A

Dissertation

On

POSITIONING OF BASE STATION FOR MAXIMIZATION OF LIFETIME IN WSN USING MODIFIED ABC

Submitted in Partial Fulfillment of Requirement

For the Award of the Degree of

Master of Technology

in

Computer Science and Engineering

by

Jayant Likhar

University Roll No. 2K14/CSE/22

Under Esteemed Guidance of

Vinod Kumar

Associate Professor, Computer Engineering Department, DTU

2014-2016

Delhi Technological University

Delhi-110042, India

ABSTRACT

Wireless sensor networks consists of large number of resource constrained sensor nodes which performs the functionality of data gathering and transfer their data to the base station for analysis to gain meaningful information from it. In recent years, wireless sensor network has found its applications in varied areas like border surveillance, hazard management, security, weather monitoring etc. With such critical and real time applications it is very important to make wireless sensor network reliable and robust. One of the open problems in wireless sensor network is optimal positioning of the base station

In this dissertation, an approach to solve the base station positioning problem in wireless sensor networks using Artificial Bee Colony (ABC) optimization technique is presented. Along with conventional ABC approach, five modified ABC approaches are presented in this work. The first modification incorporates Grenade Explosion method instead of local search mechanism of conventional ABC and also applies Cauchy operator while exploration. The second modification uses three neighbors as reference while local search, it also selects the worst food source and replaces it at the end of every cycle. Third modification is based on Chaotic search where it creates a chaotic sequence which helps in exploitation and searching a new food source within some defined radius R. Fourth modification uses DRF GBest method where it uses two dynamic regulatory factors to control the perturbation towards the global best solution.

Fifth modification uses a novel method which enhances the local search ability of ABC while exploitation using the global best solution as reference. All of these modified algorithms are simulated and results are compared with the conventional ABC algorithm, which was outperformed by all six modifications in finding a location for the base station for maximizing lifetime of the network under same constraints.

Keywords: Artificial Bee Colony, network lifetime, wireless sensor network, base station positioning.

ACKNOWLEDGMENT

I would like to express my deep sense of respect and gratitude to my project supervisor Mr. Vinod Kumar for providing the opportunity of carrying out this project and being the guiding force behind this work. I am deeply indebted to him for the support, advice and encouragement he provided without which the project could not have been a success.

I am grateful to Prof. O.P.Verma, HOD, Computer Engineering Department, DTU for his immense support. I would also like to acknowledge Delhi Technological University library and staff for providing the right academic resources and environment for this work to be carried out.

Last but not the least I would like to express sincere gratitude to my parents and friends for constantly encouraging me during the completion of work.

Jayant likhar
Roll No. 2K14/CSE/22
M.Tech (Computer Science & Engineering)
Department of Computer Engineering
Delhi Technological University
Delhi-110042

CERTIFICATE

This is to certify that the project report entitled **"POSITIONING OF BASE STATION FOR MAXIMIZATION OF LIFETIME IN WSN USING MODIFIED ABC"** is a bonafide record of work carried out by **Jayant likhar (2K14/CSE/22)** under my guidance and supervision, during the academic session 2014-2016 in partial fulfillment of the requirement for the degree of Master of Technology in Computer Science & Engineering from Delhi Technological University, Delhi.

Vinod Kumar

Associate professor

(Project Guide)

Department of Computer Engineering

Delhi Technological University

Delhi-110042

Table of Content

List of Figures

List of Abbreviations

WSN	Wireless Sensor Network
SN	Sensor Node
BS	Base Station
CH	Cluster Head
NP	Non Polynomial
PSO	Particle Swarm Optimization
GA	Genetic Algorithm
ABC	Artificial Bee Colony
ACO	Ant Colony Optimization
ABCCG	Grenade Explosion method and Cauchy operator based ABC
Diff_ABC	Different ABC
CABC	Chaotic search based ABC
DRF_GBest ABC	Dynamic Regulatory Factor based GBest guided ABC
NABC	Novel ABC
MEC	Minimum Enclosing Cicle

CHAPTER 1

INTRODUCTION

Wireless sensor network is a wireless interconnection of sensor units which are predominantly used to monitor and observe environmental conditions. ad-hoc network contains Wireless sensor network as its subfield. Although Wireless sensor network is a subfield of ad-hoc network but still we cannot use techniques and mechanisms used for managing ad-hoc network for WSN because of the following reasons:

- Numbers of nodes in a WSN are far more greater than number of nodes than that of ad-hoc network.
- Nodes in WSN works in resource constrained environment, whereas in ad-hoc network nodes, there is no such constraints.

Because of the reason (discussed above) WSN behaves differently than ad-hoc network. The energy constraint on WSN nodes is of utmost importance which is studied intensively to make WSN robust, dependable and usable in real life scenarios. Sensor nodes work with limited amount of energy provided by a small DC source incorporated in the sensor node unit which cannot be replaced after the network is deployed. Along with energy other important constraints of WSN are:

- Limited sensing range.
- Limited energy.
- Limited transmitting range.
- Limited processing capability of sensor node.
- Limited memory capacity of sensor node.

The sensing range of a sensor node represents the range of area around the sensor which can be observed or monitored by it while the transmitting range represents the range of area around the sensor till where the sensor node can send data collected by it. The two operations, sensing and transmitting are the two most expensive operation performed by a sensor node therefore these two constraints are important. Limited processing and memory capacity comes into light when the sensor nodes preprocess the data before sending it. In preprocessing process, data is being compressed and the amount data to be transmitted is reduced which in result reduces the energy consumption. We can easily identify the three main operations of a sensor node in WSN as:

- Processing
- Communication
- Sensing

Sensing includes gathering the data about the parameters in question in the sensing range of the sensor. These parameters could be anything like temperature, humidity, etc depending on the motive and design of the Wireless Sensing Network. Processing includes the mechanism of managing the data gathered in the sensing operation, interpreting and manipulating data . Communication includes the transmission of data to other nodes in the network to gather some useful information from the data gathered. Communication includes routing of data packets which itself is a typical problem in standard communication network. So, routing in constrained environment is much more difficult problem to face.

1.1 Architecture of WSN

The scalability and performance of a Wireless Sensing Network highly depends on architecture of the network. The network is generally designed according the application area for which we are designing the network. The design chosen depends on the performance measures and to which extent we want to scale the network. The architecture of a WSN are generally of multiple types, but the two mail architectures in-depth studied are as follows:

- Flat
- Hierarchical or Two tier

1.1.1 Flat Architecture

In the figure shown below, In flat architecture , we have multiple sensor nodes (SN) which are deployed in the sensing area to gather useful data. Along with multiple sensor nodes we have a single base station (BS) to which every sensor node sends its gathered

data to. In flat architecture, sensor node sends its data using multi-hop transmission in which sensor node sends data to another sensor node in its transmission range,and then that intermediate sensor node sends the data received to the base station provided that the sensor node is placed far away from base station and not reachable in its transmitting range then.
A general structure of flat architecture of WSN is shown below in figure.

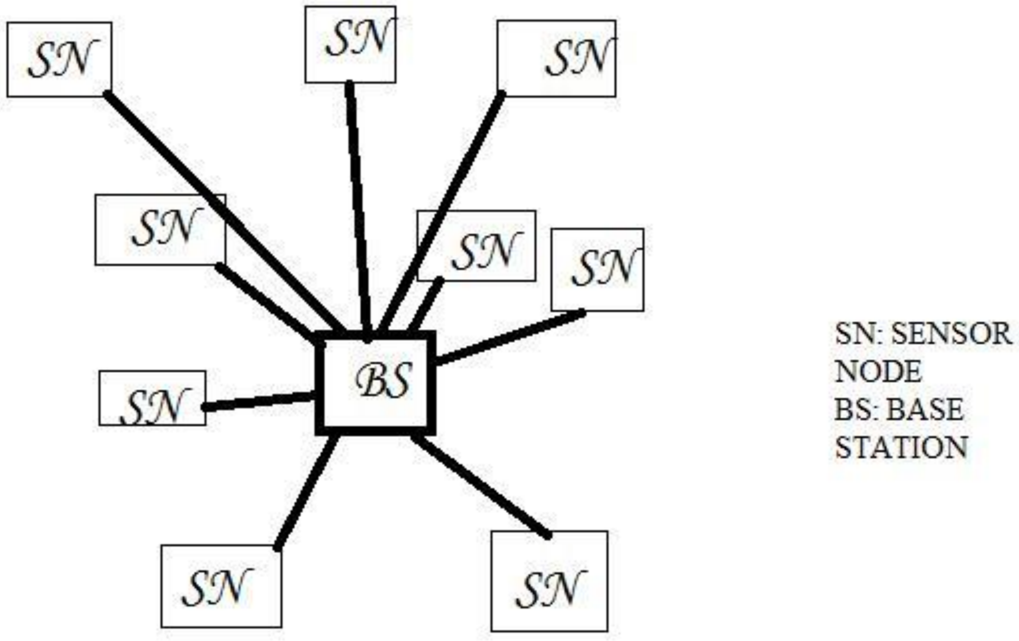

Fig.1.1 Flat Architecture of WSN

1.1.2 Hierarchical Architecture

In figure below shown, In Hierarchical Architecture ,useful data in the sensing range of the multiple sensor nodes is gathered by the multiple sensor nodes deployed. As shown in figure 2, in hierarchical architecture, There are multiple clusters created and each cluster has a cluster head which collects the data from every sensor in its cluster and then sent it to the base station itself .The procedure of selection of cluster head is typical one in Wireless Sensing Network. There are variety of techniques used for selecting cluster head (CH), one of them, changes the cluster head periodically ,in this readily used technique select a sensor node within that cluster as cluster head for specific period of time and after that select another node in the network as cluster head for the next time period and relieve the previous one from the duties of cluster head. Here the cluster head is changing ,In this way, the energy consumption of all the sensors in the cluster is managed to make sure that a single designated cluster head does not die (run out of energy) considerably before other nodes in cluster.

