TJER 2013, Vol. 10, No. 2, 33-45

Mobility Assisted Sensor Node Self-Deployment for Maximizing the Coverage of Wireless Sensor Networks using A Genetic Algorithm

VV Juli[*a] and J Raja[b]

[*a]*Department of Electronics and Communication Engineering, Anna University of Technology, Tamilnadu, India*
[b]*Department of Electronics and Communication Engineering, Anna University of Technology, Tiruchirapalli, Tamilnadu, India*

Received 14 February 2012; accepted 4 December 2012

Abstract: Abstract : Wireless sensor networks extend the capability to monitor and control far-flung environments. However, sensor nodes must be deployed appropriately to reach an adequate coverage level for the successful acquisition of data. Modern sensing devices are able to move from one place to another for different purposes and constitute the mobile sensor network. This mobile sensor capability could be used to enhance the coverage of the sensor network. Since mobile sensor nodes have limited capabilities and power constraints, the algorithms which drive the sensors to optimal locations should extend the coverage. It should also reduce the power needed to move the sensors efficiently. In this paper, a genetic algorithm- (GA) based sensor deployment scheme is proposed to maximize network coverage, and the performance was studied with the random deployment using a Matlab simulation.

Keywords: Coverage, Deployment, Network, Population size and sensor

بمساعدة التنقل لنقاط حساسات الانتشار الذاتي لتغطية شبكات الحساسات اللاسلكية باستخدام خواريزم جيني

فوليت جولي* راجا جانا كي رامن راجا

الملخص: شبكات الحساس اللاسلكية زادت من المراقبة والتحكم في البيئة البعيدة عن النظر. وكان لابد من توزيع نقاط الحساسات بشكل مناسب لتصل لمستوى تغطية كاف لتحصل على المعلومات بنجاح. أجهزة الحساسات الحديثة قادرة على التحرك من مكان لآخر وتشكل شبكة الحساس المتحرك. هذه القدرة النقلية للحساسات يمكن استخدامها لتعزيز شبكة تغطية الحساسات. بما أن نقاط الحساسات المتنقلة لها إمكانات محدودة و قدرة مقيدة فإن الخواريزم الذي يقود الحساسات للمكان الأمثل لابد أن يحتوي على نطاق واسع التغطية. أيضا لابد من تقليل القدرة المطلوبة لتحريك الحساسات بكفاءة. في هذه المقالة يتم تقديم مقترح لنظام انتشار الحساسات باستخدام الخواريزم الجيني لتكبير نظام التغطية ودراسة الأداء باستخدام نظام انتشار عشوائي عن طريق حزمة الماتلاب.

الكلمات المفتاحية: تغطية، الانتشار، الشبكات، حجم المجموعات أو السكان، الحساسات

*Corresponding author's e-mail: violetjuli.prince@gmail.com

1. Introduction

A wireless sensor network (WSN) is a type of network in which tiny sensor nodes are spread out into smaller groups around premeditated positions and connected by wireless media. The sensor nodes have the ability to sense the phenomenon, process the data, and communicate with other sensor nodes and are placed one by one, either manually or by robot.

The deployment of a large number of sensor nodes needs very attentive handling of topology maintenance. In military applications or when the environment was unknown the sensors are scattered randomly throughout the field, whether on land or water. The sensor nodes could be deployed statically or in a highly mobile environment, but generally the nodes are thrown out en masse from aircraft or ballistic propulsion systems to acquire satisfactory coverage. As a result, the density of distribution is not uniform using this method (Chen and Sun 2009). Of course, the method for initial deployment should reduce the installation cost, eliminate the need for pre-organization and pre-planning, enhance the flexibility of arrangement, and promote self-organization and fault tolerance (Akyildiz *et al.* 2002).

Failure of a sensing device is a general and standard occurrence due to energy depletion or destruction. Invariably, there are changes in topology as well as changes in the nodes' positions, reachability, malfunctioning, and task details that transpire after deployment. Those malfunctioning nodes could be replaced by redeploying additional nodes at any course of time, but the addition of new nodes requires a need to reorganize the network.

As a result of this, the sensor network topologies undergo frequent and continuous fluctuations after deployment as the deployment of nodes influences the efficiency of the network. A good deployment will improve resource allocation, communication, and information gathering. However, because of inevitable device failure, topology maintenance has turned into a challenging task (Chen and Sun 2009).

Coverage is considered one of the basic and fundamental difficulties in sensor networks. Getting and providing complete network coverage has been a challenging problem. In order to achieve complete coverage, it has been required to deploy a massive number of sensors in the field. Besides the execution of a wide set up, coverage formulations should endeavor to detect weak areas in the monitoring domain and suggest schemes for the future utilization and reconfiguration of sensors to improve the overall coverage of the network.

Today, sensing devices are freely and easily transported from one place to another, possibly by being attached to robots, vehicles, tanks, animals, and even human beings. Despite the fact that mobile devices are of high operational value, they also are associated with high power consumption and increased collisions in wireless transmissions. However, it is possible that mobile sensors can be used as energy sources to facilitate in energy transfers to stationery devices. In a mobile network, sensor nodes or the sink can be moved; in some networks, both can be kept mobile.

In this paper, we address the problem of moving the randomly dropped sensors from their initial locations to optimal locations using a genetic algorithm (GA). The main goal of the paper is to better the distribution pattern of the nodes by redeployment of the same from crowded areas to nearby uncovered areas, thereby maximizing coverage with minimal displacement of already active nodes.

This paper is organized as follows: The deployment objectives and existing deployment algorithms are presented in section 2. The problems associated with sensor node deployment and desired properties of algorithms is presented in section 3. The proposed algorithms and solutions are described in section 4. The performance evaluation of deployment strategies is narrated in section 5. Finally, the conclusion and future works are highlighted in section 6.

2. Deployment Algorithms

2.1 Deployment Intentions

Despite several deployment objectives that can exist when establishing sensor networks (eg. coverage, lifetime, routing, etc), maximizing coverage is an area of primary concern. Cardei and Wu (2004) identified three types of sensor coverage: area, point, and barrier. Gage (1992) sorted the same into barrier, blanket, and sweep coverage. Moreover, (Huang and Tseng 2003) mentioned that coverage was considered a decision problem which identified whether every point in the region was covered by at least some minimum number of sensors. In their study, an algorithm determined the level of coverage in each location, and application areas of the algorithm were also discussed. The same concept was examined by (Zhou and Gupta 2004). In their study, a minimum size connected k-coverage problem was addressed. The study found that the problem resulted in fault tolerance and energy conservation techniques because only a few selected sensors were made active at a particular time.

