

International Journal of Control Theory and Applications

ISSN : 0974-5572

© International Science Press

Volume 10 • Number 18 • 2017

Dynamic Positioning of mobile sensors using Modified Artificial Bee Colony Algorithm in a Wireless Sensor Networks

Rajesh K. Yadav¹, Daya Gupta¹ and D.K. Lobiyal²

¹ Computer Science & Engineering Department, Delhi Technological University, New Delhi, India, Emails: rkyadav@dce.edu, dggupta@dce.ac.in

² School of Computer & System Sciences, Jawaharlal Nehru University, New Delhi, India, Email: lobiyal@gmail.com

Abstract: The advancements in communication and transmission technology has led to the heightened usage of Wireless Sensor Networks in plethora of applications. As the utility of WSNs is being realized, in hindsight the various challenges related to sensor networks are coming to the fore, one such issue is the dynamic deployment of the sensor nodes in a monitoring area. This paper proposes a modified artificial bee colony algorithm (M-ABC) that is used to deploy the mobile sensors with the aim of increasing the coverage area of the network in turn improving the performance of the network. The performance of the proposed algorithm is compared with that of the artificial bee colony algorithm when used for deploying a wireless sensor network. The outcome of the comparison demonstrates that the proposed algorithm outperforms the artificial bee colony algorithm and the simulation results establish the validity and the efficiency of proposed approach.

Keywords: Wireless Sensor Network, dynamic positioning, artificial bee colony, probabilistic sensing model and modified local search.

1. INTRODUCTION

With considerable technical advancement being achieved in the field of electromagnetic theory and communication theory, wireless sensor networks (WSNs) are being increasingly used in diverse applications in research, business, and surveillance domains. The WSNs can be made up of homogenous sensors or heterogeneous sensors. WSNs constantly monitor the mission area detecting events of interest and processing the gathered information which is then relayed up to the base station for further analysis. While deploying WSNs special consideration is given to the problem of coverage and energy usage, optimization of these problems are crucial for the lifetime of the network.

The positioning of sensors of a WSN directly influences resource management.

A crucial issue faced in sensor resource management is that of determining the precise positions of the sensors in the mission area that reduces communication cost, overheads and has a high area coverage, as a result tackling the problem of dynamic deployment of WSNs becomes increasingly important.

Researchers have extensively studied the problem and have proposed different approaches of tackling the problem but all these problems fall into two broad categories namely static and dynamic deployment. In static deployment, the position of a sensor once fixed, it can't be altered throughout the lifetime of the network. On the other hand dynamic deployment approaches are full-bodied in comparison to their static equivalent. In dynamic deployment the sensors are mobile and hence have the ability to alter their positions if need arises like in the case of a node failure, in which case the topology of the network needs to be altered to compensate for the loss. The precise location where a sensor should be positioned can be determined either randomly or deterministically, but as it has been shown that the problem of dynamic node deployment is NP Complete hence deterministic approaches are less preferred.

2. RELATED WORK

(Bhondekar and Vij, 2009) proposed a node deployment approach using the genetic algorithm. The authors used a multi-objective fitness function whose parameters included field coverage, sensor overlap error, network energy etc. Every deployment was coded as bit string sequences and each was evaluated using the fitness function.

(Zou and Chakrabarty, 2004) suggested a new approach using a probabilistic localization algorithm along with virtual forces (VF) for maximizing coverage area. The VF algorithm uses a force directed approach for moving the sensors to improve the coverage. The VF algorithm had the advantage of minimal computational overhead and one time sensor repositioning.

(Wu et al., 2006) used a metric called DT- score on the basis of which a deployment sequence was generated. DT- score aims at maximizing area coverage in a static environment with obstacles. The initial deployment of sensors was done using a contour based method to minimize the number of holes which were later filled using Delaunay triangulation.

(Kukunuru et al., 2010) proposed a new approach that used particle swarm optimization (PSO) for maximizing coverage area at the same time decreasing the distance between the sensors.

(Wang et al., 2007) put forward a new approach for energy efficient coverage in WSN using distributed PSO and Simulated Annealing (SA). The fitness of a solution was gauged on the parameters of coverage and energy consumption. For reducing the energy intake, the authors proposed the use of a hybrid algorithm comprising of PSO and SA. The local best and the global best solutions of the PSO are calibrated and corrected using SA which is performed on the nodes of the sensor network.

(Wang et al., 2012) suggested a dynamic deployment strategy for coverage control in sensor networks using Biogeography Based Optimization (BBO) meta-heuristic. In BBO, the initial solutions are called 'habitats' and the fitness function is called the Habitat Suitability Index (HIS). New solutions are generated by using two operations called the migration operation and the mutation operation.