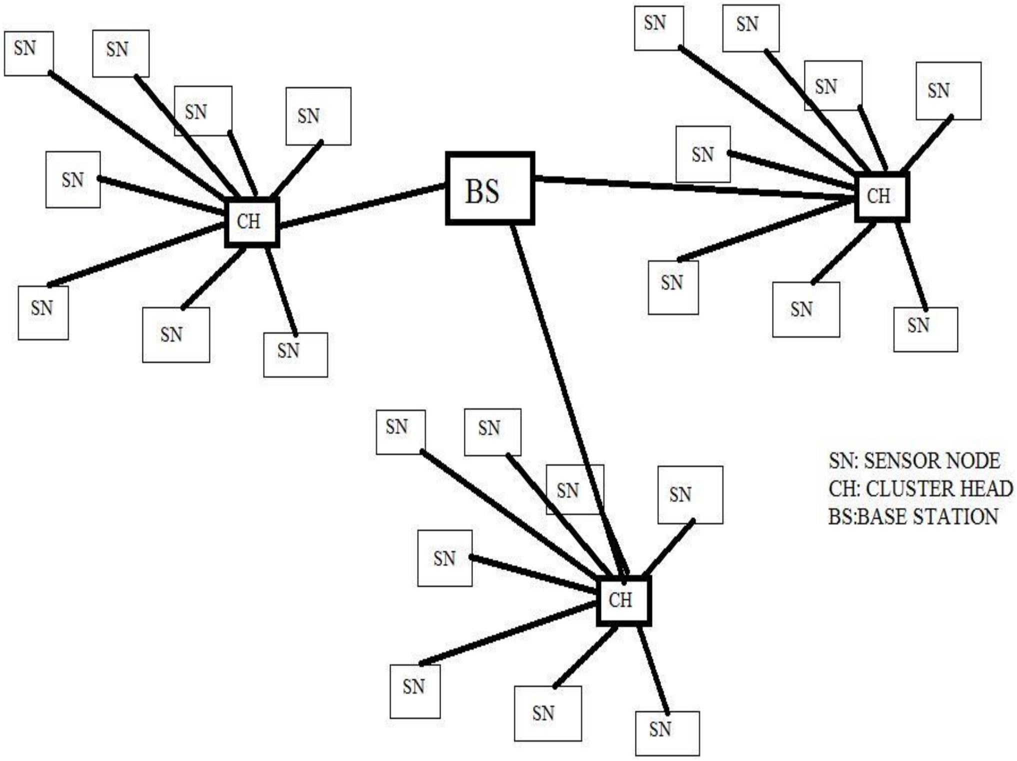

Fig.1.2 Hierarchical Architecture of WSN

In hierarchical architecture too we can have multi-hop transmission. When a cluster head is far away from base station such that the base station is not in cluster head's transmission range then the cluster head can transfer the data gathered by it from all the sensor nodes in its cluster to cluster head of another cluster that can transmit the data received to the base station.

1.2 Components of WSN

The components of Wireless Sensing Network is required for wireless connectivity within the network, connecting multiple sensor nodes to the base station and cluster heads. These are the main components of a WSN:

- Sensor Node
- Base Station

1.2.1 Sensor Node

A sensor node consists of few sensors and a processing unit. Sensor node is required to collect useful data around them in their sensing range and to send it to the base station. These are the following five main components of a sensor node:

- Controller
- Memory
- Transceiver
- power source
- Sensors

Controller is considered as the brain of the sensor node. It controls all the operations and functioning of a sensor node. it requires Memory unit to store the data collected by the sensors before sending it to the base station or any intermediate node (or relay node). Transceiver is a component which is used for communication purposes. It is used for receiving and sending data amongst nodes. Power source is the component which provides energy to all the components of sensor node. Power source uses a lithium-ion battery for storing energy. Sensors are the hardware device which senses the area surrounding the sensor for physical parameters like temperature, pressure, humidity, or any other problem specific parameter. To convert analog data to digital data, These are connected to analog to digital converter so that the analog data collected by them can be converted to the digital form which can be processed and manipulated by the controller and other components of sensor node.

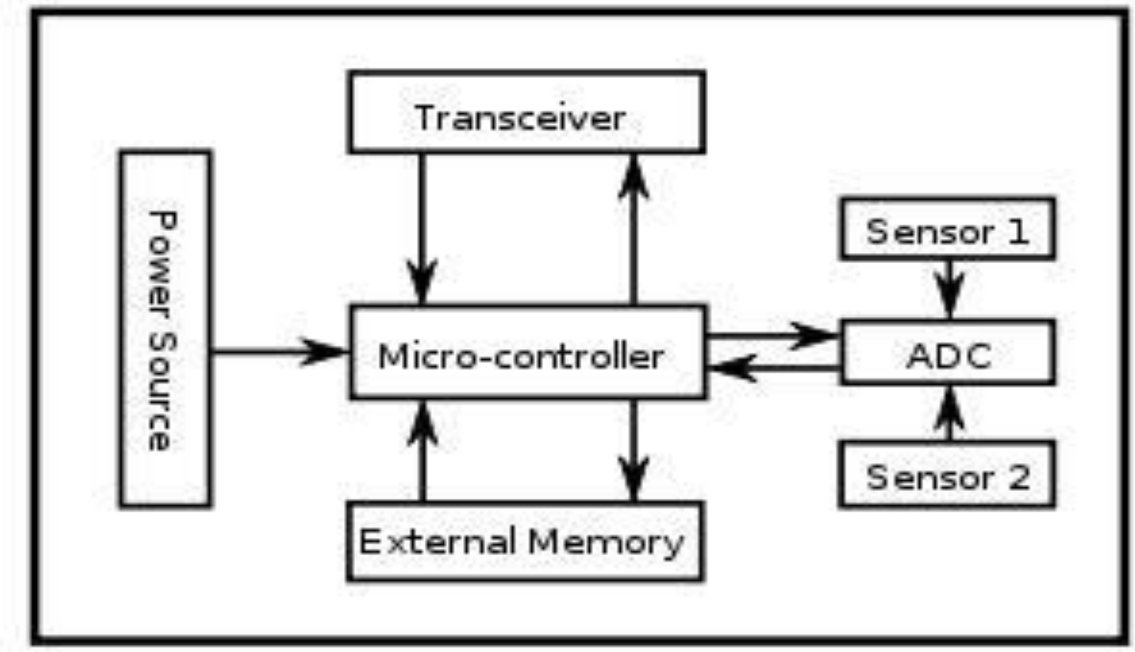

Fig.1.3 Architecture of Sensor Node

As shown above in the figure above sensor node inside structure which consists of all the five components discussed above, Micro-controller is considered as the brain of the sensor node as we can se all the other four components (sensors via Analog to Digital Converter) are connected to it. The other functionality along with these is of sensing environment, sensor node can act as a relay node in multi-hop architecture where sensor node just receives the data and forward it to base station.

1.2.2 Base Station

Base station is the component of Wireless Sensing Network responsible for collecting data from the sensor nodes. In some notation, it is also called sink node.. Along with collecting data it also posses the functionality of processing the data received by it an analyzing the collected data for capturing meaningful information from the data. On opposite to sensor nodes, base stations do have much better resources in respect of processing power, memory and energy. In some applications(where the base station is static) in fact base stations are directly connected to direct power supply rather than being battery powered if the base stations are static (i.e. not mobile). They also carry out the functionality of sending the processed data collected by sensor nodes to the remote server in some applications like border surveillance or if the network is established in some remote locations.

Base stations are also important component of Wireless Sensing Network in terms of energy utilization and reliability of the network. To save energy and have lasting network is the main requirement ,Since every sensor node in the network sends collected data to the base station and transmission of data wirelessly is the most energy consuming operation of a node, if we can somehow place the base station in position where energy required by all the nodes for transmission of data is minimum, then we can save a lot of energy and out network can last longer.

Every sensor node sends its data to the base station and if base station is malfunctioning or down then the whole network is down so along with energy utilization base stations are also important for proper conduction of Wireless Sensing Network which is not the case when a sensor node is down since we can still collect data from other live sensor nodes.

1.3 Motivation

Base station positioning problem in wireless sensor network is a very important problem because it affects the functionality of the whole sensor network. With the extensive use of wireless sensor network in real world now in form of different applications it is very important to work on the problems of WSN and make it more usable and reliable. WSN is used in some critical application areas like border surveillance where every aspect of the network is very critical because even a short glitch in network functionality can result in tremendous after effects.

Optimal base station positioning problem is a well known NP-complete problem [1]. Since it cannot be solved deterministically in polynomial time, we try to find the near optimal solution for it using different approximation techniques. The fact that we cannot solve it deterministically in polynomial time, makes us look for approximate solution for this problem since it is a readily occurring problem in WSN.

There are different population based optimization techniques applied on this problem like particle swarm optimization (PSO) [2] and genetic algorithm (GA) [3]. In recent time, a new swarm based optimization technique called the Artificial Bee Colony (ABC) proposed by Dervis Karaboga and Bahriye Akay [4]. They compared ABC on different benchmark functions in their comparative study and showed that for most benchmark functions, ABC performs better as compared to other well known population based techniques like PSO and GA.

1.4 Research Objective

As explained in the previous section of motivation for this work, following are the research objective of this work:

- To find the optimal position of the base station of the sensor network deployed. This is the basic core of our work
- In order to find the optimal location of the base station for the given sensor network, our criteria is the energy consumption by each sensor.
- So, our objective in this work is to maximize the minimum energy left with all sensor.
- We try to optimize our objective function using Artificial Bee Colony (ABC) and try to improve it via different modification over basic ABC.

1.5 Report Organization

We start this dissertation report with the introduction in chapter 1 describing wireless sensor network (WSN). In chapter 2, a detailed background of the base station positioning problem along with literature review is given. Chapter 3 explains about the proposed algorithms for finding optimized solution. Chapter 3 explains different modifications used to optimize the solution. Chapter 4 consists of performance evaluation and results of the proposed algorithms. We conclude about the word done and discuss the future prospect of work in this field in chapter 5.

CHAPTER 2

BACKGROUND AND RELATED WORK

In recent times, with increased use of wireless sensor network (WSN) in varied real life application areas, a lot of research is being done to optimize WSN and its various functionalities to make more dependable. By dependability we mean the responsiveness, lifetime, availability, security, etc. Since WSN finds its major application area as border surveillance, disaster management, and security surveillance which are highly critical applications areas, it is very important to make our WSN more dependable since no human assistance will be present in those areas. To improve the overall performance of WSN various researchers have worked on various functional and non functional requirements. A lot of work is done on different layers of communication protocol for WSN, routing of data packets, data aggregation, collision avoidance, data authentication and encryption, load balancing, query optimization, etc. with minimization of energy utilization as prime concern.

Along with above areas, a lot of work is also done to find the optimal position of base station in a WSN. The motivation for this research area is if we can place the base station at a optimal position with respect to all the sensor nodes in the network, then all the communication between sensor nodes and base station will in result be optimized. And since communication is the main energy consuming activity in sensor networks, placing the base station at best possible location will decrease the energy consumption which will result in increased network lifetime and dependability.

Base station positioning problem can be classified according to environment of WSN as follows [5]:

- Static base station positioning
- Dynamic base station positioning

2.1 Static Base Station Positioning

Considerable amount of work is done in the area of static base station positioning in recent times. Work of different researchers differs because of the assumptions they make for their network model, energy model, performance matrices, etc. We can classify static base station positioning problem according to the number of base stations to be positioned as follows:

- Single base station positioning
- Multiple base station positioning

2.1.1 Single Base Station Positioning

In this variant of static base station positioning problem our objective is to place a single base station for the whole sensor network. Approach to solve this problem differs according to the network structure assumed. If network architecture is flat, and sensor nodes are homogeneous i.e all sensor nodes have same amount of initial energy and other parameters as well, then it becomes a multi-hop data transfer problem which is a typical problem because in this, some nodes will act as relay node (node which receives data and transfers it further without sensing data in its own range) also.

