoc(n_i) = NetLoc(n_{i+1}), \quad i = 0, 1, \cdots, n-2 \quad (9)$$

$n_i \in$ nodes on shortest path from source to destination

n_0=source node

n_{n-1}=destination (virtual node)

The process is illustrated in Fig. 1 using an example of 4 sensors and 1 virtual node along the shortest path. Sensor node n_3 moves to the virtual node point n_4, n_2 moves to n_3 … finally the redundant sensor n_0 moves to n_1, and leave the original location of n_0 empty. The network coverage is defined as the ratio of the union of areas covered by each node and the area of the entire ROI. It can be calculated using Monte-Carlo method by meshing the ROI as has been done in [10].

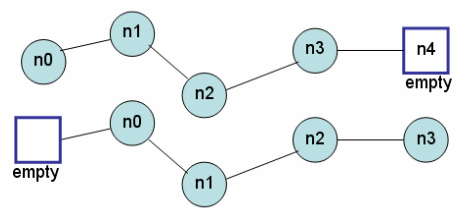

Figure 1. Illustration of sensor nodes relay shift along the shortest path

4. Performance evaluations

4.1 Optimal clustering results

For FCM based optimal CH determination, MATLAB fuzzy logic toolbox [17] is used. 30 nodes are randomly placed in a region of size 10×10. The communication range of each sensor node is 4 units with a fixed remote base station at (5, 20). The optimal number of clusters in the problem space is determined by Eq. (8), so it is 5 here. The nodes are organized into clusters by the base station. Once the position of the CH is identified, if there is no node in that position, the one nearest to the CH location will become a CH. Here the CHs determined are nodes labeled 6, 9, 26, 27, and 28, as shown in Figure 2.

4.2 Sensor movement experiment results

The performance of the proposed movement assisted algorithm A*-RSBA is evaluated by simulation. For the convenience of comparison, we set the initial parameters the same as in [9]: 30 randomly placed nodes in a region of size 10×10 are used for initial deployment; the sensing range r and communication range R used in the experiment are 2 and 4 m, respectively.

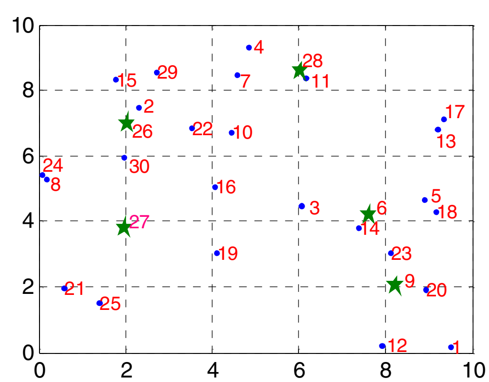

Figure 2. CH determination in initial random deployment (CHs are highlighted)

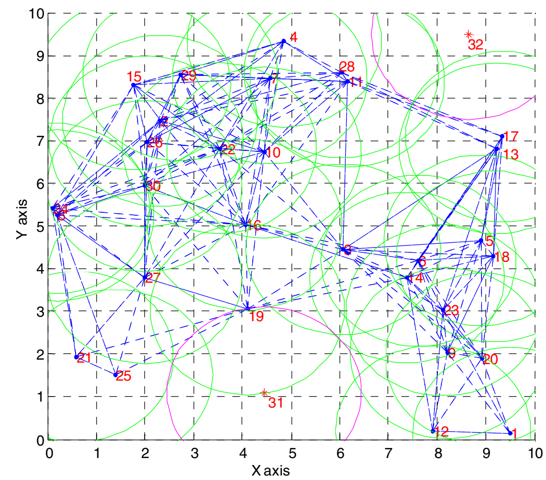

Figure 3. Determine virtual node point in uncovered area and redundant nodes

In Figure 3, the initial random deployment is represented by sensors 1~30. The green circles are used to show the sensing range r of the nodes. Communications are possible between nodes that are connected by a dashed line. Fig. 3 also shows the detected virtual node points by labeled points 31 and 32 in coverage hole and the redundant nodes nearest to 31 and 32 by 14 and 17 respectively. Both the coverage holes and the redundant nodes are judged by CHs. This information is then broadcasted by CHs to the whole network. The parameter values needed are: T_1=1.2 and T_2 = r/4.

Figure 4 shows the two shortest paths found (14→19→31 and 17→32) by A* algorithm from redundant nodes to virtual node points. This is also actual path of individual nodes as they move by relay shift, in which sensor node move only one hop at a time which can guarantee the connectivity. Each node moves a distance of 2.6157 on average and the standard deviation of distance traveled is 0.5714. When the average distance traveled is small, the corresponding energy for locomotion is small. Also, when the standard deviation of distance traveled is small, the variation in energy remaining at each node is not significant and a longer system lifetime with desired coverage can be achieved. Figure 5 shows the final node positions with desired coverage=0.9923 after executing A*-RSBA. Note that the original 30 sensor nodes are finally reorganized and relabeled.