(Banimelhem et al., 2013) proposed a GA based algorithm for reducing the number of holes left after the random deployment of static sensor nodes. They suggested an algorithm that not only guides the mobile nodes to cover the holes by calculating their best position, but also determines the minimum number of mobile sensors required to achieve the objective.

(Chen et al., 2014) proposed a memetic based multi-objective optimization of the coverage problem in sensor networks. In their work, the authors propose the use of multiple local searches to find better deployment sequences that had high area coverage, efficient node utilization, and increase network lifetime.

3. Coverage calculation strategy and fitness calculation model

3.1 Grid based coverage calculation strategy

In our work, we have employed the grid based coverage calculation strategy wherein instead of covering the entire area we try to cover as many grid points as possible. The fitness of a deployment is measured in terms of

the number of such grid points covered. In this approach, the entire area was divided into square grids of equal size and the four endpoints of each of these grids were considered as grid points. The coverage is measured as the total number of such grid points covered by all the sensors divided by the total number of grid points in the mission area i.e.

$$\text{Coverage} = \frac{\sum_{j=1}^n C_j}{\text{Total number of grid points}} \times 100\% \quad (1)$$

Where C_j = number of grid points covered by sensor j .

3.2. Sensor Detection Model

There are two commonly used detection models for calculating the area coverage in wireless sensor networks namely the binary model and the probabilistic model. The binary model only cares about the detection range of the sensor and the Euclidian distance between the sensor and the point under consideration. It assumes that if the point to be sensed is within a sensor's sensing radius then the sensor will with absolute certainty be able to sense the point.

The probabilistic model, on the other hand, states that the probability of detection is not constant and degrades as a factor of distance, environmental condition etc. The probabilistic model is expressed using the equation (2)

$$C_{ab} = \begin{cases} 0 & \text{if } r + reff \leq D(S_i, X) \\ \frac{-\gamma_1 \alpha_1^{\beta_1}}{e \alpha_2^{\beta_2}} + \gamma_2 & \text{if } r - reff < D(S_i, X) < r + reff \\ 1 & \text{if } D(S_i, X) \leq r - reff \end{cases} \quad (2)$$

Here $\gamma_1, \beta_1, \beta_2$ are measuring parameters, $\alpha_1 = r_{eff} - r + D(S_i, X)$ and $\alpha_2 = r_{eff} + r - D(S_i, X)$ and γ_2 is the disturbing factor, r_{eff} denotes the uncertainty in the detection range.

In our work, we have adopted the probabilistic model of detection. If an area X_{ovp} is covered by k_{ovp} sensors then a measure of the coverage is given by equation (3)

$$C_{ab}(X_{ovp}) = 1 - \prod_{si \in Sovp} (1 - Cab(Si)) \quad (3)$$

The fitness of a particular coverage can be calculated with respect to a coverage threshold (C_{thres}) as in equation (4).

$$C_{ab}(X_{ovp}) \geq C_{thres} \quad (4)$$

4. DYNAMIC DEPLOYMENT OF SENSOR NETWORK USING ARTIFICIAL BEE COLONY ALGORITHM

Artificial Bee Colony algorithm was developed by (Karaboga and Akay, 2009). The ABC algorithm mimics the foraging behavior of honey bees. The food sources in the ABC algorithm denote a solution in the search space. The fitness of a food source is determined by its nectar value. In ABC algorithm a colony of bees is used as the swarm, the swarm of bees is made up of three types of bees namely the employee bee, the onlooker bee and the scout bee. The bee that initially wanders in search of food is called the employee bee, the bees that wait in the dancing area to choose a food source that was previously found by an employee bee are called the onlooker bees, and lastly the bee that looks for new food sources to replace the abandoned food source is called the scout bee.

(Ozturk et al., 2011) were the first to use the ABC metaheuristic to direct the movements of sensors in a sensing field for maximizing coverage area. The authors used a sensor network that comprised of both static and mobile sensors. The authors used a food source of the ABC algorithm to represent a deployment sequence which contains the positions of the sensors in the mission area and the coverage of the deployment which was calculated using equation (1) as its nectar value. The results showed that the ABC algorithm outperformed the PSO algorithm by generating a deployment sequence that covered more area.