The complexity of problem can be reduced if we assume that each sensor node can transmit data directly to the base station in a single-hop transfer mode. Since the base station problem is NP-complete, various approximation techniques are given by researchers. Along with approximation techniques, some computational geometry solutions are also given [6]. In this technique, we try to find the smallest circle which encloses all the sensor nodes of the network and then place the base station at the centre of that circle. The above technique is shown in the figure below.

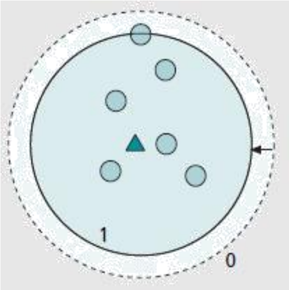

Fig.2.1 Base station positioning using computational geometry

As shown in the figure above, initially the dotted circle (0) is the solution containing all the sensor nodes represented by small circles, and after processing we were able to find a smaller circle (1) for which we place the base station at the centre shown as triangle.

2.1.2 Multiple Base Station Positioning

Multi base station positioning problem is much more difficult than single base station positioning problem. The problem can be defined as finding the optimal number of base stations to be placed along with their optimized location. If we somehow know the number of base station to be placed then its complexity reduces and the problem converges to single base station positioning problem.

Multiple base station positioning problem can be approached in one of the following ways:

- Approximation algorithms
- Integer programming

To solve this problem using approximation algorithms for increased data rate and network lifetime following techniques are used:

- Greedy algorithm
- local search strategy

In the greedy algorithm, the base station position is restricted to the location of sensors, and then the base stations are individually placed in an arbitrary order for increased data rate and network lifetime. In local search strategy, we start with some random solutions and then perform local search around those solutions to find the optimal solution.

In integer linear programming, the data routing load among the sensors is evenly distributed to minimize the maximum energy consumption individual sensors for minimizing the total communication cost.

2.2 Dynamic Base Station Positioning

All the techniques described in previous section considered only static network i.e. the sensor nodes does not move after the deployment of the network. So, before deploying the network we can calculate the optimal location for the base station and then place the base station along with the sensor nodes for which the optimal location is found out using the static base station positioning techniques.

But if we use the same static base station positioning techniques for dynamic network, where sensor nodes can move in the environment where they are placed, then in this scenario, when the sensor node will move, the optimal location of base station might change and the previous optimal location of base station might become obsolete. Therefore, dynamically repositioning the base station after particular amount of period according to the movement of the sensor node of the network can increase the performance of the sensor network.

Repositioning of base station according to network changes is very difficult to incorporate because for this we have continuously sense the network and its changes. But if we do incorporate dynamic repositioning of base station according to the network, then we can defiantly increase the following parameters of the network:

- Network lifetime
- Data delivery delay

If we do not reposition the base station then the network lifetime might decrease because the nodes which will be closed to base station will die faster as compared to nodes which are further away form the base station. Along with network lifetime, data delivery delay will also get hit due to the fact that data from nodes far away will take more time to get transmitted to the base station as compared to the nodes which are closer to base station.

Repositioning of base station is also required because of one more reason. When a network is established in a hostile environment like disaster management, if the base station is somehow gotten closer to the sensor nodes which are gathering information like fire, flood, or any other natural disaster, then the base station might get caught itself in the natural disaster.

2.3 Literature Review

In recent times, a lot of researchers have worked on this field of base station positioning problem because of the growing importance of wireless sensor networks. To make wireless sensor network more dependable and secure, researchers are working on lot of different aspects of a wireless sensor network. One important area of wireless sensor network that has come up in recent times is base station positioning. Researchers have tried varied techniques to solve different versions of this problem under different assumptions like population based algorithms, combinatorial geometric based solutions, neural network based solutions, etc. Rest of this section will include a brief description of various techniques used for base station positioning.

Authors Ivan Vilovic, Niksa Burum, and Zvonimir Sipus in [7] proposed a technique for solving base station positioning problem using a population based optimization technique called Ant Colony Optimization (ACO). ACO was proposed by Marco Dorigo and Gianni di Caro in [8], where they explained the functioning of ACO which is inherently used for discrete optimization problems. To make it applicable to continuous base station positioning problem, authors in [7] proposed to make pheromone matrix comprising of all the initial location of ants and their corresponding pheromone value. Ants then choose the path to be traversed by selecting a path using the pheromone probability equations of the ACO. This technique was compared by authors with PSO and GA to which the results showed that ACO takes more cycles as compared to others to converge to optimal solution.

Authors Yunyue Lin, QishiWu, Xiaoshan Cai, Xiaojiang Du, and Ki-Hyeon Kwon in [9] discussed strategies to place base station in two variants of the problems based on communication i.e. single-hop and multi-hop. For single-hop, authors proposed minimum enclosing circle (MEC) in which, first we randomly choose k sensor nodes as base stations out of total n. after selecting base stations, we cluster rest of the n-k nodes in clusters closest to them. After making clusters, we reposition the base station of that cluster to the centre of the MEC of that cluster. We continue this process till there are changes in previous step. For multi-hop, first select the base station position randomly as it was case in the previous single-hop technique. After choosing the base stations, we cluster rest of the sensor nodes to cluster by finding minimum energy path for every sensor node and then placing that node to the cluster where energy consumption is minimum.

Authors Andrej Bogdanov Elitza Maneva, Samantha Riesenfeld in [10] considerd the base station positioning problem with main concern on power consumption of the sensor network. They proposed two techniques for solving the base station positioning problem i.e. greedy algorithm and local search algorithm. The greedy algorithm is a deterministic algorithm which picks up base station locations one by one while keeping the previous base stations fixed to improve the rate of network. In local search algorithm, a set of initial solutions is randomly selected and then using local search technique, better solution in near vicinity of selected solution is searched. The number of iterations that the local search algorithm runs is predefined and result is published after those number of iterations are completed. Authors published that the greedy algorithm performs better as compared to local search algorithm by finding a solution closer to optimal solution.

Authors Tzung-Pei Hong, Guo-Neng Shiu in [2] proposed a technique for positioning multiple base stations in two tier wireless sensor network architecture using Particle Swarm Optimization (PSO). Constraint used by authors for positioning multiple base stations is power consumption. Authors in this paper assumed a heterogeneous two tier network in which each sensor node may own different initial energy and data rate. PSO is a population based optimization technique proposed by James Kennedy and Russell Eberhart in [11] for continuous optimization problems. Authors used PSO for solving base station positioning problem by selecting randomly a set of particles which represents the possible solution. In every iteration of the algorithm, position of particles is updated using the velocity associated with every particle. At the end of every cycle, velocity of every particle is updated using PSO equations and a global best solution is remembered. After completion of all iterations, this global best solution represents out solution. In result, the PSO technique works much faster and provides much better solution than the exhaustive search technique.

Authors P.D. Hossein Zadeh, C. Schlegel, M.H. MacGregor in [12] proposed a distributed technique for finding optimal location of the base station in dynamic wireless sensor network. In this work, authors provided power aware and network traffic aware positioning scheme for positioning of base station. Two techniques given in this paper are proposed based on weighted linear least squares and weighted non-linear least squares depending on the path loss exponent value which is a major characteristic in wireless communication.

Authors Rajiv Kr. Tripathi, Sateeshkrishna Dhuli2, Y. N. Singh, Nishchal K. Verma in [13] analyzed the effect of weighing factor on weighted centroid optimal base station positioning algorithm. The authors concluded that varying th weighing factor (alpha) do varies the energy consumption in wireless sensor networks,

In [14] Dorottya Vass , Attila Vidács proposed a technique for placing base station in a static wireless sensor network. In this work, base station is assumed to be dynamic i.e. movable and rest of the sensor nodes are static. So when one of the sensor node decides to transmit data, it can then send a request message to base station which move in the direction of the sensor node to an optimal position to gather data form the sensor node. They introduced three different mechanisms to optimize base station location. First, minimizing the average energy consumption. Second, minimizing the maximum energy consumption and third, minimizing the relative energy consumption.

In [15] Yong Seouk Choi, Kyung Soo Kim, Nam Kim proposed a evolution based optimization technique solution for solving base station positioning problem. The optimization technique used by authors to solve base station positioning is Genetic Algorithm (GA). GA was proposed by J.H.Holland and K.A.de Jong. In this work, author describe the representation of solution as location of base station along with number of base station to be positioned. Roulette wheel method is used for selection operation and the fitness function is multi-objective with coverage and power both as prime concern for optimization in this work.

In [16] Soo Kim, Jeong-Gil Ko, Jongwon Yoon, Heejo Lee presented a multi objective approach for solving base station positioning problem. In this, authors devised four different metrics as follows:

- Coverage of sensor nodes
- Fault tolerance of sensor network
- Energy consumption
- Congestion in network

Coverage of sensor nodes means number of nodes that can send data to the base station via single-hop or multi-hop. Fault tolerance means average number of nodes live after the failure of base station. Energy consumption converges to the distance between sensor nodes and the nearest base station. Congestion in network means average delay in network due to the traffic congestion. The authors used greedy search technique to implement these metrics.

In [17] Shashidhar Rao Gandham, Milind Dawande, Ravi Prakash, S. Venkatesan, proposed a model in which lifetime is spitted into equal periods of time and energy consumption is evaluated after every round. The authors used integer linear programming to solve find the base station position.

In [18] Sabbir Mahmud, Hui Wu, Jingling Xue solved the problem of placing multiple base station in wireless sensor network. The technique used by then includes dividing all the sensor nodes in k clusters with each cluster having a cluster head. Now, the problem is reduced to placing a base station optimally between these k cluster heads which will communicate with the base station.

In [19] M.Amac GUVENSAN, Z.Cihan TAYSI, A.Gokhan YAVUZ devised a new algorithm for placing multiple base stations in wireless sensor called k-means local+. Authors provided results in which they showed 45% better network lifetime than standard k-means algorithm. In k-means local+ algorithm first, the location of base station is found using standard k-means algorithm and then the base station is moved closer to their one-hop neighbor nodes with greater traffic load. Both of these algorithms i.e. k-means local+ and standard k-means are routing dependent. By routing dependent we mean, they evaluate the routing path also along with the position of the base station.