Rabie *et al.* (2006) addressed the coverage question of how to monitor the chiefly imperative areas in the meadows to which the sensors had been deployed by deploying extremely reliable sensors. Maximizing the sensor network life was the second most important objective in that study of sensor deployment.

Zhao *et al*. (2007) proposed an algorithm in which integer planning was adopted to deploy the relay sensors in static heterogeneous networks. Greater energy efficiency was achieved by reducing the path length. The sensors' batteries as well as the quantity of data sent per sensor played a major role in increasing the life of the sensor network. Reducing the power needed for routing and communication was another key goal of deployment. As a result, sensor communication range played the most important role in maintaining quality network connectivity.

In our study, deployment algorithms were categorized as random, incremental, or movement assisted according to the method of sensor deployment.

2.2 Random Deployment

Random deployment was the method used to place the sensor nodes when the environment was unknown or subject to a severe change in conditions. Sensors could be "scattered" randomly from a helicopter or thrown using flying robots.

Clouqueur *et al*. (2002) used a random deployment method for target detection where the sensors were deployed sequentially. A restricted number of sensors were deployed in each step until the required revealing probability was achieved. A mechanism for sensor collaboration to perform target detection was proposed, with the goal of maximizing exposure of the least exposed path in the region. The cost function used in that study depended on the number of sensors deployed in each step and the cost of each deployment. The cost function was minimized by appropriately choosing the sensors. They found that the optimal number of sensors deployed in each step varied with the relative cost assigned to deployment and sensors.

Morteza and Massoud (2005) considered the problem of energy efficient random deployment of sensor nodes. The objective of the deployment was to find the node density at every point inside the specified deployment region, subject to restriction of the quality of monitoring (QoM) and network lifetime. Using random methods, the cost of and time needed for deployment were reduced. However, the actual landing positions of the sensors were different from the intented ones due to the presence of wind, moving objects, and other obstacles present in the environment. Their finding was that deploying the sensor nodes randomly might not produce a uniform distribution of nodes and could possibly result in an inefficient WSN. As a result, thorough coverage was not guaranteed and possibly would not have been able to meet the application requirements.

2.3 Incremental or Centralized Deployment Algorithms

In the incremental deployment, the nodes were launched one by one into the target field, with a powerful centralized sink node responsible for node deployment. Each node maintained bidirectional communication with the sink node. Information about the previously deployed nodes was communicated by the sink node to the sensor nodes and that information was used by the nodes to determine their ideal locations.

Howard *et al*. (2002) proposed an incremental deployment algorithm in which the sensor nodes were added one at a time. They discussed four phases of the algorithm; namely, initialization, selection, assignment, and distribution. They also considered the obstacles present in the target area while calculating the coverage probability. The network coverage was maximized with the constraint that nodes should maintain line of sight with their neighboring nodes during the deployment process.

Wu *et al*. (2006) proposed Delaunay triangulation based on an algorithm called the delaunay triangulation (DT-) score. The goal was to maximize the coverage area of an obstacle-ridden sensor field. In that method, the sensors were added incrementally. The coverage increased when the number of deployed sensor nodes increased. For all random scenarios, the deployment method increased coverage.

In the incremental deployments, the nodes were positioned in optimal locations during each step of the deployment process. Hence, when the nodes were moved to improve coverage, only a minimum amount of energy was consumed for sensor mobility. However, the deployment time increased. Scalability was one of the main problems and, since a large number of sensors was used in the networks, numerous messages were simultaneously reported to the centralized node. Generally, in highly dynamic networks, the cost and power consumption of the network increases due to the large number of messages being communicated. Hence, such algorithms were prone to single point failure. In such situations, the mobility assisted distributed deployment algorithms would provide better solutions for solving the scalability problem.

2.4 Mobility Assisted Deployment Algorithms

In previous studies, random and incremental deployment strategies resulted in poor network performance and reduced network lifespan. Hence, in this study, the mobile sensor nodes were deployed in the field to improve coverage and other network performances such as energy consumption and network lifetime. The sensors were capable of self-deploying after the initial random deployment and arranging themselves in the monitored field based upon a local decision. Here the powerful sink node was not involved and the nodes made their own decisions in determining their optimal locations. There were mainly two approaches: a computational geometry-based approach and a virtual force-based approach.

2.4.1 Computational Geometry-Based Approach

In the geometrical approach, the sensor regions were represented by a set grid or i-polygon. The grids and polygons changed when the sensor nodes were moved. Luo *et al.* (2005) proposed a novel method called the Grid Method. The deployment area was divided into individual grids. The locations of the pre-deployed nodes and obstacles were already known and marked by the base station. The total weighted value of each grid was calculated through the consideration of factors such as boundary, pre-deployed nodes, hot zones, and obstacles. The section of the grid with the lowest weighting score was deployed with a new mobile node. Coverage and uniformity improved rapidly when mobile nodes were placed in the target field without altering the distribution of previously deployed nodes. The calculations involved were simple. Wang *et al.* (2006) proposed two sets of distributed protocols for controlling the movement of sensors to achieve target coverage: basic and virtual movement protocols.

In the basic movement protocols, sensors detected coverage holes using a Voronoi diagram. To heal the holes, the nodes were moved. In the virtual movement protocols, sensors did not perform iterative physical movements. Instead, after calculating the target locations, sensors moved virtually and exchanged those new virtual locations with the sensors which would have been their neighbors if they had actually moved. The actual movement occurred only when the communication cost to reach their logical neighbors was too high, or when they determine their final destinations. Those protocols were effective in terms of coverage, deployment time, and movement.

In computational geometry methods, the nodes were uniformly distributed in the sensor field, but the presence of obstacles was not taken into consideration and their geometrical areas were not modeled. In each iteration, since each node was moved after calculating the optimal location, the energy consumption for driving the sensor was high.

