ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 6, June 2014

Novel Discrete Binary P.S.O based deployment of Wireless Sensor Network

Ankita Saini[1], Neelam Saini[2], Prachi[3]

M.Tech Scholar, Electronics and Communication Department, D.C.R.U.S.T, Murthal, Sonipat, India[1]

M.Tech, Electronics and Communication Department, D.C.R.U.S.T, Murthal, Sonipat, India[2]

Assistant Professor, Electronics and Communication Department, D.C.R.U.S.T., Murthal , Sonipat, India[3]

Abstract: Coverage is one of the main research interests in wireless sensor networks (WSNs), which is used to determine the quality of service (QoS) of the networks. Therefore, this paper aims to solve coverage problem in distributed wireless sensor network by increasing sensor nodes coverage based on Novel Discrete Binary PSO Algorithm. This algorithm is capable of very efficiently deploying the sensors with an objective of maximizing the coverage and minimizing the network cost. There is a requirement of a minimum number of sensors to cover the entire grid area; below that number it will not be possible to cover the entire work field efficiently. Therefore, if the WSN is not deployed efficiently, the coverage issue is raised with an error message to improve the coverage area. These issues in WSNs are formulated as multidimensional optimization problems, and approached through various bio-inspired techniques. Particle swarm optimization (PSO) is a easy to implement, effective and computationally efficient optimization algorithm. It has been applied to address various WSN issues such as optimal deployment of sensors, node localization, clustering and data-aggregation. PSO is a real value algorithm[1], and the discrete binary PSO is proposed to be adapted to discrete binary space. Simulations have been done to illustrate the significant and effective impact of this novel approach of discrete particle swarm optimizer on sensor placement in WSN.

Keywords: Novel Discrete Binary PSO, Distributed Sensor Placement, Coverage, Sensor Placement, Wireless sensor networks

I. INTRODUCTION

Wireless Sensor Networks (WSNs) have gained tremendous importance in recent years because of its potential use in a wide range of applications such as target tracking, surveillance and security management etc. Due to the increasing potential of WSN, along with its unique characteristics such as easy accessibility to difficult terrain, efficient transmission and user's friendly interface etc., there will be a significant spur of research in coming years in the field of developing network protocols specifically tailored for sensor networks.

Wireless Sensor Networks consist of a number of battery powered sensor nodes, endowed with physical sensing capabilities, limited processing, memory, and short-range radio communication [2]. In these networks a large set of nodes may be distributed over a wide geographical area, indoor or outdoor enabling a number of sensing and monitoring services in the areas of vital importance such as industrial production monitoring and environmental monitoring. These nodes collectively form a network and forward the gathered information to a data sink or gateway [3].

In general, a sensor node includes a sensing device for data acquisition from the physical environment, a processing subsystem for local data processing and storage, and a wireless communication module. Additionally, a power source supplies the energy needed by the device to perform all the programmed tasks. The power source in most of the cases is battery having limited life. Therefore, maximizing the network lifetime is also an important challenge and should be tackled either at the planning stage or by means of a recursive optimization of the network lifetime under minimal coverage constraint. Thus, in order to support planning and deployment as well as to enable testing of new protocols and applications, simulation platforms have been extended to include simulation frameworks for WSN. While studying the coverage area, most of the researchers assumed that the sensor nodes are static[4]. However, these days new type of mobile sensor nodes are used which have the limited ability to relocate themselves after their deployment. Different algorithms [5], [6], [7], [8] have been proposed by researchers to relocate the sensor nodes to optimize the coverage and time. The method proposed by Howard et al. [5] uses iterative sequences for determining location each sensor node needs to move to in order to optimize the coverage. Traditional analytical optimization techniques require a lot of computational efforts, which grows exponentially as the problem size increases. An optimization method that requires moderate memory and computational resources and yet produces good results is desirable for implementation on an individual sensor node. Optimization methods inspired by biological activities are computationally efficient alternatives to analytical methods. Particle swarm optimization (PSO) is a popular multidimensional optimization technique [9]. Ease of implementation, high quality solutions, computational efficiency and speed of convergence are strengths of PSO[10]. Keeping in mind the deployment needs of sensors in wireless network and the characteristics of PSO a new algorithm has been proposed to deploy wireless sensor network using novel discrete binary particle swarm optimization technique. The algorithm is implemented and illustrated using MATLAB for the purpose.