In [20] Anurag Singh Tomar, Gaurav Kumar Tak proposed a unique solution to black hole attack on a wireless sensor network using base station positioning. A wireless sensor network is under black hole attack when an intruder tries to capture some of the sensor nodes reprogram them to not send sensed data back to base station and even collect different type of data than it was previously programmed initially at the time of its deployment in the wireless sensor network. The solution proposed by author for black hole attack is to position multiple base station in the wireless sensor network using genetic algorithms (GA). In this work author published the results of black hole attack protection on the basis of data delivery rate of the proposed technique increased significantly but by positioning multiple base stations, transmission of same data multiple .times increase the energy consumption of the network significantly.

In [21] Yi Shi and Y. Thomas Hou presented two approximation algorithms for solving base station positioning problem. Authors considered flat sensor network architecture with possibility of multi-hop communication. In first algorithm, authors converted their continuous cost function to discrete form and then the continuous search space to finite number of sub areas. In second technique, authors exploited the cost property of each sub area and used this property to find the optimal position of the base station.

CHAPTER 3

PROPOSED WORK

3.1 Problem Statement

In the proposed thesis, let we have 'n' sensor nodes in a sensor network where initial energy is represented by E(i) and data rate is represented by D(i) for a sensor node, for i= 1,2,3…n. Here, objective is to find a location of the base station for which the „lifetime maximization " of the sensor network can be achieved.

The amount of time the network is alive after its initial deployment represents the lifetime of a wireless sensor network i.e. all the required operations performed successfully by all the sensor nodes of the network. The network is considered to be non-functional at the moment when any one of the „n" sensor nodes of the network dies.

To calculate the lifetime of wireless sensor network, we calculate the lifetime of each individual sensor for a particular location of base station and then find minimum of all of them, which will tell us the time when the first sensor node of the network will die.

Our objective is to lifetime maximization of the network which in turn is the minimum lifetime of all the sensor nodes in the network. Following assumptions are taken into account:

- The system is heterogeneous i.e. initial energy and data rate of sensors may vary from node to node.
- Every sensor node in the network are at the same level in network hierarchy i.e. flat network architecture.
- Single-hop communication i.e. all the sensor nodes can transmit their data to the base station in single hop only.

- Irrespective of the location of sensor node(i) and base station(x(i)), the base station(x(i)) is always in transmission range of sensor node(i) for i = 1,2,3….n i.e. transmission range.

3.1 Energy Model

The energy model used for a sensor network, to transmit a composite bit stream at data rate ‘r’ over the Euclidean distance ‘d’, is represented in the equations below which was adopted by [22] to calculate the energy consumption(p) per unit time is:

$$p(r,d) = r(\alpha_1 + \alpha_2 d^n) \qquad (1)$$

where, r represents data rate transmission

d represents Euclidean distance

α_1 represents distance independent parameter

α_2 represents distance dependent

n represents path loss exponent

p(r,d) represents the energy consumed per unit time for node with ‘r’ data transmission rate and ‘d’ Euclidean distance between receiver and sender. [22] Studied that for same transmission rate i.e. homogeneous system, our problem only remains to finding the minimum enclosing circle containing of the wireless sensor network consisting all the sensor nodes.

We are considering heterogeneous system, therefore, initial energy and data rate for i^{th} sensor node is represented as:

- “$e_i(0)$” represents initial energy of i^{th} sensor node.
- “r_i” represents data transmission rate of i^{th} sensor node.

The lifetime of i^{th} sensor node for the j^{th} location of the base station is calculated as follows [2]:

$$\text{lifetime}_{ij} = e_i(0) / r_i (\alpha_{i1} + \alpha_{i2} d_{ij}^{\,n}) \qquad (2)$$

To find out the lifetime of wireless sensor network we have to find out the minimum lifetime value amongst all sensor nodes as from the formula:

$$\text{lifetime}_{wsn} = \text{Min} \{ \text{lifetime}_{ij} \} \qquad (3)$$

Where i represents i^{th} sensor node for i = 1,2,3….m.

So, our objective function can be represented as follows:

$$\text{Fitness} = \text{Max} \{ \text{lifetime}_{wsn} \} \qquad (4)$$

3.3 Artificial Bee Colony(ABC) Approach

As you might have gathered from the previous chapters that node deployment in sensor networks still remains one of those open problems that have fascinated researchers. The previous chapter covers some of the approaches suggested by researchers for tackling the problem. One of the recent swarm based approaches that has provoked the interest of many scientists for solving complex combinatorial problems is the Artificial Bee Colony algorithm. The emerging importance of wireless sensor networks in real world applications induces us to solve this problem somehow have to a satisfactory level. To achieve this, we propose Artificial Bee Colony (ABC) approach. ABC was proposed by Dervis Karaboga and Bahriye Akay [4]. ABC is a population based optimization technique inherently used for optimizing continues functions. As our objective function is also continues problem and base station positioning problem is NP-complete irrespective of whether it is a single base station problem or multiple base station, this technique is a perfect fit for our problem.

In [4] authors also described the applicability of ABC by comparing it with other readily optimization techniques like Genetic Algorithm and Particle Swarm Optimization over standard mathematical benchmark functions. The comparison results showed that for most of the benchmark functions ABC performs better than PSO and GA. Although this comparison does not proves that ABC is better than PSO or GA for every problem, but it is a very good metric to start with for the given problem.

ABC is an optimization technique based on the natural behavior of bees in the process of gathering nectar from food sources. According to the behavior of bees, they are classified three categories:

- employee bee
- onlooker bee
- scout bee

Each bee performs a specific task in nature. In ABC algorithm, we simulate all these tasks in a computerized way. Employee starts the process of collecting nectars from different food sources. They bring back their nectars to the hive(base) and exchange information about the food source they visited by a particular dance called waggle dance. Physical interpretation of waggle dance is, more a bee dances higher the quality of the food source visited by it. After seeing the waggle dance, onlooker bee decides the food source they are going to visit and this process continues until a food source is dead. When a food source is dead, it is abandoned, and scout bee starts searching for a new food source to replace the abandoned one.

To imitate the above functionality in a computer program, we divide it in three phases namely employee bee phase, onlooker bee phase, and scout bee phase. In employee bee phase, each employee bee goes to the different food sources and perform a local search around the selected food source and evaluate the objective function value of the nearby solution. If the nearby solution is better than the previous one then we replace the previous solution by the new solution and store the objective function value of the new food source and also initialize the trials count of that food source to 0. But if the previous solution is better than the new solution than we increment the trials count of that food source by 1.

On the basis of the objective function values, onlooker bee finds probability of selection of each food source and uses it to select food source to visit. The onlooker bee also performs local search around the selected food source and tries to find a better solution. If it is successful in finding a better food source around previous one, then it replaces it and set the number of trials of that food source to 0, otherwise increment the trials of that food source by 1.

After onlooker bee, in scout bee phase, we check the trials count of every food source and find the food source with maximum trials value. If the maximum trial value found is greater than the predefined „limit‟ then, scout bee abandons that food source and replace it with new one and initialize its trials count to 0.

After the scout bee phase, we find out the best food source among all according to the objective function value i.e. fitness value. We repeat about process „maxcycle‟ times and returns the best food source found after maxcycle times. The above process represented in pseudo code form as follows:

```
Initialization();
find_best_food_source();
While(maxcycle)
{
        employee_bee_phase();
        onlooker_bee_phase ();
        scout_bee_phase ();
        find_best_food_source();
        maxcycle--;
}
return best_food_source; //food source with maximum fitness
```

As shown in the pseudo code above, before employee bee phase for the first time we initialize the food sources. The number of food sources to be employed is fixed by the authors as follows:

$$\text{Colony size} = \text{Number of employee bees} + \text{Number of onlooker bees} \qquad (5)$$

And along with this, the number of scout bees is fixed as 1 by author. Although number of scout bees can be changed according to the problem domain.

The local search performed by employee bee and onlooker bees follows the following equation:

$$v_{ij} = x_{ij} + \phi_{ij} (x_{ij} - x_{kj}) \qquad (6)$$

where, x_{ij} represents the jth parameter of ith food source

x_{kj} represents the jth parameter of kth food source

ϕ_{ij} represents a random number in range [-1,1]

v_{ij} represents jth parameter of the newly formulated food source

the parameter „limit" of ABC is a very important factor. The value of limit defines the exploration and exploitation capabilities of ABC. If we keep a high value of limit, then more exploitation will happen and algorithm might get caught up local optimal solution. If we keep limit value low, then more exploration will happen and algorithm might not be able to exploit the food sources completely. So, in order to find a optimal value of limit, we use the following formula:

$$\text{limit} = (\text{ colony size } / 2) * D \qquad (7)$$

where, D is the number of parameters in the problem.

Now, in our problem to initialize the food sources, we are randomly generating food sources using a random number generator in the specified range. The number of parameters in our problem of base station positioning problem is 2 since we are working in a 2-dimensional area only two coordinates of the location of base station is our parameters. After generating initial food sources and employee bee phase, onlooker bees calculate the probability of selection of food source by the following formulae:

$$\text{probability}_i = ((0.9 * \text{fitness}_i) / \text{best\_fitness}) + 0.1 \qquad (8)$$

where, best_fitness represent fitness of the best solution.

3.4 Grenade Explosion Method and Cauchy Operator

To improve the performance of ABC, a novel ABC combined with grenade explosion method (GEM) and Cauchy operator, namely, ABCGC, is proposed. GEM is embedded in the onlooker bees' phase to enhance the exploitation ability and accelerate convergence of ABCGC; meanwhile, Cauchy operator is introduced into the scout bees' phase to help ABCGC escape from local optimum and further enhance its exploration ability.

Unfortunately, ABC is good at exploration but poor at exploitation and its convergence speed is also an issue in some cases, so a lot of its variants have been proposed in recent years to further improve the performance of ABC.

In the below equation for finding a new source while performing a local search

$$xij = x\text{min } j + \text{rand } (0,\ 1)\ (x\text{max } j - x\text{min } j)\ ,$$

j is a crucial parameter since it directly influences the position of a new food source. However, the randomly chosen dimension j may not always guide ABC toward more high fitted positions of food sources and lead to slow convergence or even make the search easily trapped in local optimum.

GEM first presented by Ahrari et al. [23] in 2009 is inspired by the mechanism of a Grenade explosion, where objects are hit by pieces of shrapnel. Damage caused by each piece of shrapnel hitting an object is calculated. A high value for damage per piece in an area indicates there are valuable objects in that area. To intensify the damage, the next grenade is thrown where the greatest damage occurs. This process will result in finding the best place for throwing the grenade. GEM is introduced into the onlooker bees phase of ABC to select the optimal search dimension instead of a random chosen one for each onlooker bee in hope that they collectively move towards the optimal position. Here, the overall damage caused by the hit is considered as the "fitness" of a solution. Note that the number of pieces of shrapnel per grenade should be large enough so that far regions can be explored for new high fitted regions and the algorithm would not be trapped in local optimum. To eliminate the need for setting the parameters of GEM, there is only one grenade and let the grenade throw D pieces of shrapnel in each cycle.