2.4.2 Virtual Force Based Approach

Virtual force-based techniques have been used to solve navigation problems in mobile robotics. In these methods, the nodes and obstacles were considered points with attractive and repulsive forces. The resultant force depended on the distance between the neighboring nodes and obstacles. The nodes move from denser areas to the uncovered area similar to the way in which a charged particle would move in an electrostatic field.

Howard *et al.* (2002) considered the problem of deploying a mobile sensor network in an environment that may be both hostile and dynamic. They presented a potential field-based approach to deployment in which nodes would be treated as virtual particles subject to virtual forces. These forces repelled the nodes from each other and from the obstacles. Hence, the nodes were forced to spread throughout the target field, and the approach was both distributed and scalable. The nodes were quickly spread out to maximize the coverage area of the network. In addition to these repulsive forces, the nodes were subjected to a viscous friction force. This force was used to ensure that the network would eventually reach a state of static equilibrium where all nodes would ultimately come to a complete stop. If something was moved, the network would automatically reconfigure itself for the modified environment before returning once again to a static equilibrium. Thus, nodes moved only when it was necessary to do something and saved considerable energy. The algorithm was both robust and highly scalable.

Sameera and Gaurav (2004) studied a sensor deployment strategy that maximized the area coverage of the network with the constraint that each of the nodes had at least K neighbors, where K was a user-specified parameter. They proposed an algorithm based on artificial potential fields which were distributed and scalable. The algorithm did not require a prior map of the environment. It included two different kinds of forces. One was a repulsive force that tried to maximize coverage while the other was an attractive force that imposed a constraint of K degrees. As a result of these forces, a group of nodes placed close together spread out into a network to maximize the coverage while satisfying the constraint of K degrees. The repulsive forces from the obstacles were considered to avoid obstacles during the movement of the nodes. The algorithm was effective in terms of energy and coverage in the presence of obstacles.

In virtual force-based methods, the presence of obstacles were also identified and considered during node movement. The degree of coverage could be controlled by the setting threshold on the distance between sensor nodes. But in those methods, a powerful cluster head node was responsible for collecting information and organizing node movement. In many environments, such as disaster areas, the cluster head node might not be present. In such situations, organizing nodes into clusters became difficult and resulted in network partitions.

2.4.3 Other Approaches for Mobility

Wu *et al.* (2007a) suggested three sensor relocation algorithms to match the mobility degree of sensor nodes. The particle swarm optimization algorithm (PSOA), relay shift based algorithm (RSBA), and energy- efficient fuzzy optimization algorithm (EFOA) were proposed to relocate the sensors. In PSOA, the mobility of the sensors was not limited. But in RSBA and EFOA, the sensors' mobility was limited by setting a threshold in mobility so that the energy consumption was reduced. The algorithms improved the coverage.

Wu *et al*. (2007b) presented a method for the redeployment of sensor nodes in a hybrid network which consisted of both static and mobile sensor nodes. The sensors were moved without altering the position of static sensors. The direction of node movement was based on the Analytical Hierarchy Process (AHP). Four factors, namely coverage hole, hot spot, obstacle, and the node boundary were considered for optimal deployment. When the number of mobile nodes was increased, the coverage also increased since the field became more flexible due to the movement of the sensors. The coverage improved with limited node movement without compromising connectivity.

Li and Lei (2009) presented the improved particle swarm optimization (PSO) strategy for optimizing the position of mobile nodes and increasing the whole coverage area. After initial random deployment, the nodes were able to communicate with the cluster head node. The cluster head executed the PSO algorithm and managed a one-time movement of sensor nodes to the desired location. The coverage of the sensor was represented by a number of intersection points on the established grid. The coverage ratio was used as the fitness function. The connectivity between nodes was ensured. The algorithm provided better coverage performance than the VFA, and a reduced network.

In our work, we aspired to better the distribution pattern of the sensor nodes in the target field. The nodes were moved from the crowded areas to the uncovered areas using a GA so that the coverage was improved.

3. Problem Specification

The deployment of the sensors could not be carried out effectively by humans in various unfavorable working environments, such as remote harsh fields, disaster areas, and toxic urban regions. To accomplish that task, it was suggested to scatter the nodes by aircraft. But while implementing that technique, many obstacles and hindrances such as the presence of wind, trees, and buildings were encountered, and the actual landing positions of the sensors were altered. That improper landing of devices led to poor network coverage. The density of the nodes was not uniform throughout the sensor field. To overcome those difficulties, it was imperative to consider the mobile sensors for the usage because of their capability of moving to the right place for providing the required coverage. While congregating the algorithms, care must be taken not to jar the nodes while moving them. Such jarring creates a dead zone. An ultimate solution could be obtained by putting a maximum limit on the algorithm iterations and runtimes.

Gathering information about the neighboring nodes was the biggest hurdle to overcome. In such work, a sophisticated global positioning system- (GPS) based network scenario was assumed. The following were some of the desired properties of the proposed solution:

* **Convergence and Connectivity**: The algorithm should converge and provide an ultimate solution for the specified objective, and the whole network should be connected.
* **Coverage**: The final layout of the network should endow the area with the best coverage of the agreed target area with the fewest "dead zones".
* **Minimalism**: The algorithm should not pave the way for the nodes to spend excessive time finding their environment and calculating a solution since this causes enormous drain on node energy.
* **Scalability**: The algorithm protocol should operate with an arbitrary number of nodes and should properly adjust to work when the nodes are added or removed from the network.
* **Competence**: The modus operandi ought to evade loops where nodes incessantly move in a cycle and constantly move between the same positions.

In this study, the problem of placing mobile sensors in the premeditated positions to maximize the coverage of the network was explored. The GA-based protocol was proposed for controlling the movement of sensors to improve network coverage. In that protocol, sensors moved iteratively and eventually reached the final destination. In the GA-based method, in each iteration (generation), the fitness of each set of locations of the population was calculated to select the two best solutions (genes) based on the coverage. Those two selected genes would undergo crossover and mutation operations to create a new population with a new set of potential solutions. The procedure was repeated for a number (n) of generations or until acquiring an optimum solution (*ie*. coverage). In this study, a GA-based sensor deployment scheme was implemented and the results were compared with the random deployment scheme using a Matlab (MathWorks, Inc. Nattick, Massachusetts) simulation.