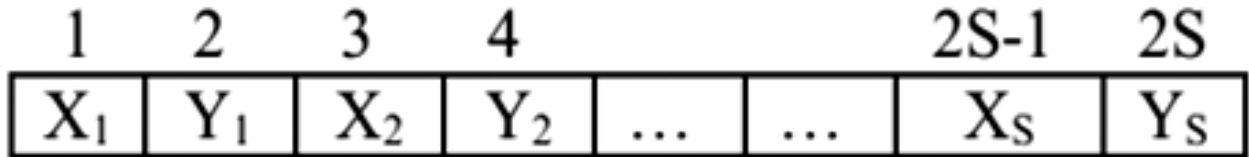


Figure 1: A food source of the ABC algorithm here X_i, Y_i represents the Cartesian coordinates of the sensor S_i

5. PROPOSED WORK

A potential area of improvement in the ABC is in its local search. The ABC algorithm uses the equation (5) for exploring around an existing solution (local search) and equation (6) for calculating its normalized fitness value in the interval (0, 1).

$$X_{ij} = Y_{ij} + \varphi * (Y_{kj} - Y_{ij}) \tag{5}$$

$$P_i = \frac{fitness_i}{\sum_{j=1}^{No. of food sources} fitness_j} \tag{6}$$

Both the employee bee as well as the onlooker bee use the same equations for finding new solutions. Researchers have studied the local search characteristic of the algorithm and have suggested multiple approaches for improving searching capability of the algorithm, their study exposed two areas where the local search falls short, namely the undirected or haphazard search for new food sources around old food source and the way it balances exploration and exploitation.

One approach suggested by researchers for overcoming the first shortcoming of the ABC is to modify the local search so that instead of randomly searching around an old solution, the search for new solutions be guided along the global best solution found till that instant. The results of their experiments using the modified local search in the canonical ABC algorithm. The global best guided local search expressed in equation (7).

$$X_{ij} = gBest_j + \varphi * (Y_{kj} - Y_{ij}) \tag{7}$$

Where, gBest_j represents the jth parameter of the global best solution found so far, Y_{kj} and Y_{ij} represent the current solution under consideration and a randomly generated solution around the current solution respectively and φ is a random number ranging from -1 to 1.

The second shortcoming of the ABC is in the way it balances exploration and exploitation phase. The studies done in the past pointed out the fact that the local search of ABC favoured exploration to exploitation, which generally results in valuable domain knowledge gathered by the swarm of bees being thrown away. The solution to this problem is the use of variable length step size.

In our work, we incorporated a memetic search (Bansal et al., 2013) for balancing the exploration and exploitation. To summarize, the proposed hybrid local search contains the following computations.

Pseudo code for hybrid local search

For a given food source Y_{ij} , generate new solutions namely,
 U_{ij} using the local search of conventional ABC algorithm using equation (5)
 V_{ij} using the guided best search using equation (7)
 W_{ij} using memetic search
 Make a greedy selection between U_{ij} , V_{ij} , W_{ij} based on their fitness value.
 $X_{ij} = \max(\text{fitness}(U_{ij}), \text{fitness}(V_{ij}), \text{fitness}(W_{ij}))$

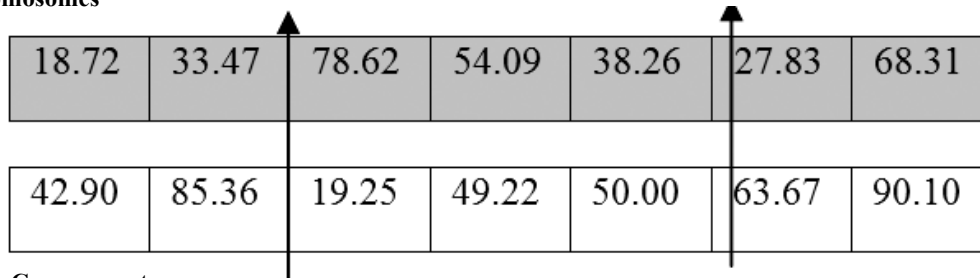
Another possible scope of improvement in the functioning of ABC algorithm is to use the fittest food sources food to generate new food sources. The ABC algorithm relies on a random search (equation (5)) that is based on selecting a random food source and a random dimension for new food sources and hence doesn't make use of the fittest solutions that have already been found.

In this paper, a crossover operator similar to the one used in Genetic Algorithm (GA) has been incorporated before send scout bee phase in ABC. The genotypes for the crossover operator is the food source as shown in Figure 1. For the crossover operator, a population of size equal half of number of food sources is generated using tournament selection.