II. BACKGROUND OF PARTICLE SWARM OPTIMIZATION

PSO is inspired by observing the bird flocking or fish school [10]. Scientists found that the synchrony of flocking behavior was through maintaining optimal distances between individual members and their neighbours. Thus, velocity plays an important role of adjusting each other for the optimal distance. Furthermore, the scientists simulated the scenario in which birds search for food and observed their social behavior [12]. They perceived that in order to find food the individual members determined their velocities by two factors, their own best experience of past and the best experience of all other members [10]. This is similar to the human behavior in making decision where people consider their own best past experience and the best experience of how the other people around them have performed [11].are required. According to the above concept, Kennedy and Eberhart [10] developed the so-called PSO for optimization of continuous nonlinear functions in 1995. In PSO algorithm, each answer to the problem is considered as a bird in the search space which is called a particle. Each particle has its own fitness determined by the fitness function. A bird which is close to food source has a better fitness.

A. The PSO Algorithm

In PSO algorithm the individuals, called particles are "flown" through a multidimensional search space. Optimization of the multidimensional search space using the PSO requires very simple operations for creating a computationally efficient algorithm. After the PSO selects the most likely parameters for an optimum solution, it multiplies them by a uniform random term, which prevents premature convergence which is a major concern for such search space. Particles flown at the beginning of the PSO algorithm remain fully functioning until the solution is found.

The movement of the particles in the swarm is influenced by two factors:

- the particle's local iteration-to-iteration best solution called "*pbest*" and
- the particle's global particle-to-particle best solution called "*gbest*".

As a result of iteration-to-iteration information, each particle stores in its memory the best solution it has visited so far, called "*pbest*", and experiences an attraction towards this solution as it traverses through the solution search space. This attraction is strong if the best solution is far from the current particle's location and not related to its performance. Due to the particle-to-particle information, the particle stores in its memory the best solution visited by any particle and an attraction towards this solution, called "*gbest*". The *pbest*, and *gbest*, factors are called the cognitive and social components, respectively. After each iteration the *pbest* and *gbest* are updated for each particle if better, more dominating solutions (in terms of performance or fitness) are found. This process continues, iteratively, until either the algorithm achieves the desired result, or until an acceptable solution cannot be found within computational limits determined by the application.

The PSO defines each particle in the D-dimensional space as $X_i = (x_{i1}, x_{i2,\dots\dots,} x_{iD})$, where the subscript '*i*' represents the particle number and the second subscript is the dimension number of parameters defining the solution. The memory of the best position achieved previously is represented as $P_i = (p_{i1}, p_{i2}, \dots \dots\dots, p_{iD})$,and velocity of each iteration, the velocity term is updated, and the particle is moved in the direction of its own best position, *pbest*, and to the global best position, *gbest*. This is shown in the velocity update equation as

$$V_{id}^{(t+1)} = \omega \times V_{id}^{(t)} + U[0,1] \times \psi_1 \times \left(p_{id}^{(t)} - x_{id}^{(t)}\right) + U[0,1] \times \psi_2 \times \left(p_{gd}^{(t)} - x_{id}^{(t)}\right)$$

The position is updated using the velocity (1) and

$$X_{id}^{(t+:} \qquad (2)$$

where U [0,1] samples a uniform random distribution, '*t*' is a relative time index ψ_1 and ψ_2 are weights trading off the impact of the local best and global best solutions' on the particle's total velocity. Here represents the personal best position of particle in the dimension and represents the global best position of the particle in the dimension.

B. DISCRETE PARTICLE SWARM OPTIMIZATION

In the above discussion, PSO is restricted in real and continuous number space. However, many optimization problems are set in a space featuring discrete or qualitative distinctions between variables. To meet the need, Kennedy and Eberhart [10] developed a discrete version of PSO which deals with discrete sample space. Discrete PSO differs from the original or continuous PSO in two characteristics. First, the position particle is considered as binary variable. Second, the velocity must be changed into the probability, which is the condition of the binary variable taking the value one [12].