4. GA-Based Sensor Node Deployment

In this study, an autonomous deployment scheme using GAs was described detailing a method to move the sensors from the regions of dense deployment to the areas of sparse deployment requiring enhanced coverage.

4.1 Genetic Algorithm (GA)

GA is used in computing near perfect solutions for optimization. Techniques stimulated by evolutionary biology such as inheritance, mutation, selection, and crossover (also called recombination) are used by GAs. These types of algorithms are applied as a computer simulation in which a population of conceptual representations (*ie*. chromosomes, the genotype, or the genome) of candidate solutions (*ie*. individuals, creatures, or phenotypes) are compared to an optimization problem in a movement toward developing better solutions. Binary values as strings of 0s and 1s are usually adopted to represent the solutions but there are also possibilities for other encodings. A population of randomly generated individuals is the initial point for the progression to start and progress in generations. The robustness of every individual in the population is assessed. Manifold individuals are stochastically selected from the current population purely based on their fitness, and then are customized to form a new population in each generation, and they are used in the subsequent iteration of the algorithm. At that point, when a suitable fitness level is obtained for the population or when a maximum number of generations have been produced, the algorithm is terminated.

4.2 The Proposed GA-Based Sensor Node Deployment

Figure 1 shows the design of the GA-based node deployment algorithm. The two-dimensional sensor field is assumed and hence the z-direction is assumed as zero. In the sensor field, 100 nodes are randomly deployed. The step-by-step procedure is discussed below:

Step 1: Initial Population

To change the location of a node, one has to specify the displacement as well as the directions of the nodes. So, for coding the displacement and direction, four numbers will be needed-two for the x-direction and two for the y-direction. The four attributes (t_x, t_y, x_d, y_d) are coded with binary notation of bit string, where tx and ty are the small displacements in the x- and y-directions, x_d and y_d represent the x- and y-directions. Since there are n nodes and n sets of displacements, directions are used to handle the movement of all the nodes in the network. If the population size, or future locations for each node, is N, then there should be N x n x 4 numerical values to represent the N number of n sets of node movements. If we use 4 bits for representing the maximum possible movements (15), then two bits will be used to represent the x- and y-directions.

Later, during calculating the fitness of a solution, if the direction bit is 0 then the small distance will be added to the corresponding coordinate of the location of the node. If the direction bit is 1, then the small distance will be subtracted from the corresponding coordinate of the location of the node.

As far as the problem is concerned, if we need to optimize the location of only one node then there will not be a change in direction of the other nodes. But in this case, we will be changing the location of 100 nodes. Hence, there are many chances for a change in direction of the sensor nodes during each generation because of the change in location of all the neighbors. Further, a node cannot go and fill a distant coverage hole. It can be allowed to move only a short distance and fill the coverage hole to minimize the power needed for moving the sensors. That is why the node is limited to 15 movements.

Step 2: Evaluation

After creating random locations, the first step is to calculate the fitness value of each location in the population based on the similarity of the previous optimum locations. The process of evaluating the fitness of a location consists of following three steps:

* Conversion of the chromosome's genotype to its phenotype. That means converting the binary string into corresponding real values.

* Evaluation of the objective function. The objective function is the sensor network coverage and one has to maximize the coverage in each generation.

* Conversion of the value of objective function into fitness value.

Step 3: Objective Function Values and Fitness

The set of the nodes' new locations are calculated using the current location of the nodes, displacements, and directions given by the GA. The objective function (coverage area) values of F (fitness values) are calculated using the new location of the nodes as follows:

Let,

n is the number of nodes,

A_i is the coverage of the single node i = πr^2

r is the sensing range.

V_j is the total covered region, $V_j = \bigcup_{i=1}^{i=n} A_i$

For j=1 to N, evaluate Vj N is the population size (*ie.* assumed future locations for each node).

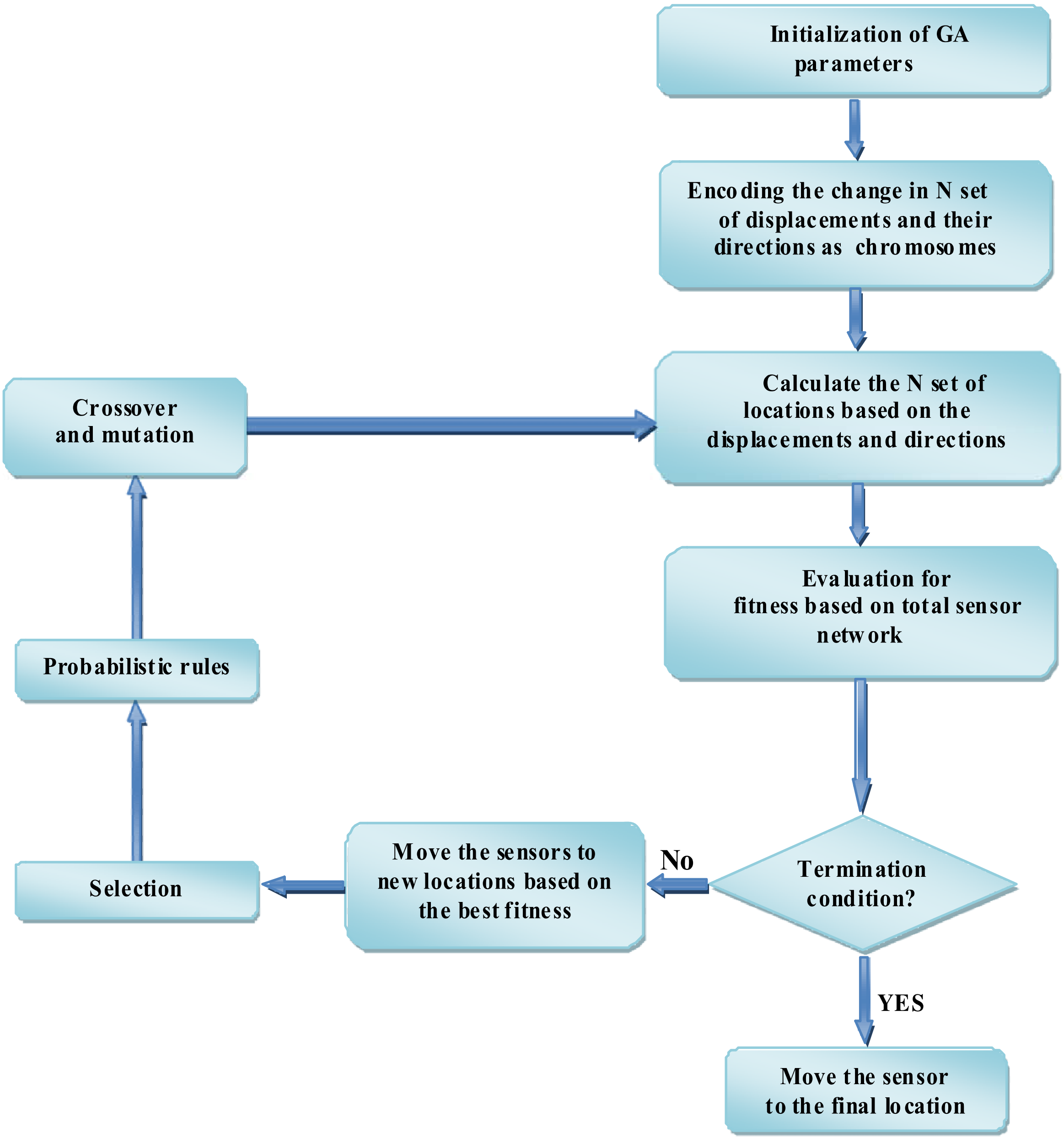
Initialization of GA parameters
Encoding the change in N set of displacements and their directions as chromosomes
Calculate the N set of locations based on the displacements and directions
Evaluation for fitness based on total sensor network
Termination condition?
No
Move the sensors to new locations based on the best fitness
Selection
Probabilistic rules
Crossover and mutation
YES
Move the sensor to the final location