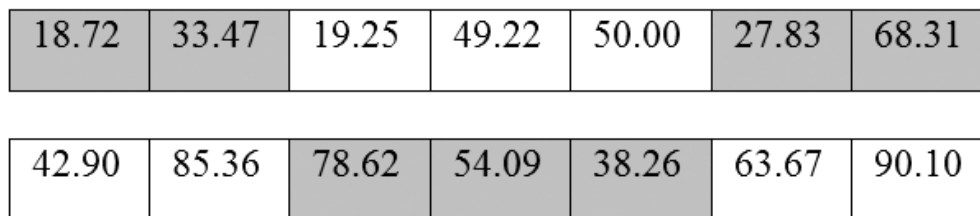
For generating the population we sorted the food sources (let say there were 'F' food sources in total) in decreasing order, in accordance to their fitness values and then we selected 3 random food sources out of them and included the fittest food source amongst the three, in the population. Once the population was created we chose a fixed amount of food sources from the median of the sorted food sources for replacement using crossover. The number of food sources chosen for replacement is the product of crossover probability and the number of food sources.

For each of the selected food source pick two parents randomly from the population and after generating two random crossover points we crossed the two parents to produce two new solutions(food sources) as shown in Figure 2. The selected food source is then replaced by the best among the two newly created food sources and itself.

Parent chromosomes



Crossover pts



Child chromosomes

Figure 2: Two point crossover

Pseudo code for dynamic node deployment using proposed algorithm

1. Initialize the Control Parameters of the ABC Algorithm: initialize the number of food sources, number of bees = colony size/2, limit etc., randomly generate the positions of the sensor nodes, perturbation rate = 0.4, $\epsilon = 0.01$, set the crossover_probability = 20%.
2. Set Iteration = 0
 Initialize the food source positions.
 Evaluate the nectar amount (fitness) of food sources.
3. Repeat Until (cycle is not equal to MAXCYCLE) for each employee bee do
 - a) Search the neighbourhood of the food source for new solutions using hybrid local search
 - a) Check if X_{ij} is within the bounds of the monitoring area.
 - b) Evaluate the fitness of the new food source using equation (6).
 - c) Make a greedy selection between old solution and the new solution.
4. Compute the probability $Prob_i$ of the solution using equation

$$Prob_i = \frac{0.9 * fitness_i}{fitness_{best}} + 0.1$$

5. For each onlooker bee do
 - a) Generate a random number $r \in (0, 1)$ and select a food source depending on the value of $Prob_i$ and r .
 - b) Look for new food source in the neighbourhood using hybrid local search
 - c) Check if the new solution is within the bounds of the are
 - d) Evaluate the fitness of the new food source using equation (6).
 - e) Make a greedy selection between old solution and the new solution.
6. Select a fixed number of food sources (N) for replacement using the formula
 $N = crossover_probability * number\ of\ food\ sources.$
 - a) Generate the mating pool for crossover operation using tournament selection.
 - b) For each selected food source do
 - i) Make a random selection of two food sources that will serve as parent genotype.
 - ii) Generate two random crossover points and swap genes as in Figure 2
 - iii) Make a three way greedy selection amongst the selected food source and the two newly created food sources as a candidate for replacing the selected food source.
7. Iterate through the trial array of food sources
if (the trials for a food source is greater than max trials)
 - a) Replace the food source with a randomly generated food source.
 - b) Set trial = 0.
8. Memorize the best solution found so far.
9. cycle = cycle +1.

6. RESULT ANALYSIS

In this paper, the results of our proposed algorithm has been compared with the ABC. The experimental setup is as follows, we modeled a sensor network with 45 mobile sensors. Each sensor has a sensing radius of 7m, the detection error r_{eff} is 3.5m, the monitoring region is a square area of 10000m², the value of the parameters of the probabilistic model are as follows: $\gamma_1=1$, $\gamma_2=0$, $\beta_1=1$, $\beta_2=0.5$, $C_{\text{thres}} = 0.9$. The crossover probability is 0.2 and the tournament population size is 20. The colony size is 80 and the number of food sources is 40. The scenario of a random deployment of sensors is run 30 times with each run having 1000 cycles.

For comparing the performance of our proposed algorithm with the ABC, in each iteration we generated random food sources that served as input for both the algorithms. Both algorithms worked on the same input