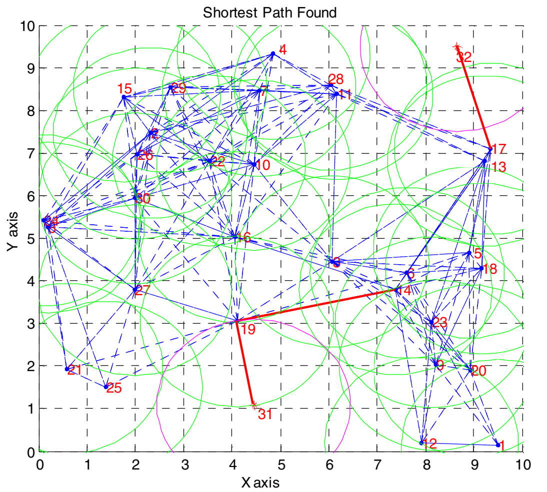

Figure 4. Find shortest path by A* algorithm from redundant node to virtual node point

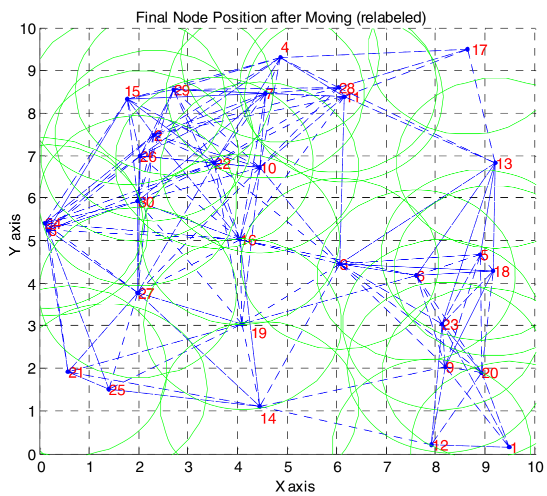

Figure 5. Final node positions after executing proposed deployment algorithm

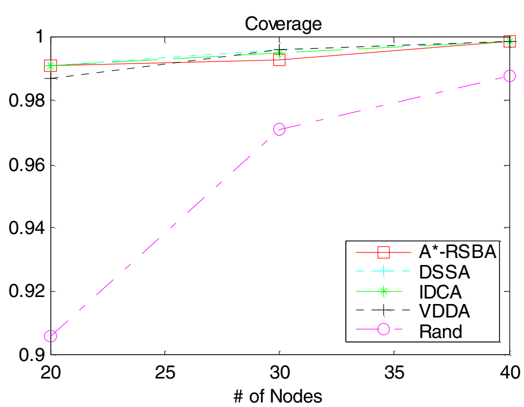

Figure 6. Coverage comparison

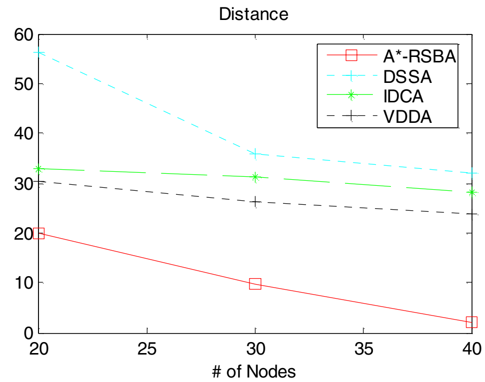

Figure 7. Total distance traveled comparison

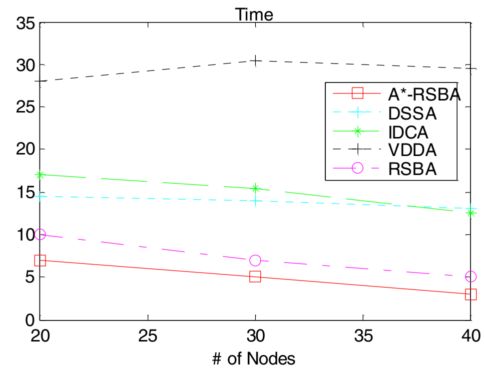

Figure 8. Termination time comparison

Next, the performances of A*-RSBA are compared with DSSA, IDCA, and VDDA [9] in terms of coverage, movement distance until convergence, and time. Since A* algorithm finds the same shortest path as Dijkstra's algorithm, we only compare A*-RSBA with RSBA in terms of time. Results are presented in Figures 6~8. These results are obtained for different number of nodes dispersed over a fixed ROI of size 10×10, i.e., for different node densities to examine the relation between node densities and the performance metrics. The number of nodes varies from 20 to 40 and results are averaged over 10 runs (initial random distributions) for each node density.

Figure 6 shows the improvement in coverage area from the initial random deployment for A*-RSBA, DSSA, IDCA, and VDDA. All four algorithms exhibit a similar performance. Although the coverage of A*-RSBA (99%~1) is not always the highest among 4 algorithms, this number is often satisfactory for many application requirements.