Figure 1. GA model

Evaluation is the process which compares the total coverage achieved by n number of nodes when each node is located in any one of the assumed future locations N.

If there is n number of nodes, there will be n corresponding locations. For each node, we consider N possible future locations. Out of N, we consider the first best two locations, which contribute to maximizing the coverage achieved by n number of nodes in the target field. The best set of new locations as well as their previous displacements and directions would be preserved for further processing.

Step 4: Creation of a New Population

After evaluation, a new population from the current generation is created using the previous set of displacements and directions of the best two solutions to generate the new population. The three operations, reproduction, crossover, and mutation, are used. The crossover and mutation rates should be selected in accordance with the convergence factor. Otherwise, the algorithm results in a primitive random search. The population size is fixed with respect to the convergence factors. For that, the previously selected best displacements and directions are also included in the new population.

Reproduction: For each node, from the newly found locations in the population of a generation, the locations with the best fitness and the second best fitness are allowed to exist and are selected as new parents to produce the next generation. The best set of new locations as well as their previous displacements and directions would be preserved for further processing.

Crossover: The crossover used here is the one-cut-point method, which randomly selects one cut-point and exchanges the right parts of two parents to generate offspring. The crossover point would be chosen selectively with respect to the convergence factors.

Mutation: After crossover, the mutation is carried out. In accordance with the convergence factors, the mutation level is selected. One or more genes (direction bits) are modified by mutation with a probability equivalent to the mutation rate, and a sequence of random numbers *rk* is generated. (The number of bits in the whole population) If r_i is 1, change the ith bit in the whole population from 1 to 0 or from 0 to 1. The chromosomes (directions) reproduced are not subject to mutation, so after mutation, they should be restored. After the finishing point of a single iteration of the GA, a new population is created. Then the entire process is repeated for as many times (*eg*. ten) as preferred until the best value of the objective function in each generation is evaluated. Ultimately, the two best sets of optimum sensor locations based on the best fitness are reached.

5. Results and Discussions

The random and proposed GA-based sensor network deployment scenarios were simulated using MATLAB and the performance of deployment algorithms were analyzed in terms of displacement of nodes, cumulative displacement and coverage of network. The following network is considered:

Sensor network size	:	600 m x 600 m
Total number of nodes	:	100
Sensing range of the sensors	:	50 m

Initially the sensors were randomly dropped in a 600 meter x 600 meter field. Figure 2 shows the randomly dropped sensor in a field. The blue color points represent the randomly dropped sensors in the field. The green circles represent the sensing range of the sensors. One can see that the nodes are densely deployed in some areas and sparsely deployed in some other areas. Hence the nodes are not uniformly distributed. In Figure 3, the rectangles represent the areas where nodes are densely deployed and the ellipses represent the areas which are uncovered by the sensor nodes. In Figure 4, the green area represents the covered area by the randomly dropped sensors and the