In each cycle of ABCGC, D pieces of shrapnel are thrown in all the dimensions (i.e., each dimension is exploited by only one shrapnel) to gather information around the current position of the grenade (old food source); meanwhile, each onlooker bee computes each candidate food source along which each shrapnel is thrown and evaluates corresponding damage per shrapnel value (fitness) and then makes a decision on a new candidate food source with the greatest damage (the highest fitness), which means the selected optimal search dimension is biased towards the global or near-global optimal position more quickly. Consequently, in ABCGC, a new candidate solution based on the optimal search dimension for an onlooker bee is produced by

$$\mathrm{V}iOSD = xiOSD + \phi iOSD\ (xiOSD - xkOSD)$$

$$\text{s.t. fit}\ (ViOSD) = \max\ \{\text{fit}\ (Vit) \mid t = 1, 2, \ldots, D\}\ , \qquad (9)$$

where $k \in \{1, 2, \ldots, SN\}$ is a randomly chosen index and $k \neq i$; $OSD \in \{1, 2, \ldots, D\}$ represents the optimal search dimension; $\phi iOSD$ is a random number in the range from -1 to 1; Vit denotes a new candidate food source Vi generated by just changing the value of old food source Xi in dimension t, namely, $\mathrm{V}it \neq xit$, while the rest of Vit keep the same value as Xit; $ViOSD$ has a similar meaning as Vit and also indicates that Vi obtains the maximum fitness in dimension OSD instead of other dimensions. Similarly, after an onlooker bee determines a new candidate food source in the neighborhood of its currently associated food source using (9) and (4), a greedy selection mechanism is applied between the new food source and the old one.

New Exploration Strategy Adopted by Scout Bees:

Although a search can easily fall into local optimum, Cauchy operator ensures that the search is executed in the global region and does not trap in local optimum prematurely. Their results confirm that Cauchy operator is appropriate for global search due to its higher probability of making longer jumps.

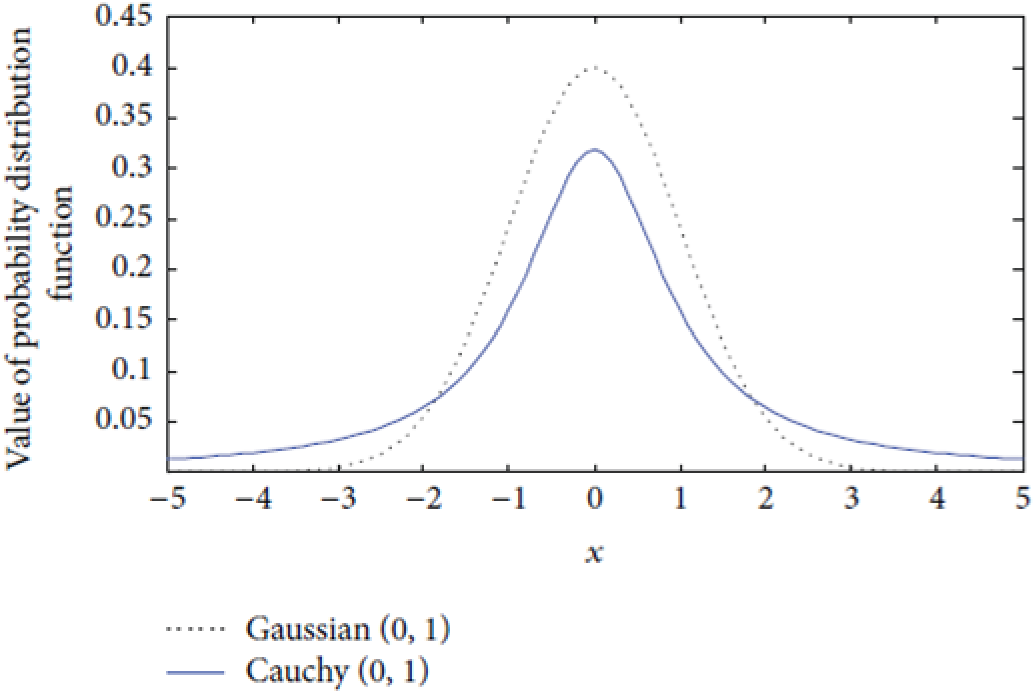

Fig.3.1 Probability density function of standard Cauchy and Gaussian distributions.

In Fig.3.1, on the interval [−3, 3], Gaussian function has a large probability, but its probability is almost 0 on the intervals (−∞, −3) and (3,∞). Cauchy distribution has a similar shape as Gaussian distribution, but it has more probability in its long tail area, so the possibility of Cauchy distribution to generate a random number away from the origin is higher than Gaussian distribution. In other words, Cauchy distribution-based random numbers explore a relatively wider search space than Gaussian distribution based random numbers. Thereby, Cauchy operator is introduced into the scout bees' phase of ABC to generate a wider solution instead of a random produced one for each scout bee. This will help ABC escape from local optimum and further enhance the ability of global exploration. In ABCGC, if the abandoned food source is X_i , a scout bee produces a new food source according to

$$xij = xij\text{Cauchy }(0, 1) \qquad (10)$$

where Cauchy(0, 1) is the standard Cauchy distribution, which denotes a random value from a Cauchy distribution centered at 0 with a scale parameter equal to 1. Concretely, Cauchy(0, 1) is defined as

$$\text{Cauchy}(0,1) = \frac{1}{\pi\left(1+x_{ij}^2\right)}. \qquad (11)$$

3.5 A Different ABC Approach

Two modifications were made to the conventional ABC algorithm. The first modification was how the EBs conduct local search. The second modification is how scouts are defined and when they move to new locations. Both modifications are described below.

In the conventional ABC the EBs conduct local search via Eq. (6). This perturbation method was replaced with a perturbation method typically found in differential evolution algorithms. As before, only one dimension is perturbed. The new perturbation method is

$$V_i' = V_j + F.(V_k - V_l) \qquad (12)$$

where j, k and l are randomly chosen subject to j != k != l != i.
F = 0.8 is a fixed constant.

Each parameter is constrained to the unit interval and the perturbation method may produce values outside this range. In that case the component is set to the boundary value. We note that the OBs still search using Eq. (6). The second method modifies the scout phase. In the conventional ABC algorithm a scout is created only after an EB has tried to find a better solution for limit iterations without success. In our modified algorithm that method is still used. However, in addition every infeasible EB automatically becomes a scout bee. [24]

Find_new_position(minfit_foodsource); (13)

This approach creates scouts far more often found creating more scouts by this method improved the performance of ABC algorithms on standard engineering constrained problems. The worst fit feasible EB also becomes a scout. This method produces far more scouts than does the conventional ABC algorithm.

The pseudo code for our proposed Scout Bee phase is shown below.

```
SendScoutBees () {
        for (all Food Sources) {
            if (trial[Food Source] >= limit)
                Find_new_position(Food Source);
            if (fitness[minfit] > fitness[Food Source])
                minfit_foodsource = Food Source;
        }
        Find_new_position(minfit_foodsource);
    }
```

3.6 Chaotic ABC Approach

The Chaotic Local Search method is applied to solve the accuracy problem of global optimal value. The premature convergence issue of the Artificial Bee Colony algorithm has been improved by increasing the number of scout and rational using of the global optimal value and Chaotic Search.[25]

In the basic Artificial Bee Colony algorithm, the best solution founded by onlooker bee which adopted the local search strategy is unable to reach the ideal level of accuracy. In order to improve the accuracy of optimal solution and obtain the fine convergence ability, we use the chaotic search method to solve this problem. In the Chaotic Search ABC algorithm (the improved ABC is also called CABC), onlooker bees apply chaotic sequence to enhance the local searching behavior and avoid being trapped into local optimum. In onlooker bee phase, chaotic sequence is mapped into the food source. Onlooker bees make a decision between the old food source and the new food source according to a greedy selection strategy.

The well-known logistic map which exhibits the sensitive dependence on initial conditions is employed to generate the chaotic sequence [25]. The chaos system used in this paper is defined by

$$x_{i+1} = \mu * x_i * (1 - x_i) \tag{14}$$

Where μ is a chaotic attractor,

If μ equal to 4 then the above system enters into a fully chaos state, $x_i +1$ is the value of the variable x_i in i iteration. After the chaotic sequence generated, the new food source will be obtained by equation (15)

$$x = x_{mi} + R * (2 * x_i - 1) \tag{15}$$

Where x is the new food source and x_i is the chaotic variable, R is the radius of new food source being generated. The food source x_{mi} is in the central of searching region. After the food source has been generated, onlooker bee will exploit the new food source and select the higher profitable one using a greedy selection.

Chaotic search method includes the following steps:

Step1. Setting the iterations (cycle parameter) of chaotic search and produce a vector *x0* = [*x0,1 x0,2,x0,3* …], which is the initial value of chaotic search;

Step2. The chaotic sequence is generated according to expression (14) and a new food source, which combining the chaotic sequence with the original food source, is obtained following the equation (15);

Step3. Calculating the profitability of the new food source and using the greedy selection select the higher profitability food source;

Step4. If the number of chaotic searches iterations greater than maximum, the artificial bee algorithm will enter the scout bee phase, or else enter the next chaotic search iteration.

Global Search Strategy

When a scout bee find the food source unchanged times greater than the limit parameter, it will produce a new food source and replace the original one .Scout bee discover the new food source using the best optimal value strategy which accelerate the global convergence rate.

Assume that the solution *xi* has been abandoned and the scout bee will generate the new solution x_m using the following equation

$$x_m = x_{best} \qquad (16)$$

$$x_m(i) = x_{best}(i) + \phi_{mi} * (x_{best}(i) - x_{neighbor}(i)) \qquad (17)$$

Where $x_m(i)$ is new food source produced by scout bee using the global optimal value $x_{best}(i)$ and φ_{mi} is a random number within the range [-1,1].

3.7 Dynamic Regulatory Factor GBest ABC Approach

Gbest-guided ABC algorithm (GABC) to address the upper problems adopts the formula (18) to replace the formula (6) in the original ABC algorithm [27].

$$V_{ij} = x_{ij} + \phi_{ij} * (x_{ij} - x_{kj}) + \psi_{ij} * (y_j - x_{ij}) \qquad (18)$$

In which, φ_{ij} is a random number in [-1, 1], ψ_{ij} is a random number, y_j is the optimal solution fitness value in the j-dimensional space.

Dynamic regulatory factor

DRF-GABC algorithm based on the dynamic regulatory factor. The search capability of algorithm could be dynamically controlled by adjusting dynamic regulatory factor “r” and “k”.