Figure 7 shows the significant reduction of total distance traveled by A*-RSBA compared with other 3 algorithms. In fact, distance moved here is used as the indicator of energy consumption. In A*-RSBA, only very few numbers of nodes need to move and each sensor movement is bounded by only one hop. However, almost every node needs to move in the

Proceedings of The Sixth IEEE International Conference on Computer and Information Technology (CIT'06)
0-7695-2687-X/06 $20.00 © 2006 IEEE

other 3 algorithms. So it is obvious that our proposed algorithm can save much more energy compared with related methods. Figure 8 shows that A*-RSBA leads to faster deployment than DSSA, IDCA, VDDA and RSBA. Termination time is measured in the number of iterations until the algorithms stop.

5. Conclusion and future work

In this paper, we proposed a self-deployment approach using sensors with limited mobility. More specifically, sensors can move only one hop at a time to a new location, i.e., the moving distance is bounded by transmission range. After initially deploying a certain number of sensors in the ROI, the sensors were clustered, the optimal number of CHs was chosen based on energy function and positions were determined by FCM. We determined an optimal movement plan by proposed A*-RSBA algorithm for the sensors to improve network coverage and simultaneously minimize the total number of movements. A* algorithm was used to find a shortest path from a redundant sensor to the virtual node point in a coverage hole, and mobility limited sensors move by relay shift along this path. Based on simulation, we evaluated and compared our approach A*-RSBA with other related works from various aspects: coverage, total moving distance (as an indicator of energy consumption), and deployment time, and show that A*-RSBA is very effective in terms of these standards.

In the future work, we will address varying sensing ranges and investigate such cases. Moreover, the uniformity problem will be further studied.

6. Acknowledgement

This work was supported by grant No. R01-2005-000-10267-0 from Korea Science and Engineering Foundation in Ministry of Science and Technology.

7. References

[1] Xiaoling Wu, Hoon Heo, Riaz A. Shaikh, J. Cho, O. Chae, and S. Lee, "Individual Contour Extraction for Robust Wide Area Target Tracking in Visual Sensor Networks", *ISORC*, IEEE Computer Society, 2006

[2] S. Meguerdichian, F. Koushanfar, G. Qu and M. Potkonjak, "Exposure in Wireless Ad-Hoc Sensor Networks", *Mobicom,* 2001.

[3] T. Clouqueur, V. Phipatanasuphorn, et al, "Sensor Deployment Strategy for Target Detection", *WSNA,* 2002.

[4] Sameer Tilak , Nael B. AbuGhazaleh, and Wendi Heinzelman, "Infrastructure Tradeoffs for Sensor Networks", *WSNA,* 2002.

[5] 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,* September, 2002.

[6] J. Wu and S. Wang, "Smart: A scan-based movement-assisted deployment method in wireless sensor networks", *Proc. IEEE INFOCOM Conference,* Miami, March, 2005.

[7] G. Wang, G. Cao, and T. La Porta, "Movement-assisted sensor deployment", *Proc. IEEE INFOCOM,* 2004.

[8] Y. Zou and K. Chakrabarty, "Sensor deployment and target localization based on virtual forces", *Proc. IEEE INFOCOM Conference,* Vol. 2, 2003, pp. 1293-1303.

[9] Nojeong Heo and Pramod K. Varshney, "Energy-Efficient Deployment of Intelligent Mobile Sensor Networks", *IEEE Transactions on Systems, Man, and Cybernetics—Part A: Systems And Humans,* Vol. 35, No. 1, 2005, pp. 78 – 92.

[10] Xiaoling Wu, Shu Lei, Yang Jie, Xu Hui, Jinsung Cho and Sungyoung Lee, "Swarm Based Sensor Deployment Optimization in Ad hoc Sensor Networks", *ICESS' 05/ LNCS (SCIE),* Springer, 2005, pp. 533-541.

[11] Sriram Chellappan, Xiaole Bai, Bin Ma, and Dong Xuan, "Sensor Networks Deployment Using Flip-based Sensors", *Proc. of IEEE International Conference MASS*, 2005.

[12] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on Wireless Communications,* vol. 1, no. 4, pp. 660–670, October 2002.

[13] Georgios Smaragdkis, Ibrahim Matta and Azer Bestavros, "SEP: A Stable Election Protocol for clustered heterogeneous wireless sensor networks," *Proc. of the Int'l Workshop on SANPA,* 2004.

[14] Halgamuge, M.N.; Guru, S.M.; Jennings, A., "Energy Efficient Cluster Formation in Wireless Sensor Networks", *10th ICT,* Vol 2, Issue, 23, 2003, pp. 1571–1576.

[15] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, "*Introduction to Algorithms"*, 2nd Edition. MIT Press and McGraw-Hill, 2001, pp. 595–601.

[16] Xiaoling Wu, Yu Niu, Lei Shu, Jinsung Cho, Y. Lee and S. Lee, "Relay Shift Based Self-Deployment for Mobility Limited Sensor Networks", *Proc. of UIC/ LNCS*, 2006.

[17] http://www.mathworks.com/