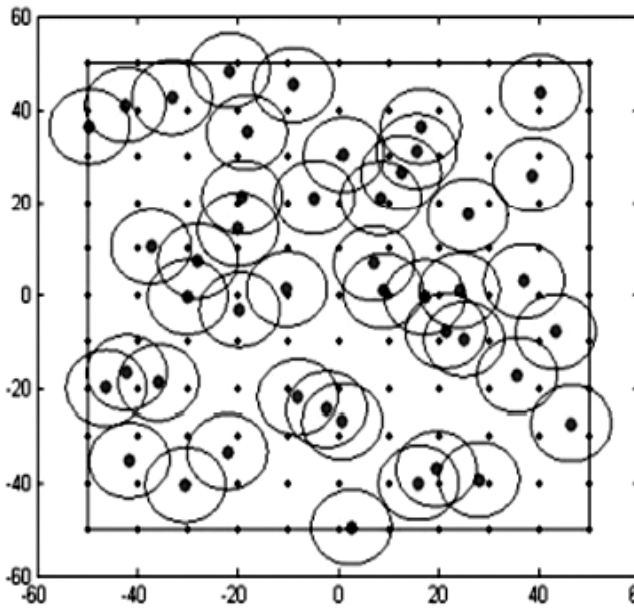


Figure 3

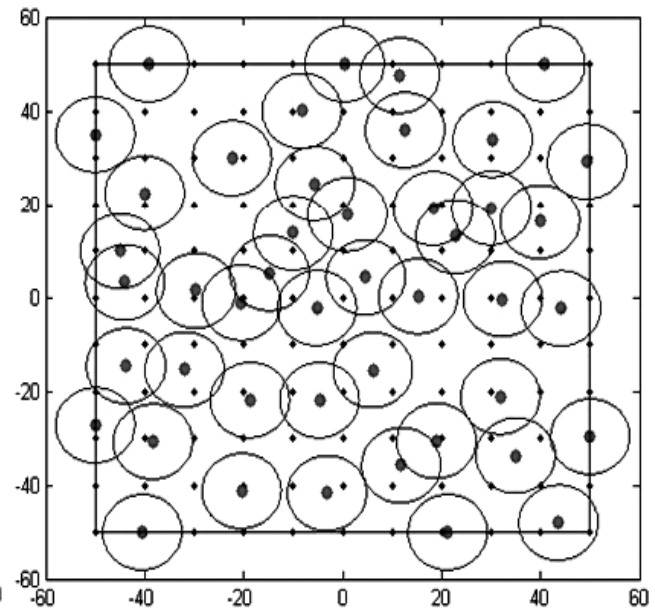


Figure 4

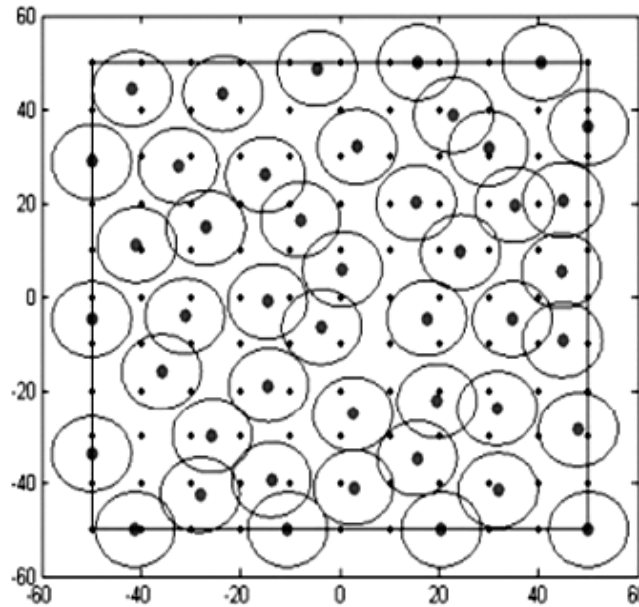


Figure 5

set and produced results in the form of a $1 \times 2S$ matrix, where ‘S’ is the number of sensors as shown in figure 1.

The Figure 3 shows the initial random deployment of sensors. The grid points are denoted in blue colour. The area was divided into squares of 5×5 dimensions, resulting in a total of 441 grid points, in the figure however 121 grid points (end points of squares of 10×10 dimensions) are shown rather than 441 for the sake of improved visibility. The points marked in red are the center points of the sensors whose sensing radius is pictorially represented by a circle.

Figure 4 shows the best deployment returned by the ABC algorithm starting with the deployment in figure 3 as input, the coverage of this deployment is 84.58%. The figure 5 shows the best deployment of the proposed algorithm, the coverage of this deployment is 89.79%. The figures 6.1 – 6.4 show the best deployment sequence found by ABC at various cycles. The deployment at 50th, 100th, 500th and 1000th cycle is shown in figures 6.1, 6.2, 6.3 and 6.4 respectively.

The figures 7.1-7.4 show the best deployment sequence found by our proposed algorithm at various cycles. The deployment at 50th, 100th, 500th and 1000th cycle is shown in figures 7.1, 7.2, 7.3 and 7.4 respectively.