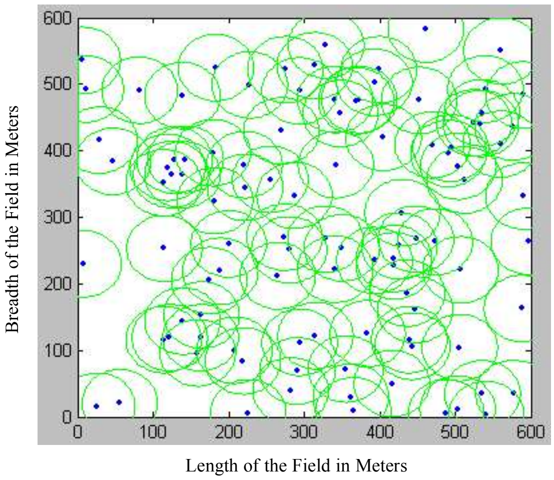

Figure 2. Randomly dropped sensors

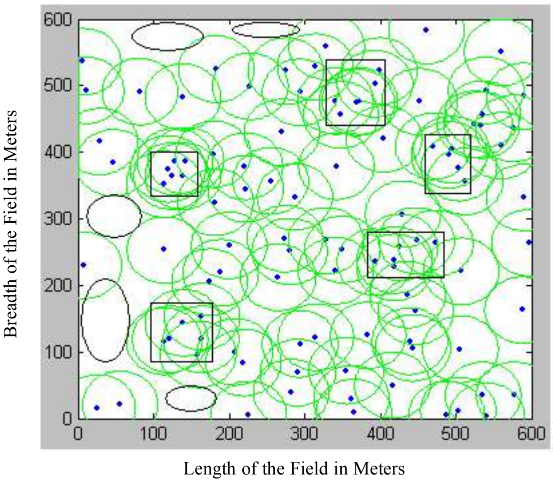

Figure 3. Field with densely and sparsely deployed sorsors

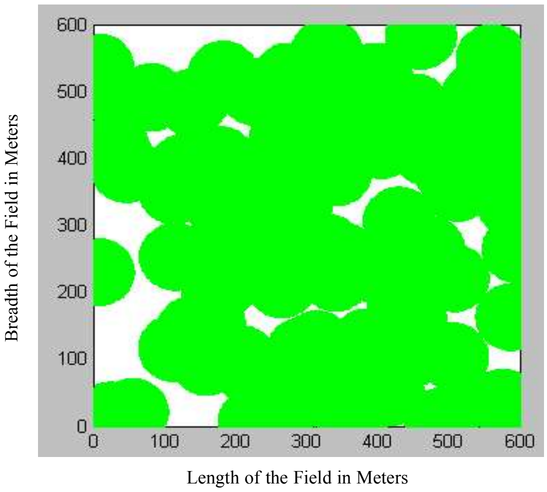

Figure 4. Covered and uncovered regions of sensor field

Table 1. Results of GA based sensor deployment

Iteration	Displacement in Meters	Cumulative Displacement in Meters	Coverage in Percentage	Coverage Improvement index
1	2640	2640	86.91	1.0279
2	2243	4883	88.71	1.0207
3	1858	6741	90.80	1.0235
4	1606	8347	92.17	1.015
5	1590	9937	93.76	1.0172
6	1520	11457	95.21	1.0154
7	1566	13023	95.43	1.0023
8	1501	14524	96.17	1.0077
9	1545	16069	96.57	1.0041
10	1524	17593	96.45	0.9987

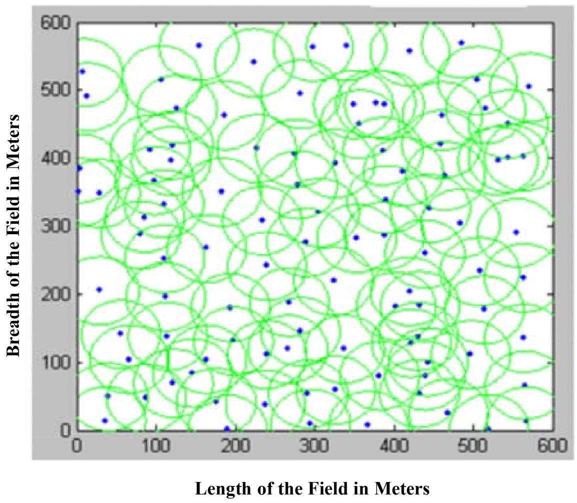

Figure 5. Sensor field after optimization

white area represents the uncovered areas.

During each iteration of GA, the system was programed to discover better locations. To achieve the results in Table 1, the following GA parameters were used:

Total Population Size	:	200
Total Number of Generations	:	10
Mutation Level	:	0.20
Crossover Rate	:	0.20

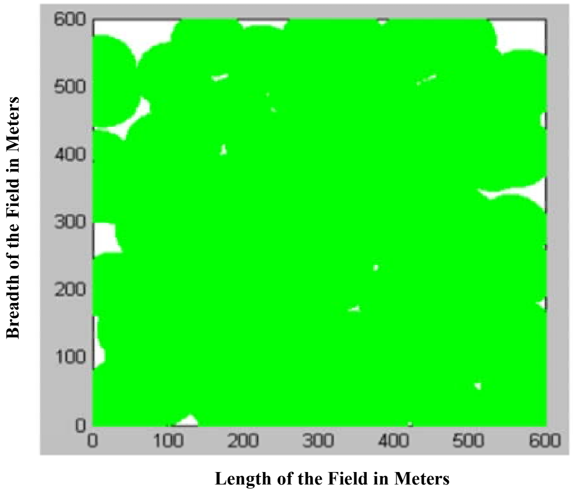

Figure 6. Covered and uncovered regions after optimization

Figure 5 shows the sensor field after the tenth iteration of the GA-based sensor node deployment method. One can clearly see that the nodes had moved from the denser areas to the uncovered areas. The covered and uncovered regions after optimization were shown in Fig. 6.

Figure 7 shows the variations in node displacement during each iteration of the algorithm. The displacement of nodes decreased when the iteration was increased. In the initial little iteration, the displacement was increased. Figure 8 represents the cumula-