In which, the global dynamic regulatory factor is

$$r = 1 + 0.5\sin(\pi\, i \,/\, N)\ ,$$

the local dynamic regulatory factor is

$$k = 1/\,3 + 2\,/\,3\sin(\pi\, i \,/\, N)\ ,$$

DRF-GABC Algorithm

In the searching process, in order to further control the degree of information sharing between bee individuals and the global search precision of algorithm, we introduced two dynamic regulatory factors in this paper, namely local dynamic regulatory factor *k* and global dynamic regulatory factor *r* respectively. The core formula based on dynamic regulatory factor of GABC algorithm is described in formula (4).

$$v_{ij} = x_{ij} + r*\varphi_{ij}*(x_{ij} - x_{kj}) + k*\phi_{ij}*(y_j - x_{ij}) \qquad (19)$$

r and k are dynamic regulatory factor

i is the current iteration number,

φ_{ij} is a random number in[-1,1],

ψ_{ij} is a random number in [0, C] (C is a positive constant, the default value is 1.5),

y_j is the optimal solution fitness in the *j*-dimensional space.

In the early stage optimization of the algorithm, the local regulator factor *k* should larger than global regulator factor *r*, which means the algorithm is mainly focusing to improve the local search ability. But as the iteration number increase, the local search ability of algorithm should be decreased. Thus, in the latter stage of the algorithm, as the iteration number increases, the global search capacity of algorithm will be enhanced.

3.8 Novel ABC Approach

Both exploitation and exploration are very important for the population-based optimization algorithms. In these optimization algorithms, the ability of the exploitation is to find the global best value while the ability of exploration is to look for better solutions in numerous unknown sector of the solution space. However the exploration and exploitation contradicts to each other in practice. In order to achieve good optimization performance, the two abilities should be well balanced. According to the search equation of ABC algorithm which is described by Eq. (6), the new candidate solution is generated by moving the old solution towards another solution selected randomly from the population. But, as the probability that generate a good solution is similar to generate a bad solution, so the new candidate solution is not definitely better than the previous one. What is more, the ability of exploration in Eq. (6) is random enough. All in all, the search equation of ABC is good at exploration but poor at exploitation. In order to overcome the limitation, we[28] modify the solution search equation by incorporating best solution into the search process. The operation process as follows

$$v_{ij} = x_{ij} + (1-\varphi)(x_{kj} - x_{ij}) + \varphi(Gbest_j - x_{ij}) \qquad (20)$$

where $\varphi \in (0, 1)$ and $Gbest_j$ is the best solution in the current population, the best solution in the current population can lead the new candidate solution to the best, namely, improve the exploitation performance of ABC algorithm. Here, we modify ABC algorithm by using Eq. (20) replaced Eq. (6). We refer to this modified ABC algorithm as NABC.

CHAPTER 4

RESULTS AND ANALYSIS

4.1 Experimental Setup

In this chapter, we provide the performance evaluation of our proposed work to compare it with the basic ABC algorithm. The proposed algorithms ABCGC(Grenade Explosion Method and Cauchy operator based ABC), Different ABC, CABC(Chaotic search based ABC), DRF_GBest ABC(Dynamic Regulatory Factor GBest guided ABC) and NABC(Novel ABC) are implemented in Java, SE7 language using Eclipse(version Kepler) IDE. These implementations are run on a system with Intel(R) Core(TM) i5-4210U CPU@1.70GHz 2.40GHz processor, 4.0Gb RAM, and windows 10, 64-bit operating system.

This chapter consists of 6 sub-sections; in which each proposed algorithm is compared with the basic ABC algorithm to analyze is performance. Each sub-section consists of performance evaluation of a proposed algorithm with respect to basic ABC algorithm on a parameter called MaxCycle.

All the algorithms are run 30 times, with different MaxCycle values, and the mean of all the 30 runs is taken to finally conclude the performance of that particular algorithm. To find the base station of a wireless sensor network, we need a sensor deployment and to generate that, we used a random function to initially generate the positions of the sensors for which we have to find the base station. The static deployment of the sensor nodes is shown in the figure below.

Every dot in the figure shown below represents a sensor node. The solution space taken for experimental result is [0,100] for both the coordinates i.e. x and y coordinate and there are total of 50 sensors. All the algorithms are compared on this static deployment only.

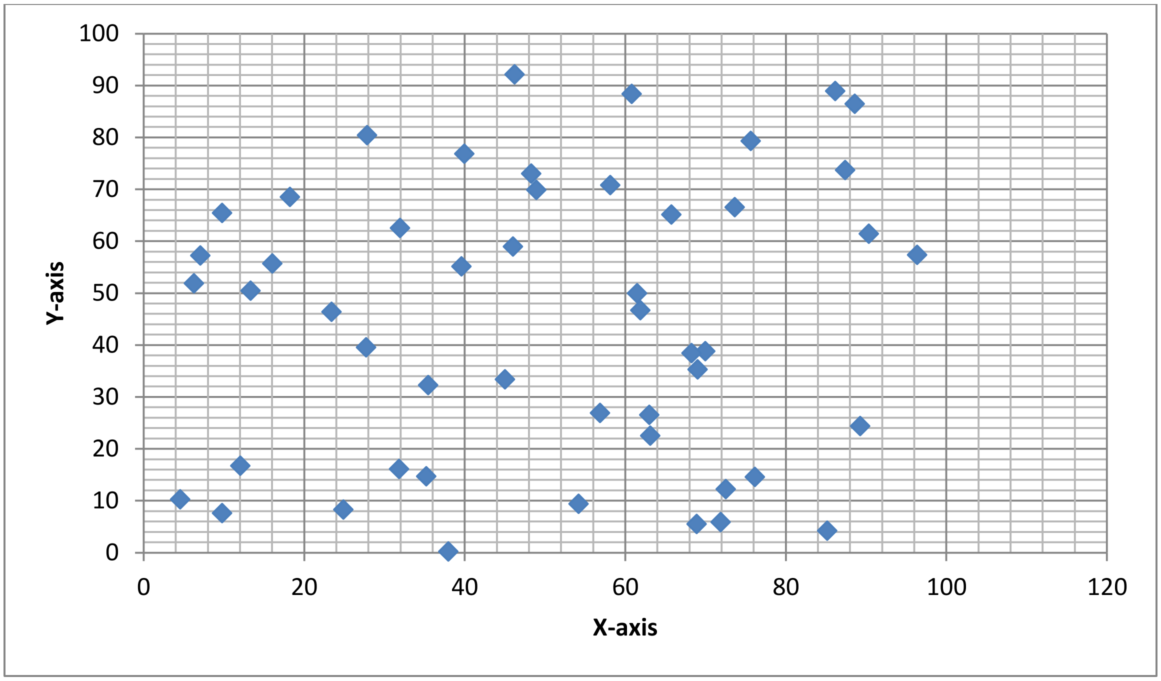

Fig.4.1 Deployment of Sensor Nodes

Along with the static deployment, initial energy and data rate of the static sensors is also generated using random number generator. The range for initial energy is 10000000 to 99999999 and range for data rate is 1 to 10. the relation between data rate and initial energy is maintained so that, if the data rate of sensor $node_i$ is greater than data rate of sensor $node_j$ then, initial energy allocated to sensor $node_i$ will also be greater than the initial energy allocated to sensor $node_j$.

The colony size taken for all the proposed algorithms are 10 i.e. 5 employee bees, 5 onlooker bees and 1 scout bee and the limit for ABC is set to 10 according to the equation (7). The parameters α_1 representing distance independent parameter, α_2 representing the distance dependent parameter in lifetime evaluation equation (2) are taken as 0 and 1.

Along with this, the path loss exponent „n‟ in equation (2) is taken as 2 for all computations. The performance of all the proposed algorithms are represented for the following values of MaxCycle i.e. 1000, 1500, 2000, 2500, and 3000 as shown in the figures in subsequent sections. For any value of MaxCycle we took value after every 50 cycles and took mean of it for every run.

In this section, we present the performance evaluation of all the algorithms and comparing it with performance of basic ABC algorithm. The results are shown for different value of our performance metric i.e. MaxCycle as follows:

4.2 For maxcyclenumber= 1000

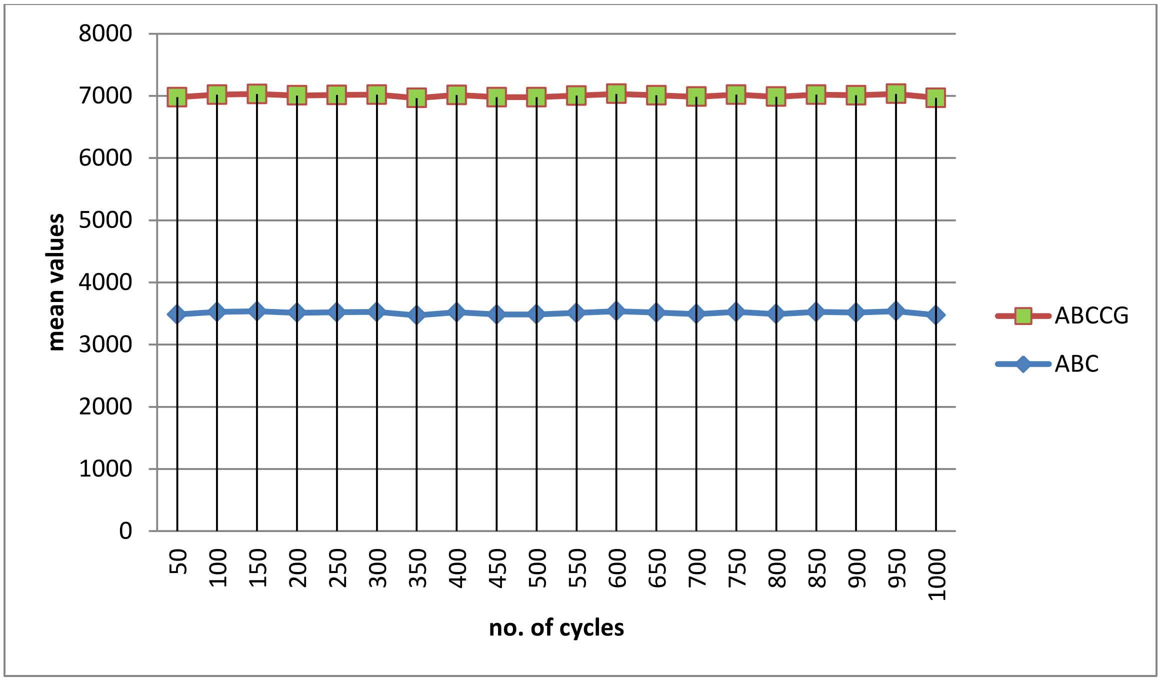

Fig.4.2 Performance Evaluation-ABCGC

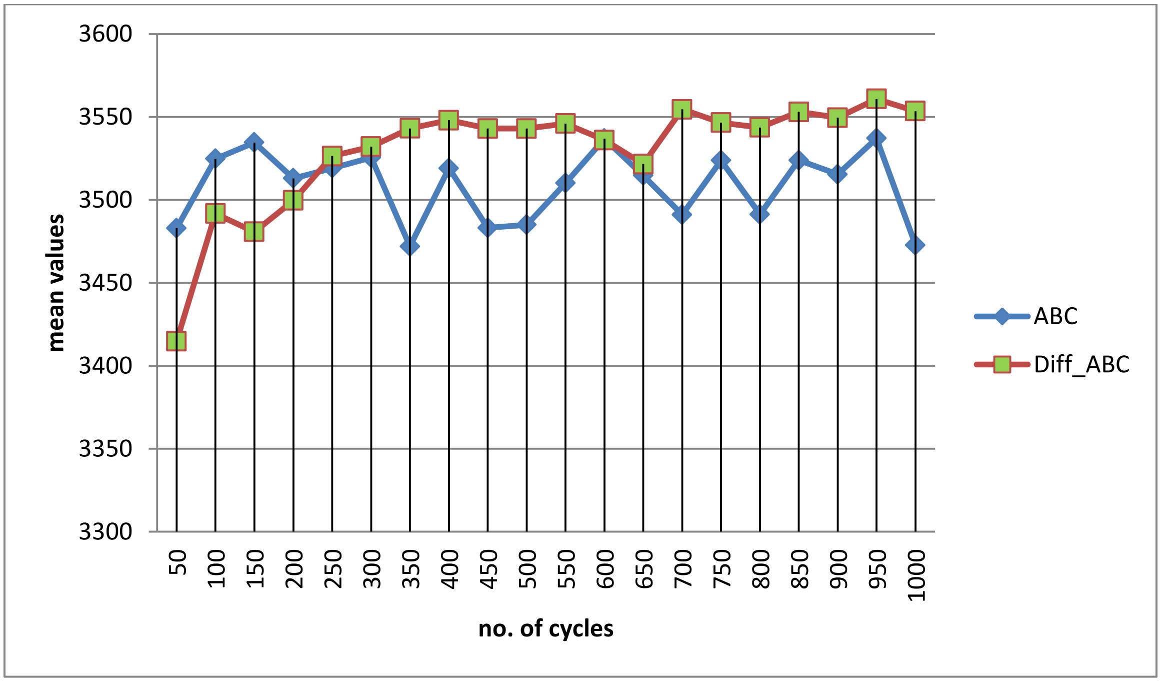

Fig.4.3 Performance Evaluation- Different ABC

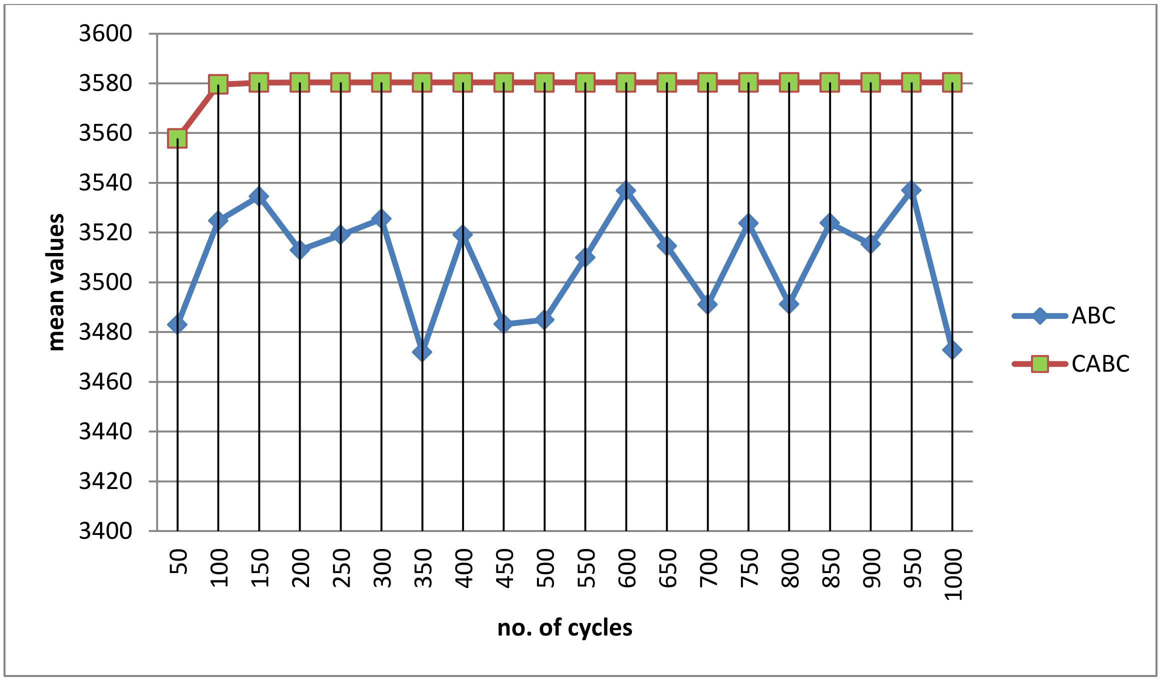

Fig.4.4 Performance Evaluation- CABC

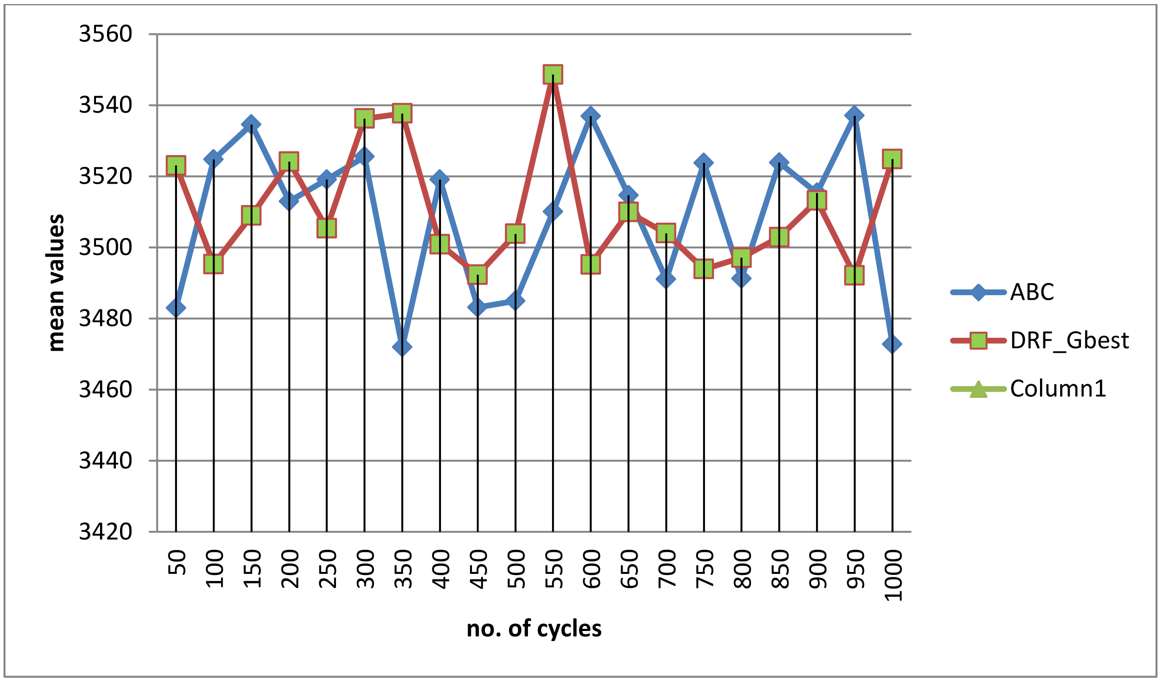

Fig.4.5 Performance Evaluation- DRF_GBest

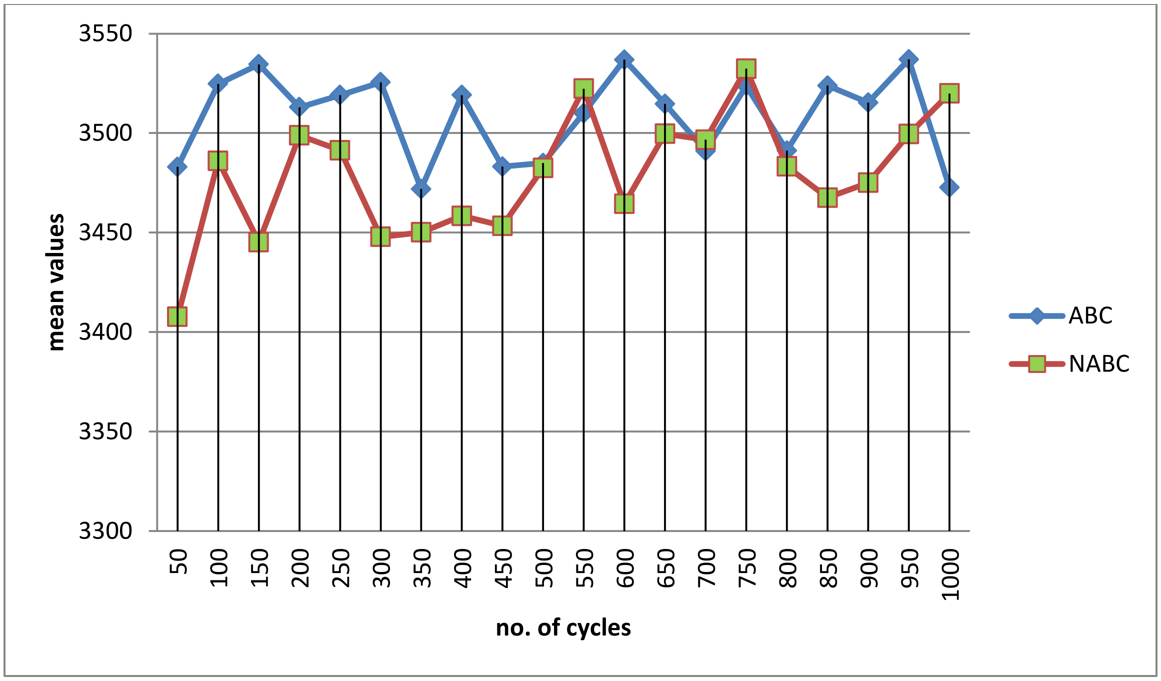

Fig.4.6 Performance Evaluation- NABC

As shown in the above figures4.2-4.6, we took the value of lifetime after every 50 cycles for every run and took mean of it which represented in these figures which explains that irrespective of the Maxcycle value, all algorithms are performing better than ABC for maxcyclenumber=1000.