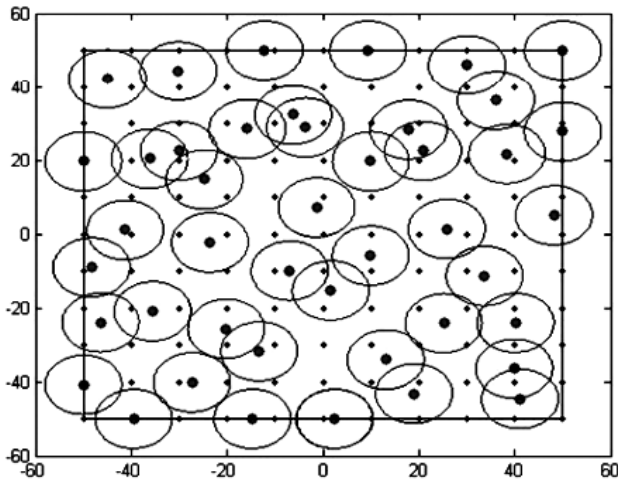


Figure 6.1

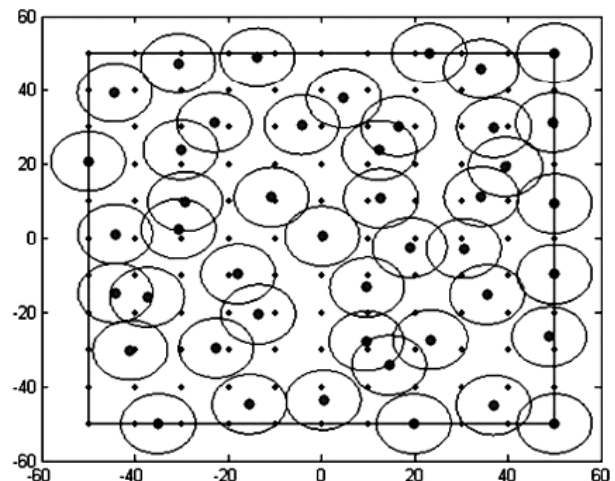


Figure 6.2

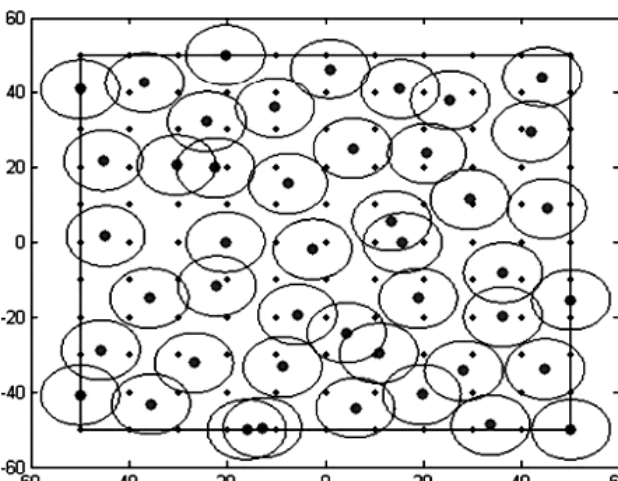


Figure 6.3

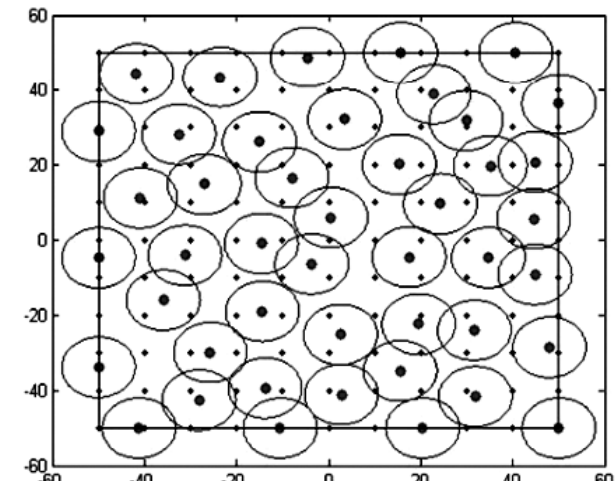


Figure 6.4

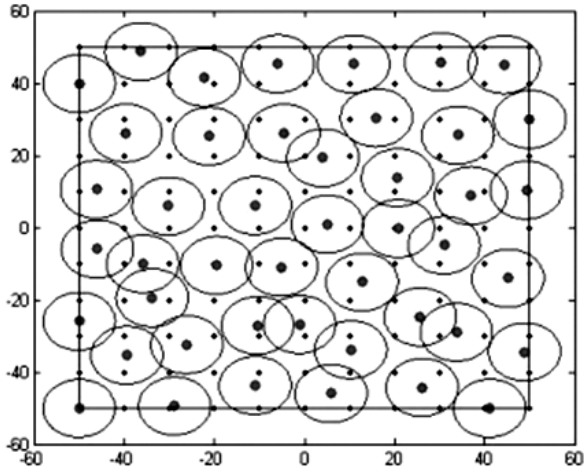


Figure 7.1

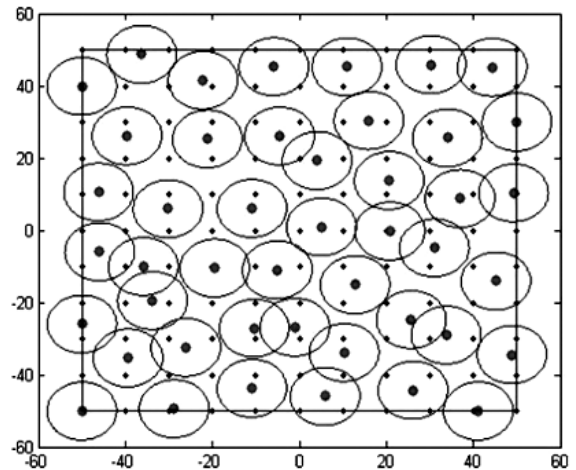


Figure 7.2

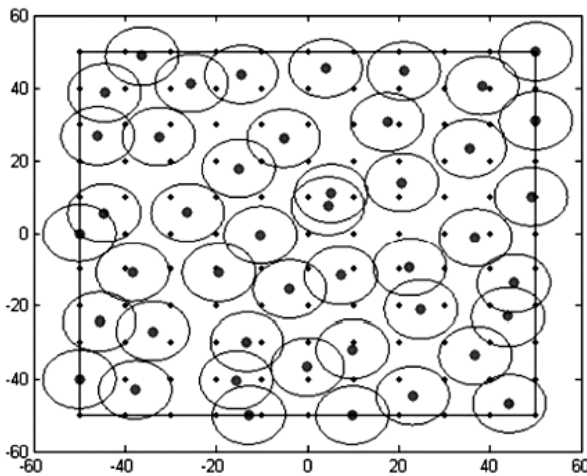