There are two types of Discrete PSO techniques:

1. Discrete Binary PSO
2. Discrete Multi-Valued PSO

Discrete Binary PSO Algorithm: In the discrete binary-valued space, the continuity has no meaning and as a result of this, the interpretation of the fitness function as a function of the position also loses its meaning. A discrete binary version of the algorithm transitions particles in a probabilistic space using the velocity of the particle. This has implied that the binary variables have a probability associated with them. The particle swarm tries to maximize the probability of a certain binary variable by having a velocity such that probability achieved is maximized. This algorithm uses the similar velocity update equation as in

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 6, June 2014
ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

(1) but the values of 'X' are now discrete and binary. For position update, first the velocity is transformed into an interval of (0, 1) using the sigmoid transformation function given by

$$S_{id} = sig(V_{id}) = 1/(1 + e^{-V_{id}}) \qquad (3)$$

where, V_{id} is the velocity of the i^{th} particle in d^{th} dimension. A random number is generated using a uniform distribution which is compared to the value generated from the sigmoidal transformation function*sig* and a decision is made about the X_{id} in the same manner. Here, S_{id} represents the binary value position of the particle after applying sigmoidal transformation to velocity component.

$$x_{id} = u(S_{id} - U[0,1]) \qquad (4)$$

Here *'u'* is a unit step function. The decision regarding X_{id} is now probabilistic, implying that higher the value of the V_{id}, higher will be the value of the S_{id}, making probability of deciding '1' for X_{id} higher. It should be noted that as Vid tends to infinty,Sid approaches to 1, making it impossible forX_{id} to return to zero after that point. Until this point, there is some probability of X_{id} returning to zero. Figure 1 shows this property of the binary PSO. The probability of $X_{id} = 1$ increases as V_{id} increases. However,$P(X_{id} = 1)$ is almost equal to 1 for$V_{id} > 10$, but not exactly equal to 1.

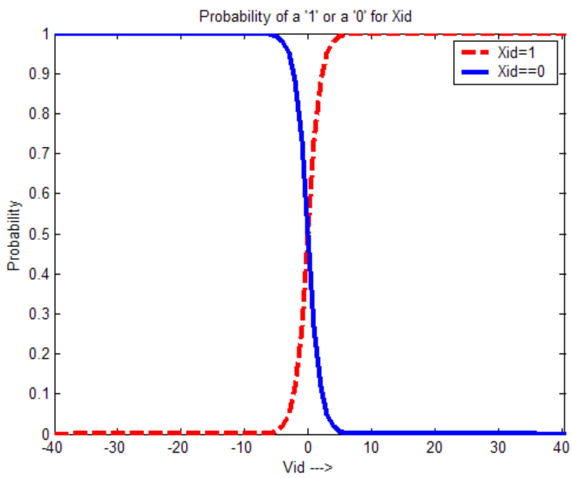

Fig.1 Transformation of particle velocity to a binary variable

Multi-Valued Particle Swarm Optimization : PSO for the multi-valued problems is designed similar to the design of binary PSO. For discrete multi valued optimization problems, the range of the variables lie between (0 , M-1), where 'M' implies the arbitrary number. The same velocity update and particle representation are used in this algorithm. The position update equation is however changed in the following manner. The velocity is first transformed into a number between (0, M-1) using the following sigmoid transformation

$$S_{id} = \frac{M}{1} + e^{-V_{id}} \qquad (5)$$

The positions of the particles are discrete values between (0, M-1). Note that for a given S_{id} , there is a probability of having any number between (0, M-1).. In the following graph , the relation between S_{id} and the probability of resultant discrete variables is given. The transformation of velocity of particle using a sigmoid function and the generation of the random variable from Gaussian distribution is illustrated in Figure 2.

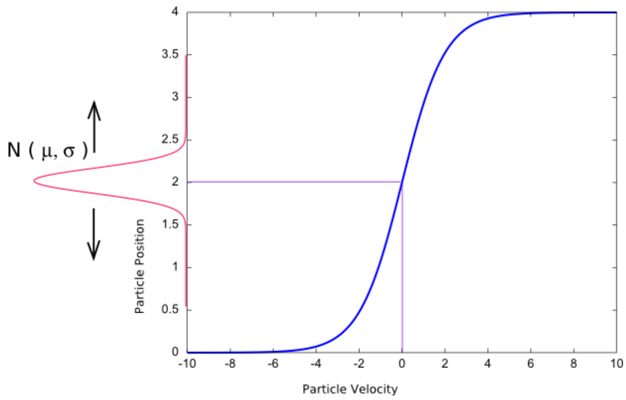

Figure 2 Transformation of the Particle Velocity to a Discrete Variable

III. PROBLEM DEFINITION

The grid-based sensor network could be considered as a two or three dimensional search space. Here, we are considering the 2-Dimensional solution space. A set of sensors are placed on different points in order to cover the sensor field. Here, we define a power vector in the algorithm for each point of the field to show whether these sensors could cover that point on the field or not.