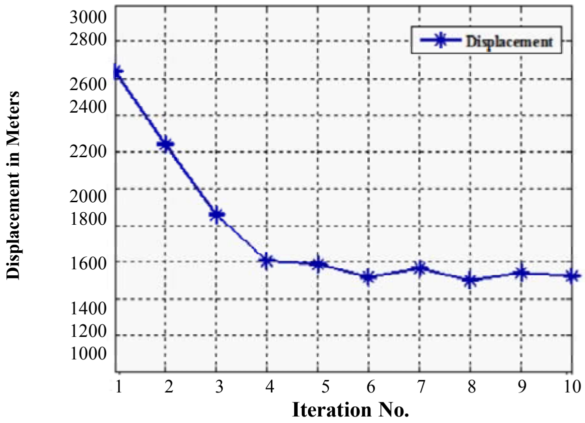

Figure 7. Iteration vs. displacement of nodes

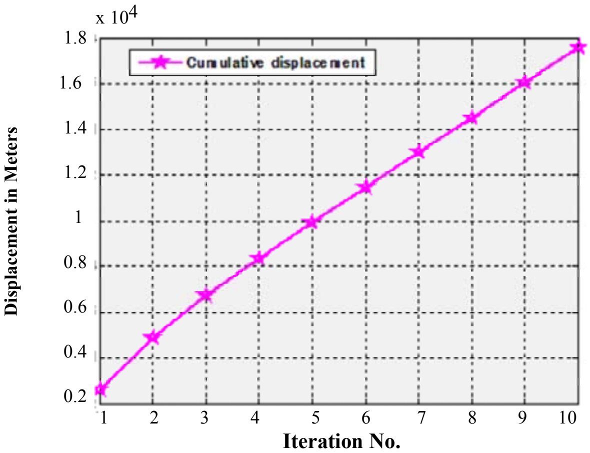

Figure 8. Iteration vs. cumulative displacement of nodes

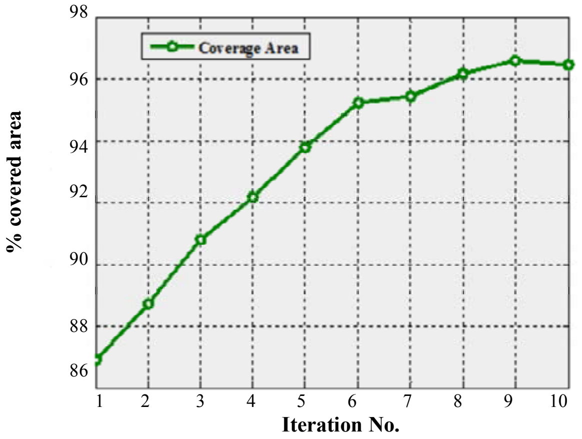

Figure 9. Iteration vs. percentage of covered area

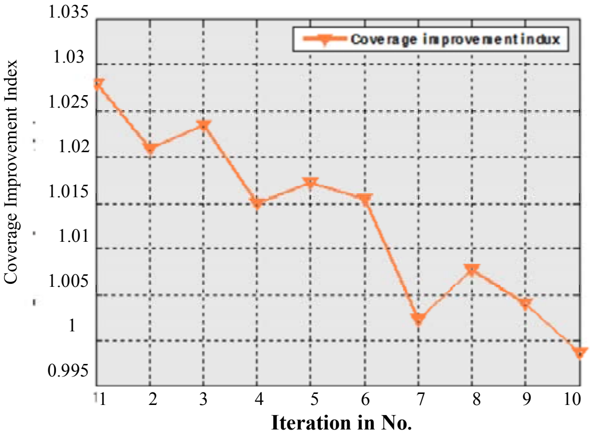

Figure 10. Iteration vs. coverage improvement index

tive total displacement of sensors in meters.

After each iteration, we calculated the area covered by the 100 sensor nodes. The coverage was described by the ratio of the union of covered area of each node in the area of interest. If the node was located well inside the target field, the complete area of that particular node was included in computation of coverage. If the node was located near the boundary regions of the field, then only the partial area covered by the node was included in computation. Figure 9 shows the variation of the percentage of area covered by the sensors with respect to the iteration. Until the seventh iteration there was significant improvement in coverage.

We analyzed the improvement in coverage of sensor network achieved iteration by iteration. The basis for the first iteration was the coverage achieved in random deployment. The basis for the second iteration was the results of first iteration *ie*. coverage achieved in first iteration. Similarly the basis for the third iteration was the results of second iteration. Thus all the iterations carried out progressively with the results of previous iteration for the subsequent iteration. For each iteration, we calculated the coverage improvement index using

$$I_n = A_n + 1 \ / A_n$$

where I_n is the coverage improvement index for the nth iteration and A_n is the coverage area achieved in the nth iteration.

(n = 1, 2, 3 …10 as the iteration number).

The coverage performance improvement index is tabulated in Table 1 and the variation in improvement index is graphically represented in Fig.10. In iteration 1, the index was initially maximized and slowly reduced when the iteration was increased. It means that the algorithm gives optimal results in the initial few iterations itself. The distribution of nodes was better; thus, the coverage improved.

Figure 11 represents the comparison of maximum coverage achieved by the random and GA- based sensor node deployments, respectively. The coverage

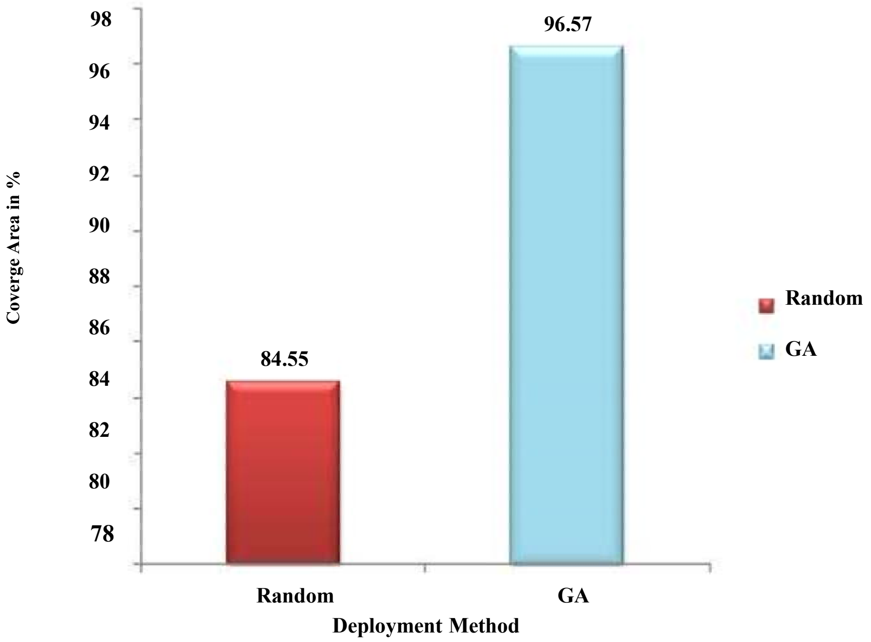

Figure 11. Coverage area

Table 3. Results for various random topologies

Covered Area in %		
Topology	Random deployment	GA based deployment
1	87.64	96.26
2	85.84	94.92
3	89.04	96.04
4	81.37	91.59
5	90.84	96.63
6	86.75	96.23
7	86.19	96.19
8	82.75	93.15
9	87.11	96.07
10	89.26	96.53

improved from 84.55% to 96.57% through GA-based node deployment.

We obtained the results from a single run but confirmed the results by running the simulation many times. We also used the algorithm to various random topologies. In all topologies, we deployed 100 nodes randomly, applied the GA, and analyzed the coverage. The results are tabulated in Table 3. The comparison of coverage achieved by the random and GA-based deployments after 10 iterations for 10 random topologies is shown in Fig. 12. The GA-based algorithm gave better results for all topologies studied. Thus, the topology could be selected as per the coverage requirement.