Figure 7.3

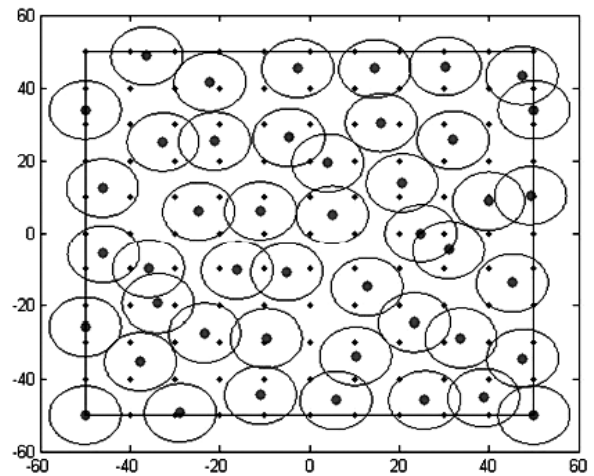


Figure 7.4

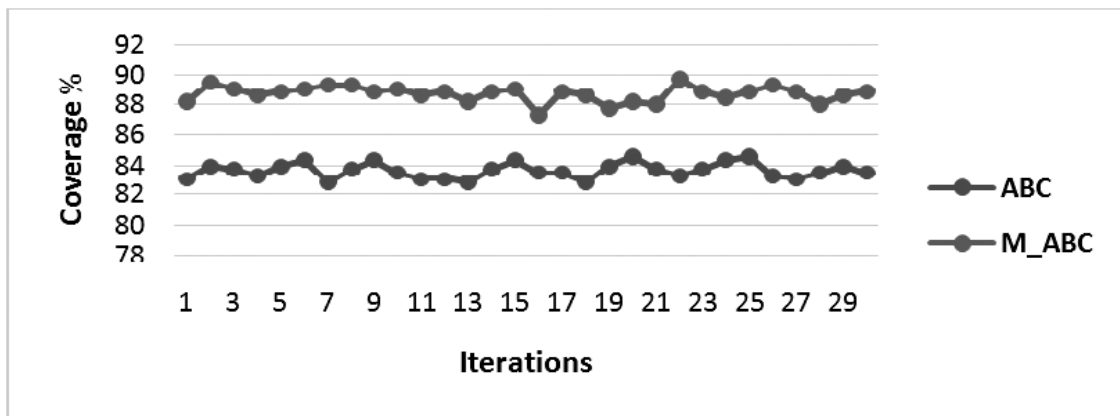


Figure 8

Figure 8 shows the execution of artificial bee colony algorithm (ABC) and our proposed algorithm i.e. Modified ABC (M_ABC). The mean coverage of ABC for this experimental setup was 83.58% whereas as that of our proposed algorithm was 88.84%.