IV. RESEARCH METHODOLOGY

In order to solve coverage problem in Wireless Sensor Network (WSN), Novel Discrete Binary Particle Swarm Optimization Algorithm is employed to locate the sensors in region of interest and hence the optimized placement of sensors in the WSN are achieved. We consider a network with homogeneous sensors where sensing radius for all sensors is similar and the sensors know their positions respectively. Besides that, coverage percentages in WSN are calculated as :

% Coverage = (Grid points covered / Total grid points) * 100 %

The interest points set are consisted of the grid points of the grid square which is obtained from the computed grid diagram and a number of points distributed evenly on the boundary of the grid. Then, dividing the number of covered grid points with the total grid points in the grid diagram can determine the total area of coverage. If the percentage coverage is greater than or equal to 100% i.e all the grid points are covered , then it is regarded as the covered area is optimum ,but if, the percentage coverage is less than 100%, a message is generated for the user to increase the number of sensors to cover the region of interest. Then ,this algorithm is executed at a base station, and thereafter the base station will transmit the sensors'

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 6, June 2014

final optimal positions to the sensors to move to their optimal positions based on this information.

V. Simulation and Results

To implement the proposed Discrete Binary PSO algorithm, a MATLAB code was developed to simulate the algorithm with different specifications of the WSN. Test cases are conducted to study the effect of varying parameters of sensors and ROI in WSN. In this simulation, some basic parameters for placement of sensors in WSN are identified: Length of Grid, Width of Grid and Number of sensors employed. The coverage area is constructed by using the area formula: Area = pi*(sensing radius)^2 and the radius of the sensors is predefined. From the simulation diagrams, the green color dots indicate grid points and blue color indicates sensor points. Sensor points are on grid points. The blue dotted circles around each sensor points represent the coverage area.

To demonstrate the implementation of the proposed algorithm, comparison between the 2 cases is done in which :

- Case1 deals with the result when all the grid points in the 2-D search space are not covered and the optimized solution is not obtained. It is then, informed to the user to increase the number of sensors to cover the ROI as shown in Fig. 3
- Case 2 deals with the result when all the grid points in the 2-D search space are covered and optimized solution is obtained as shown in Fig. 4

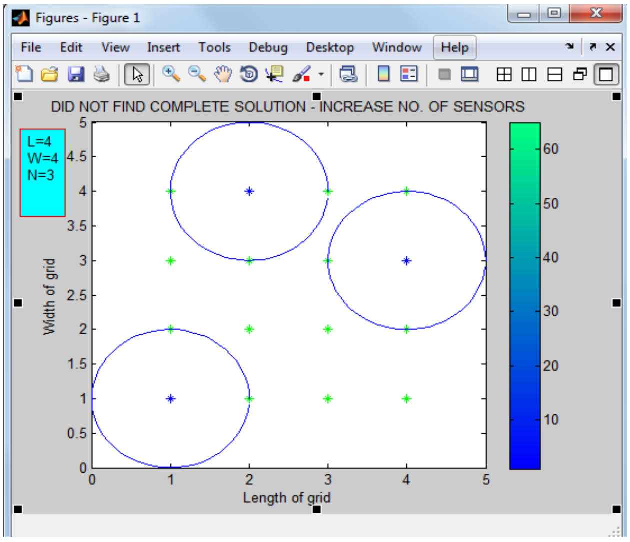

Fig 3. WSN deployment with insufficient number of sensors

The proposed discrete binary particle swarm optimization algorithm for deployment of wireless sensor network is governed by the number, locations and types of sensors within the wireless sensor network.

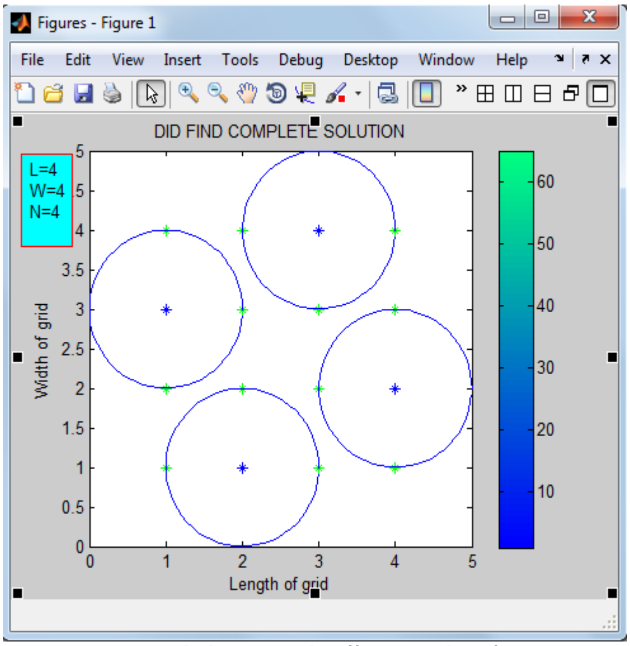

Fig. 4 WSN deployment with sufficient number of sensors

Now, here are some examples showing optimum coverage in WSN:

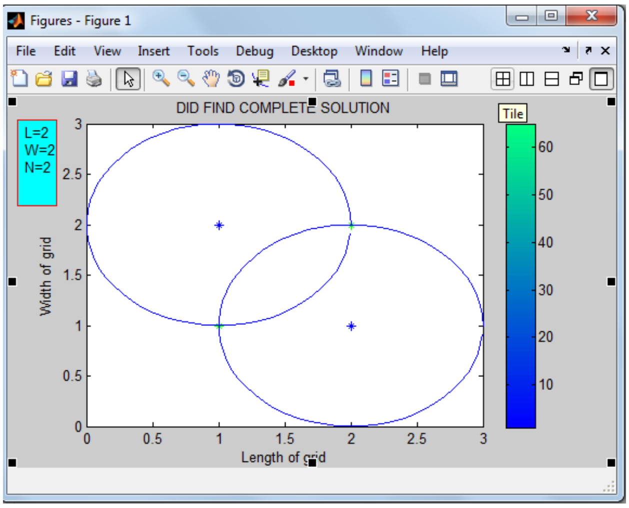

Fig 5: Area of 2*2 units square covered by 2 sensors

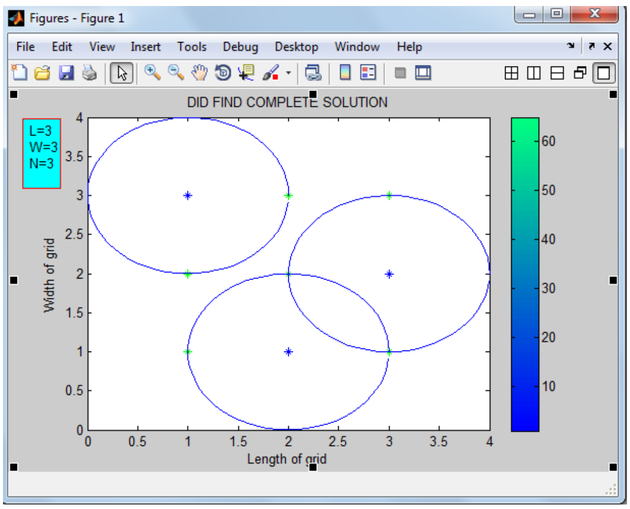

Fig 6 : Area of 3*3 covered by 3 sensors

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

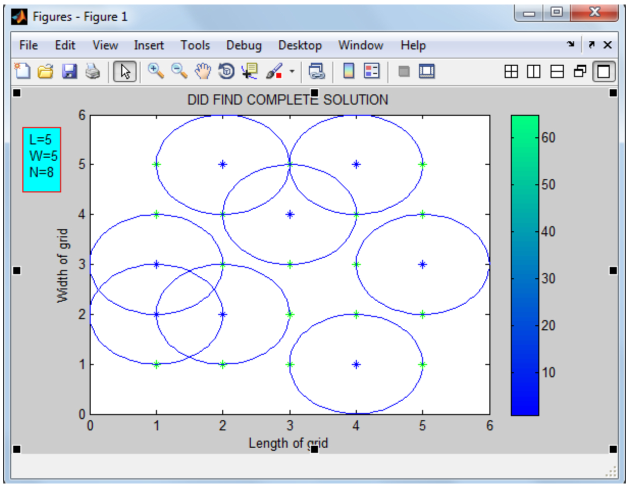

Fig 7: Area of 5*5 covered by 8 sensors

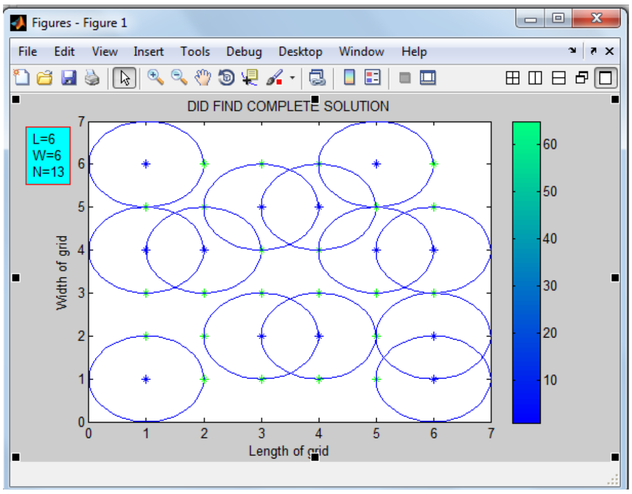

Fig 8: Area of 6*6 covered by 13 sensors

Table 1 displays the minimum number of sensor nodes required for optimum placement of sensors in ROI having varied numbers of grid points and sensors for providing complete coverage.