6. Conclusions

The study of GA-based sensor node self-deployment by using the MATLAB simulation technique improved the effectiveness of the coverage of the sensor network in comparison with that of random deployment. The conclusions of the research study were summed up as follows:

1. The coverage improved through GA-based deployment.
2. The redeployment of the sensors was accomplished using nodes from the nearby crowded areas only; hence, the power consumption for sensor movement was less than what it would be if moving them from far off places.
3. This energy saving would, in turn, prolong the life of the battery and consequently enhance the life of the network.
4. The combination of points 1 through 3 would result in greater cost effectiveness for the entire operation of sensor deployment and data collection.

As an extension of this study, hybrid sensor deployment algorithms could be incorporated for furthering the already achieved benefits of the application of GA for the deployment of sensors.

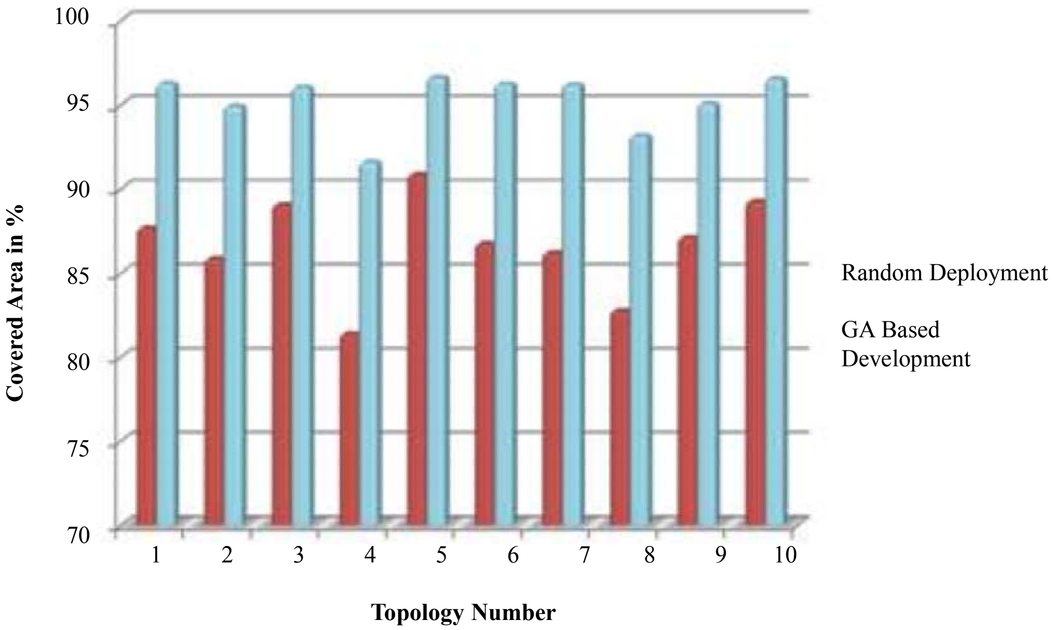

Figure 12. Comparison of coverage for different topologies

References

Akyildiz IF, Su WL, Sankarasubramaniam Y, Cayirci (2002), A survey on wireless sensor networks. IEEE Communication Magazine 40(8):102-114.

Cardei M, Wu J (2004), Coverage in wireless sensor networks. In Handbook of Sensor Networks, CRC Press, 0-8493:1968-4.

Chen JM, Sun YX (2009), The deployment algorithm in wireless sensor network: A survey. Information Technology Journal 8:293-301.

Clouqueur T, Phipatanasuphorn V, Ramanathan P, Saluja K (2002), Sensor deployment strategy for target detection. Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications 42-48.

Gage DW (1992), Command control for many robot systems. Proceedings of the 19th Annual Technical Symposium AUVS. Pages 22-24. Reprinted in Unmanned Systems Magazine 10:28-34.

Howard A, Mataric MJ, Sukhatme GV (2002), An incremental self-deployment algorithm for mobile sensor networks. Autonomous Robots Special issue on Intelligent Embedded Systems 13:113-126.

Huang CF, Tseng YC (2003), The coverage problem in a wireless sensor network. Mobile Networks and Applications 10:519-528.

Li ZM, Lei L (2009), Sensor node deployment in wireless sensor networks based on improved particle swarm optimization. Proceedings of the International Conference on Applied Superconductivity and Electromagnetic Devices 215-217.

Luo RC, Tu LC, Chen O (2005), Auto-deployment of mobile nodes in wireless sensor networks using grid method. IEEE 359-364.

Mortesa M, Massoud P (2005), QoM and lifetime-constrained random deployment of sensor networks for minimum energy consumption. IEEE 293-300.

Rabie R, Al-Nawaiseh A, El-Rewini H, Khaled A (2006), Parallel meta-heuristic approaches for deployment of heterogonous sensing devices. PDCS, accepted at the ISCA 19th International Conference on Parallel and Distributed Computing Systems.

Sameera P, Gaurav SS (2004), Constrained coverage for mobile sensor networks. Proceedings of the IEEE International Conference on Robotics and Automation 165-172.

Wang GL, Cao GH, La Porta TF (2006), Movement-assisted sensor deployment. IEEE Transactions On Mobile Computing 5:640-652.

Wu CH, Chung L, Chung YC (2006), A Delaunay triangulation based for wireless sensor networks.

Wu XL, Cho JS, d'Auriol BJ, Lee SY, Hee YY (2007a), Self-deployment of mobile nodes in hybrid sensor networks by AHP. Lecture Notes in Computer Science 4611:663-673.

Wu XL, Cho JS, d'Auriol BJ, Lee SY (2007b), Mobility assisted relocation for self-deployment in wireless sensor networks. The institute of Electronics Information and Communication Engineers (IEICE), Transaction on Communication E90-B(8):2056-2069.

Zhao C, Yu Z, Chen P (2007), Optimal deployment of nodes based on genetic algorithm in heteroge neous sensor networks. IEEE 2745:4.

Zhou ZH, Das Gupta H (2004), Connected K-coverage problem in sensor networks. Proceedings of the 13th International Conference on Computer Communication and Networks (IC3N).