7. CONCLUSION

In this paper we have demonstrated an ABC based technique for dynamic positioning of sensor network. We addressed two deficiencies of the canonical ABC algorithm using a hybrid local search and a crossover operator. The results showed that the proposed algorithm outperformed the ABC algorithm by finding positioning sequences having higher area coverage, hence the proposed changes improved the optimizing capability of the ABC algorithm when applied to the dynamic deployment problem of WSNs.

REFERENCES

- [1] A.P. Bhondekar, and R. Vij, "Genetic Algorithm Based Node Placement Methodology for Wireless Sensor Networks," Proceedings of the International Multi Conference of Engineers and Computer Scientists 2009 Vol I IMECS 2009, March 18 - 20, 2009.
- [2] R.ÖzdaL and J. A. Karc, "Sensor Node Deployment Based on Electromagnetism – Like Algorithm in Mobile Wireless Sensor Networks," International Journal of Distributed Sensor Networks Volume 2015, Article ID 507967, Hindawi Publishing Corporation.
- [3] Y.Zou and K. Chakrabarty, "Sensor deployment and target localization in distributed sensor networks," ACM Trans. Embedded Comput. Syst., vol. 3, no. 1, pp. 61–91, 2004.
- [4] C.H. Wu, K. Chuan and Y.C. Chung, "A Delaunay Triangulation Based Method for Wireless Sensor Network Deployment," in Proceedings of the 12th International Conference on Parallel and Distributed Systems (ICPADS'06), IEEE, 2006.
- [5] N. Kukunuru, B.R.Thlla and R. L. Davuluri, "Sensor deployment using particle swarm optimization," International Journal of Engineering Science and Technology, vol. 2, no. 10, pp. 5395– 5401, 2010.
- [6] G.Wang, G.Cao and T. La Porta, "Movement assisted sensor deployment," in Proceedings of IEEE Conference on Computer Communication (INFOCOM), Miami, March 2005.
- [7] M. Erdelj, T. Razafindralambo and D. Simplot Ryl, "Covering Points of Interest with Mobile Sensors," IEEE Transaction on Parallel and Distributed Systems, p. 32-43, 2013.
- [8] W. H. Liao, Y. C. Kao and Y. S. Li, "A Sensor Deployment Approach Using Glowworm Swarm Optimization Algorithm in Wireless Sensor Networks," Expert Systems with Applications, Vol. 38, No. 2011, pp. 12180–12188.
- [9] D. Karaboga and B. Akay, "A comparative study of Artificial Bee Colony algorithm," Applied Mathematics and Computation 214 (2009) 108–132.
- [10] C. Ozturk, D.Karaboga and B. Gorkemli, "Probabilistic dynamic deployment of wireless sensor networks by artificial bee colony algorithm," Sensors, vol. 11, no. 6, pp. 6056–6065, 2011.
- [11] X. Yan, Y. Zhu and W. Zou, "A Hybrid Artificial Bee Colony Algorithm for Numerical Function Optimization," 11th International Conference on Hybrid Intelligent Systems (HIS), IEEE 2011.
- [12] J.C. Bansal, H. Sharma, K.V. Arya and A. Nagar, "Memetic search in artificial bee colony algorithm," Soft Computing: A Fusion of Foundations, Methodologies and Applications, Springer 2013.
- [13] X. Wang, J.J. Ma, S. Wang and D.W. Bi, "Distributed Particle Swarm Optimization and Simulated Annealing for Energy-efficient Coverage in Wireless Sensor Networks," Sensors 2007, 7, 628-648, ISSN 1424-8220.
- [14] G. Wang, L. Guo, H. Duan, L. Liu, and H. Wang, "Dynamic Deployment of Wireless Sensor Networks by Biogeography Based Optimization Algorithm," Journal of Sensor and Actuator Networks, ISSN 2224-2708, 2012.
- [15] O. Banimelhem, M. Mowafi and W. Aljoby, "Genetic Algorithm Based Node Deployment in Hybrid Wireless Sensor Networks," Communications and Network, 2013, 5, 273-279, (<http://www.scirp.org/journal/cn>).
- [16] S. S. Dhillon and K. Chakrabarty, "Sensor placement for effective coverage and surveillance in distributed sensor networks," 2003 IEEE Wireless Communications and Networking, 2003. WCNC 2003., New Orleans, LA, USA, 2003, pp. 1609-1614 vol.3.