Table 1 :Displaying Grid Area Data Vs Sensors deployed to cover ROI completely

Sol. No.	Grid Area	No. of Sensors
1	2*2=4	2
2	3*3=9	3
3	4*4=16	4
4	5*5=25	8
5	6*6=36	13
6	7*7=49	19
7	8*8=64	25
8	9*9=81	31
9	10*10=100	41

VI. Conclusion

The following conclusions have been drawn from the various plots obtained after implementing Novel Discrete Binary Particle Swarm Optimization technique:

1. This technique has been acknowledged to be a simple, fast, computationally efficient and an easy to implement technique and is capable of very efficiently deploying the sensors with an aim of maximizing the coverage area and minimizing the network cost.
2. There is a requirement of a minimum number of sensors to cover the entire region of interest and below that specified number it will not be possible to cover the entire work field efficiently.
3. With the increase in grid node density of WSN, the number of sensors used also increase and covers area efficiently.

Acknowledgement

I take this opportunity of thanking my dissertation guide Mrs. Prachi, Assistant Professor, Department of Electronics and Communication Engineering, DCRUST, Murthal, Haryana for her indispensable guidance, generous help, perpetual encouragement, constant attention offered throughout in preparation of this dissertation. I am grateful for the support and guidance provided by her.

References

[1] M.Golshahi, M.Mosleh, M.Kheyrandish, "Implementing An ACO Routing Algorithm For AD-HOC Networlcs," IE,EE International Conference on Advanced Computer Theory and Engineering, 2008.
[2] I.F. Akyildiz, W.Su, Y. Sankarasubranmaniam Wireless sensor networks: a survey. Computer Networks Journal, 2002 , 38(4) : 393-422.
[3] Zhao F. and Guibas L. (2004) "Wireless Sensor Networks: An Information Processing Approach", Morgan Kaufmann, United States of America, pp.:30-254
[4] Ghosh A. and Das S. K., (2006) "Coverage and Connectivity Issues in Wireless Sensor Networks" in Shorey R., Ananda A. L, Chan MOoi W.T "Mobile, Wireless and Sensor Networks: Technology, Applications and Future Directions", John Wiley & Sons, Inc pp.:303-334.
[5] A. Howard, M. J. Matari´c, and G. S. Sukhatme, "An incremental self-deployment algorithm for mobile sensor networks," Autonomous Robots, vol. 13(2), pp. 113–126, 2002.
[6] X.Y. Li, P. J. Wan, and O. Frieder, "Coverage in Wireless Ad-hoc Sensor Networks," IEEE Trans. on Computers, vol. 52, pp. 753-763,2002.
[7] B. Cărbunar, A. Grama, J. Vitek, and O. Cărbunar, "Redundancy and coverage detection in sensor networks," ACM Trans. on Sensor Networks, vol. 2(1), pp. 94-128, 2006.
[8] G. Xing, et al., "Integrated coverage and connectivity configuration for energy conservation in sensor networks," ACM Trans. On Sensor Networks, vol. 1(1), pp. 36-72, 2005.
[9] J. Kennedy and R. Eberhart, "Particle swarm optimization," in Proc. of IEEE Int. Conf. on Neural Networks, vol. IV, pp. 1942-1948, Perth, Australia, 1995.
[10] Shi and R.C. Eberhart, "A modified particle swarm optimizer," in Proc. of IEEE Int. Conf. on Evolutionary Computation, pp. 69-73, Anchorage, AK, USA, 1998.
[11] Kennedy J, Eberhart RC, Shi Y. Swarm intelligence. San Francisco, CA: Morgan Kaufmann; 2001.
[12] Ching-Jong Liao, Chao-Tang Tseng, Pin Luarn, "A discrete version of particle swarm optimization for flowshop scheduling problems", Computers & Operations Research 34, pp.3099 – 3111,2007