Relay Shift Based Self-deployment for Mobility Limited Sensor Networks

Xiaoling Wu, Yu Niu, Lei Shu, Jinsung Cho*, Youngkoo Lee, and Sungyoung Lee

Department of Computer Engineering, Kyung Hee University, Korea
{xiaoling, niuyu, sl8132, sylee}@oslab.khu.ac.kr,
{chojs, yklee}@khu.ac.kr

Abstract. In this paper, we propose a relay shift based approach to solve uneven sensor distribution problem due to the initial random dropping or the existence of faulty sensors. The distinguishing feature of our work is that the sensors in our model have limited mobility. After determining the optimal cluster head positions by particle swarm optimization (PSO) method, we use proposed Relay Shift Based Algorithm (RSBA) for movement assisted sensor deployment. Dijkstra's algorithm is applied to find a shortest path from a redundant sensor 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 provide high 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 of monitoring and controlling the physical environment. Due to the inevitable relation with the physical world, the proper deployment of sensors is very important for the successful completion of the sensing tasks.

Sensor deployment has received considerable attention recently. Some of the work [1], [2], [3] assume that the environment is sufficiently known and under control. However, when the environment is unknown or inhospitable such as remote inaccessible areas, disaster fields and toxic urban regions, sensor deployment cannot be performed manually. To scatter sensors by aircraft is one possible solution. However, using this scheme, 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 during 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.

* Corresponding author.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 556 – 564, 2006.

To address this issue, a class of work has recently appeared where mobility of sensors is utilized to achieve desired deployment [6], [7], [8], [9], [10]. Typically in such works, the sensors detect lack of desired deployment objectives, then estimate new locations, and move to the resulting locations. 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. Specifically, if a sensor node chooses to move to a desired location, it can do so without any limitation in the movement distance.

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. For each hop, the fuel is metered into the combustion chamber and ignited to propel the IMLM unit into the air. The hop distance is limited, depending on the amount of fuel and the propeller dynamics. The units contain a righting system to orient itself properly after landing, and a steering system that provides directional control for movement. Some other techniques can also provide such kind of mobility, for instance, sensors supplied by spring actuation etc. This type of model normally trades-off mobility with energy consumption [12, 13]. Moreover, in many applications, the latter goals outweigh the necessity for advanced mobility, making such mobility models quite practical in the future. [13] 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 (RSBA) for mobility limited sensor self-deployment. In our model, sensors can move only one 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 sensors nodes are clustered and optimal CH positions are chosen using PSO which is borrowed from our previous work [10]. 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. We use Dijkstra's 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 CH positioning method. In section 3, we present the proposed Relay Shift Based Algorithm (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 Technical Preliminary: Particle Swarm Optimization

Particle Swarm Optimization (PSO) is an evolutionary computing technique based on the principle of bird flocking. In PSO a set of particles is initialized randomly. Each particle will have a fitness value, which will be evaluated by the fitness function to be optimized in each generation, and knows its best position *pbest* and the best position so far among the entire group of particles *gbest*. The particle will have velocities, which direct the flying of the particle. In each generation the velocity and the position of the particle will be updated. The equation for the velocity and positions are given below as (1) and (2) respectively,

$$v_{id} = \varpi \times v_{id} + c_1 \times rand() \times (p_{id} - x_{id}) + c_2 \times rand() \times (p_{gd} - x_{id}) \tag{1}$$

$$x_{id} = x_{id} + v_{id} \tag{2}$$

where ϖ is the inertia weight, and c_1 and c_2 are acceleration coefficients.

PSO shares many similarities with Genetic Algorithm (GA), however, due to the inexpensive computation in terms of both memory requirements and speed, we choose PSO as the optimization method.

2.2 Determination of Optimal Cluster Head Positions

The model of mobile sensor network is presented as follows. We assume that each node knows its position in the problem space; it is possible by using some localization method [14]. All sensor members in a cluster are homogeneous and cluster heads (CHs) are more powerful than sensor members. Sensing coverage of each node is assumed to have a circular shape without any irregularity. The design variables are 2D coordinates of the sensor nodes, $\{(x_1, y_1), (x_2, y_2), \ldots\ldots\}$.

We intend to minimize energy usage in a cluster based sensor network topology by finding the optimal CH positions. For this purpose, we assume a power consumption model [15] for the radio hardware energy dissipation where the transmitter dissipates energy to run the radio electronics and the power amplifier, and the receiver dissipates energy to run the radio electronics. This is one of the most widely used models in sensor network simulation analysis. Both the free space (*distance*^{2} power loss) and the multi-path fading (*distance*^{4} power loss) channel models are used here. Assume that the sensor nodes inside a cluster have short distance *dis* to CH but each CH has long distance *Dis* to the base station. Thus for each sensor node inside a cluster, to transmit an *l*-bit message a distance *dis* to CH, the radio expends

$$E_{TS}(l, dis) = lE_{elec} + l\varepsilon_{fs} dis^2 \tag{3}$$

For CH, however, to transmit an l-bit message a distance Dis to base station, the radio expends

$$E_{TH}(l, Dis) = lE_{elec} + l\varepsilon_{mp} Dis^4 \tag{4}$$

In both cases, to receive the message, the radio expends:

$$E_R(l) = lE_{elec} \tag{5}$$

Here we set electronics energy as E_{elec}=50nJ/bit, whereas the amplifier constant, is taken as ε_{fs} =10$pJ/bit/m^2$, ε_{mp} = 0.0013$pJ/bit/m^2$. Since the energy consumption for communication is the most significant, we neglect sensing and computation energy consumption here.

Assume m clusters with n_j sensor members in the j^{th} cluster C_j. We derive the fitness function as in [10]:

$$f = \sum_{j=1}^{m} \sum_{i=1}^{n_j} (0.01 dis_{ij}^2 + \frac{1.3 \times 10^{-6} Dis_j^4}{n_j}) \tag{6}$$

3 Proposed Deployment Approach: RSBA

Let $G(V, E)$ be the graph defined on V with edges $uv \in$E iff $uv \le R$. Here uv is the Euclidean distance between nodes u and v, R is the communication range.

We have 4 steps for implementing RSBA:

Step 1: Randomly deploy nodes in the network.

Step 2: Detect coverage holes and redundant sensor nodes. We set 2 distance threshold value T_1 and T_2. If the longest distance between 2 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 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 the widely used Dijkstra's algorithm [16] 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. Since Dijkstra's algorithm was designed to solve the single-source shortest path problem for a directed graph with nonnegative edge weights, it is feasible here.

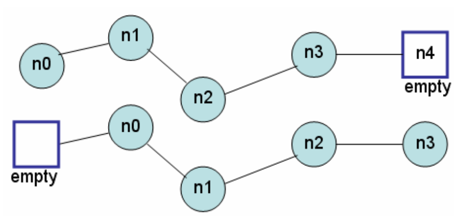

Fig. 1. Illustration of sensor nodes relay shift along the shortest path

```
1. Initialization
  initial_node_locations (netXloc, netYloc);
  sensing_range r;
  communication_range R;
 If      distance (i, j) <= R
        link i and j;
2. Create Virtual Nodes and Redundant Nodes
  calculate the longest length of 2 points A & B along the hole arc;
  detect coverage holes;
  calculate the center point C of edge AB; % C becomes Virtual Nodes
 If      distance(i, j) <= T2
           If   distance (i, C) < distance(j, C)
                define i as source and C as destination;
3. Shortest Path Finding by Dijkstra's algorithm
Function [path, totalCost] = dijkstra (m, netCostMatrix, s, d)
% m: number of nodes in the network, s: source node index, d: destination node index,
% path: node sequence of shortest path, totalCost: distance along shortest path
4. Sensor Nodes Movement
For k=1: length (path)-1
netloc (k) = netloc (k+1);
Update nodes link;
Calculate network coverage
```

Fig. 2. Pseudocode of the proposed RSBA method for sensor nodes reorganization

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 (7):

$$NetLoc(n_i) = NetLoc(n_{i+1}), \qquad i = 0, 1, \cdots, n-2 \tag{7}$$

$n_i \in$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 four sensors and one 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].

The pseudocode of the proposed algorithm is illustrated in Fig 2.

4 Performance Evaluations

4.1 Optimal Clustering Results

For PSO based optimal CH determination, a linear decreasing inertia weight value ϖ from 0.95 to 0.4 is used, and acceleration coefficients c_1 and c_2 are set to 2 according to [10]. The coordinates of potential CHs are set as particles in the sensor network. The communication range of each sensor node is 4 units with a fixed remote base station at (5, 20). The minimum number of clusters acceptable in the problem space is 2, but we choose 3 here. The nodes are organized into clusters by the base station. Fitness value is evaluated by the fitness function (6) in each generation. Our purpose is to find the optimal location of CHs. Once the position of the CH is identified, if there is no node in that position then the one nearest to the CH location will become a CH. Here the CHs determined are nodes labeled 27, 23 and 29, as shown in Fig 3.

4.2 Sensor Movement by Relay Shift: Experimental Results

The performance of the proposed movement assisted algorithm 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 r and R used in the experiment are 2 and 4 m, respectively. In Fig. 3, the node locations and coverage of the initial random deployment before running the algorithms are shown. Tiny points with red numerical label beside represent the positions of nodes and green circles are used to show the r of the nodes. Communications are possible between nodes that are connected by a dashed line. Sensor information can be collected within the r and communications between nodes are possible within the R. The initial coverage is 0.9273.

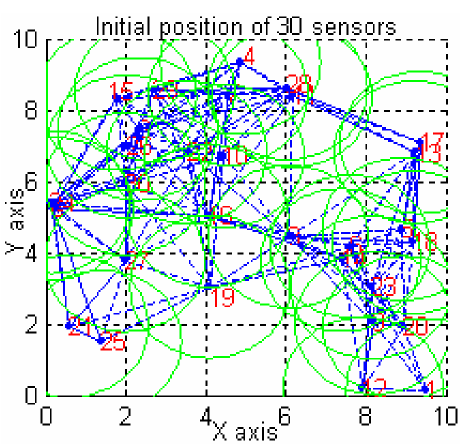

Fig. 3. Initial random deployment with sensing range 2m and communication range 4m

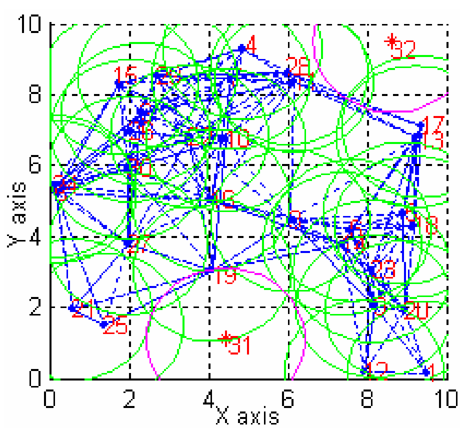

Fig. 4. Determine virtual node point in uncovered area and redundant nodes

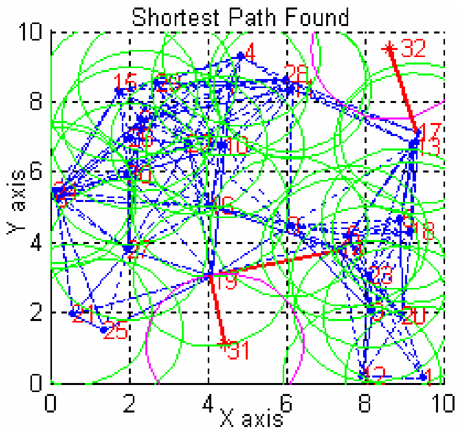

Fig. 5. Find shortest path by Dijkstra's algorithm from redundant node to virtual node point

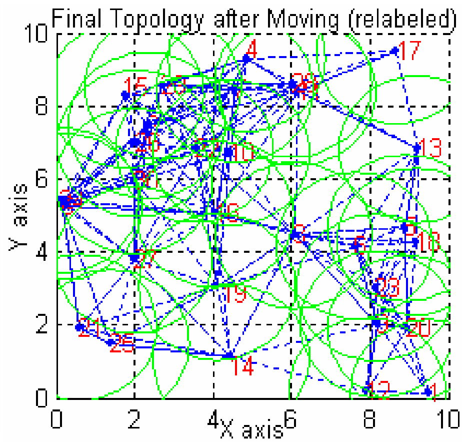

Fig. 6. Final node positions after executing proposed movement-assisted deployment algorithm

Fig. 4 shows the detected virtual node points (labeled as 31 and 32) in coverage hole and the redundant nodes nearest to 31 and 32 are 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: threshold value T_1=1.2 and T_2 = r/4.

Fig. 5 shows the 2 shortest paths found (14→19→31 and 17→32) by Dijkstra's 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 guarantees the connectivity. For the initial distribution of Fig. 3, 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. Fig 6 shows the final node positions with desired coverage=0.9923 after executing RSBA. Note that the original 30 sensor nodes are finally reorganized and relabeled.

Next, the performances of RSBA are compared with DSSA, IDCA, and VDDA [9] in terms of coverage, standard deviation of distance, movement distance until convergence, and time. Results are presented in Figs. 7–10. Fig. 7 shows the improvement in coverage area from the initial random deployment for RSBA, DSSA, IDCA, and VDDA. All four algorithms exhibit a similar performance. Although the coverage of RSBA (≈99.2%) is slightly lower than other 3 algorithms, this number is often satisfactory for many application requirements. Fig. 8 shows RSBA has lower standard deviation of distance compared with others. It means the variation in energy remaining at each node is small, so that longer lifetime can be achieved. Fig. 9 shows the significant reduction of total distance traveled by RSBA compared with other 3 algorithms. In 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 other 3 algorithms. Fig. 10 shows that RSBA leads to faster deployment than

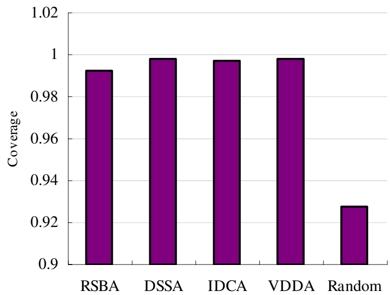

Fig. 7. Coverage comparison

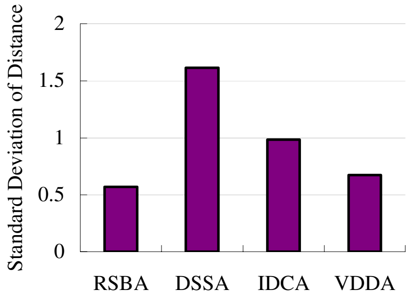

Fig. 8. Standard deviation of distance comparison

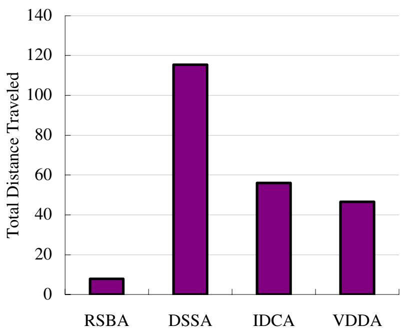

Fig. 9. Total distance traveled comparison

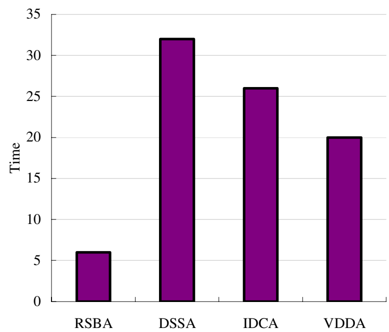

Fig. 10. Termination time comparison

the other 3 algorithms. Termination time is measured in the number of iterations until the algorithms stop.

5 Conclusion and Future Work

In this paper, we designed and evaluated our proposed movement assisted 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 which guarantees the network connectivity. After initially deploying a certain number of mobility limited sensors in the ROI, the sensors were clustered and the optimal CHs positions were chosen by PSO before movement. We determined an optimal movement plan by proposed RSBA algorithm for the sensors in order to maximize the network coverage and simultaneously minimize the total number of movements. Dijkstra's 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 from densely deployed areas to sparsely deployed areas. Based on simulation, we evaluated and compared our approach RSBA with other related works from various aspects: coverage, standard deviation of distance traveled, total moving distance, and deployment time, and show that RSBA is very effective in terms of the above standards.

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

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.

References

1. S. Meguerdichian , F. Koushanfar, G. Qu and M. Potkonjak: Exposure in Wireless Ad-Hoc Sensor Networks. Mobicom (2001)
2. S. Dhillon, K. Chakrabarty and S. Iyengar: Sensor placement for grid coverage under imprecise detections. Proc. International Conference on Information Fusion (2002)
3. T. Clouqueur, V. Phipatanasuphorn, P. Ramanathan and K. k. Saluja: 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 Conference, Hong Kong (2004)
8. Y. Zou and K. Chakrabarty: Sensor deployment and target localization based on virtual forces. Proc. IEEE INFOCOM Conference, Vol. 2 (2003) 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) 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. Proc. of ICESS' 05/ LNCS, Xi'an, China, (2005) 533-541
11. http://www.darpa.mil/ato/programs/shm/index.html
12. Sriram Chellappan, Xiaole Bai, Bin Ma and Dong Xuan: Mobility Limited Flip-based Sensor Networks Deployment. Dept of Computer Science and Eng, Ohio-State Univ. Technique report (2005)
13. Sriram Chellappan, Xiaole Bai, Bin Ma, and Dong Xuan: Sensor Networks Deployment Using Flip-based Sensors. Proc. of IEEE International Conference MASS'05 (2005)
14. Radu Stoleru, Tian He, John A. Stankovic, David Luebke: A High-Accuracy, Low-Cost Localization System for Wireless Sensor Networks. ACM conference Sensys (2005)
15. Wendi B. Heinzelman, Anantha P. Chandrakasan, and Hari Balakrishnan: An Application-Specific Protocol Architecture for Wireless Microsensor Networks. IEEE Transactions on Wireless Communications, Vol. 1, No. 4 (2002) 660 - 670
16. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein: Introduction to Algorithms. 2nd Edition. MIT Press and McGraw-Hill (2001) 595–601