4.3 For Maxcyclenumber=1500

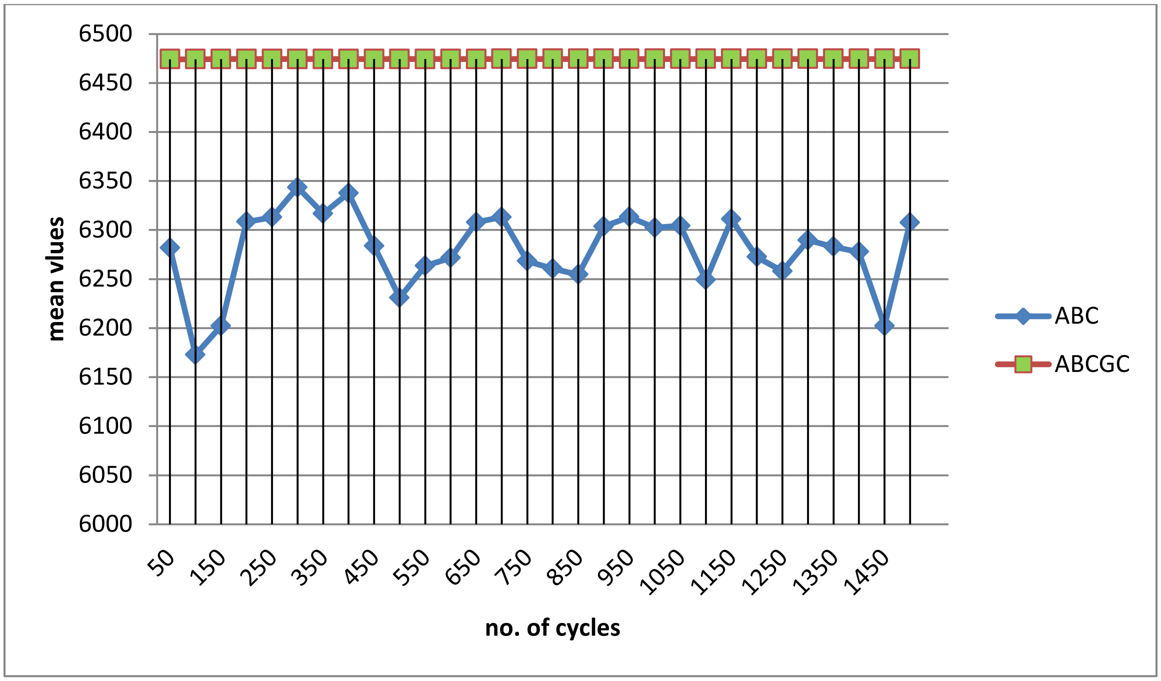

Fig.4.7 Performance Evaluation-ABCGC

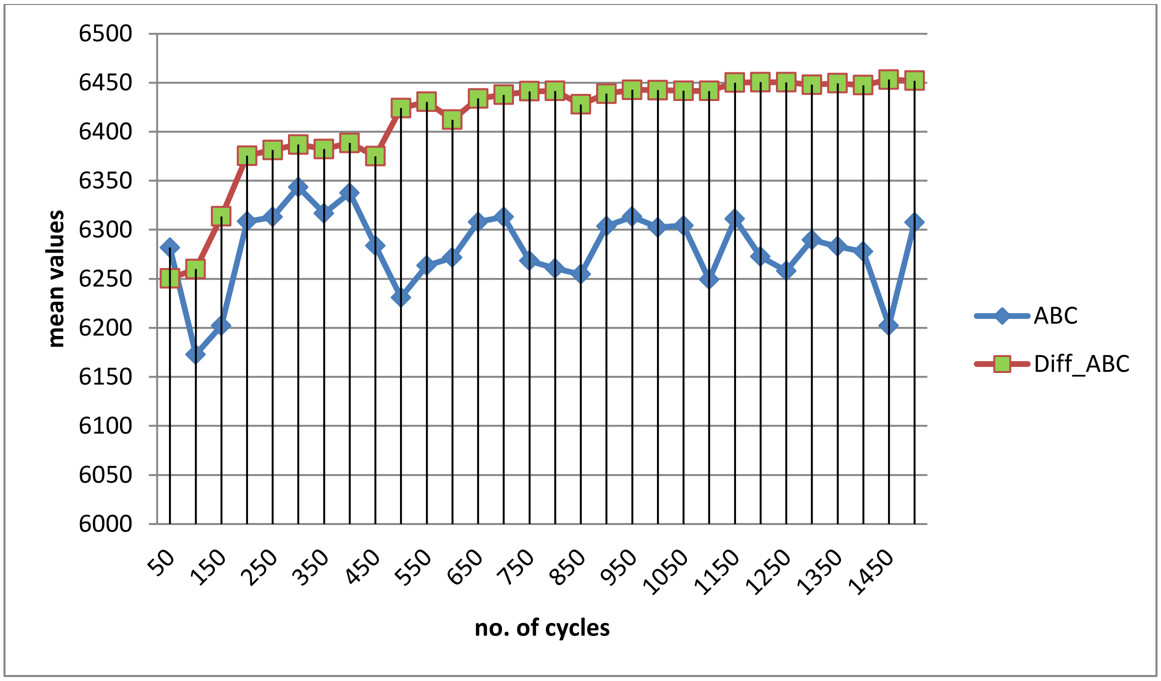

Fig.4.8 Performance Evaluation- Different ABC

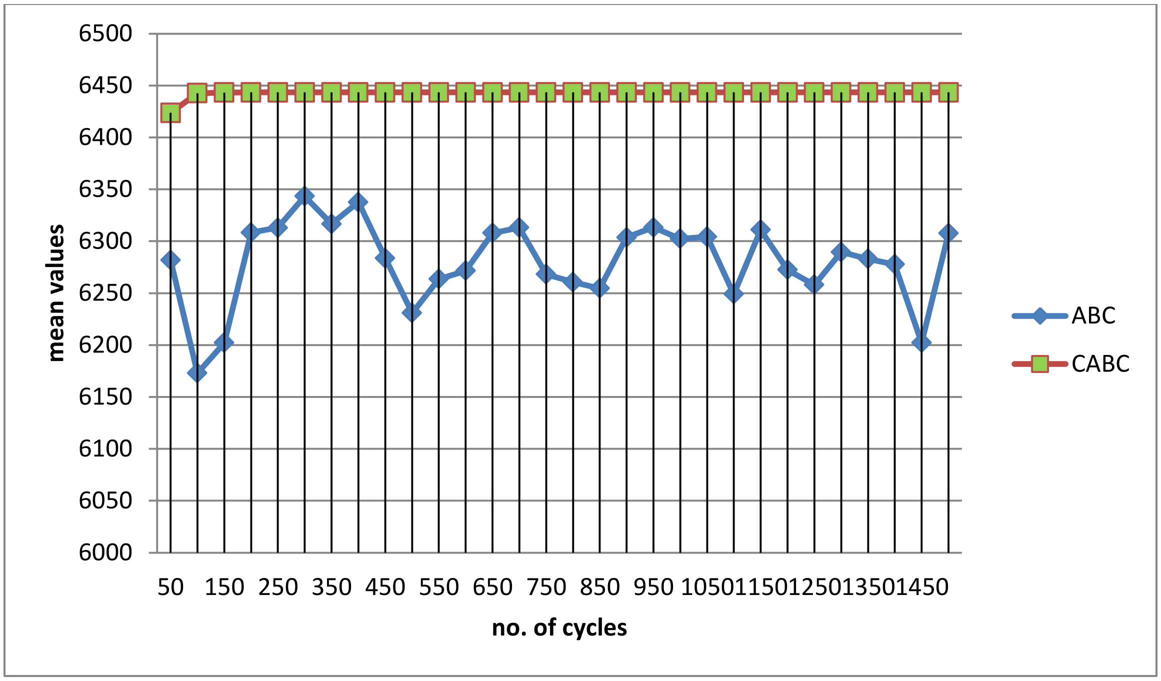

Fig.4.9 Performance Evaluation- CABC

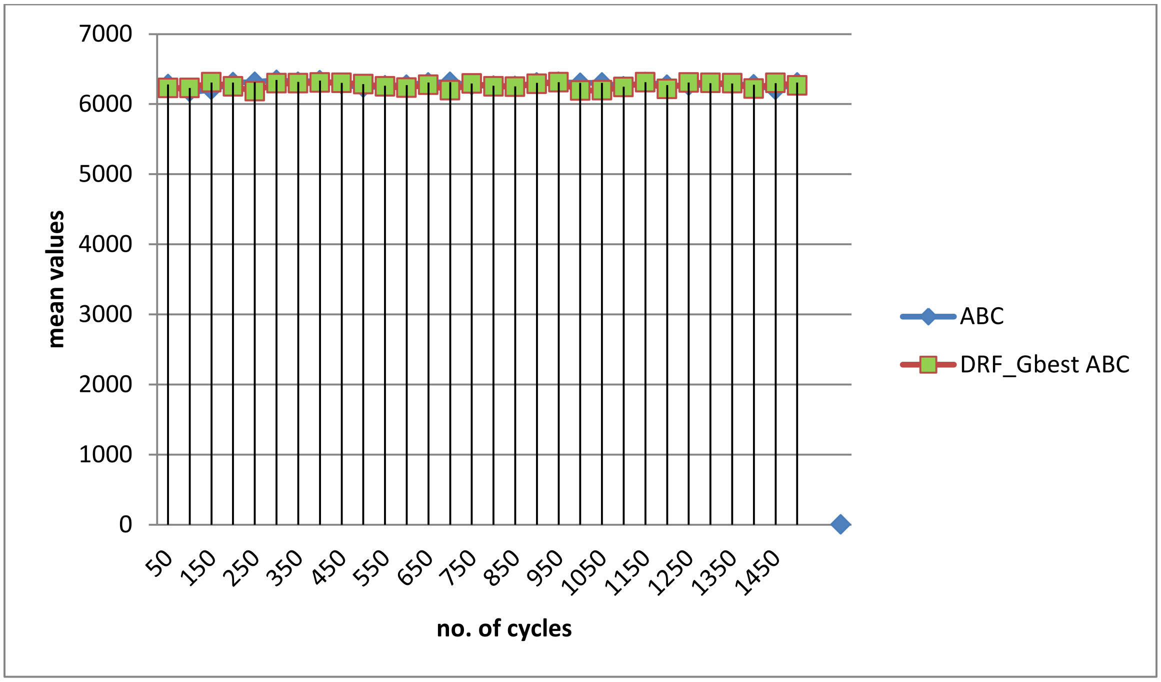

Fig.4.10 Performance Evaluation- DRF_GBest ABC

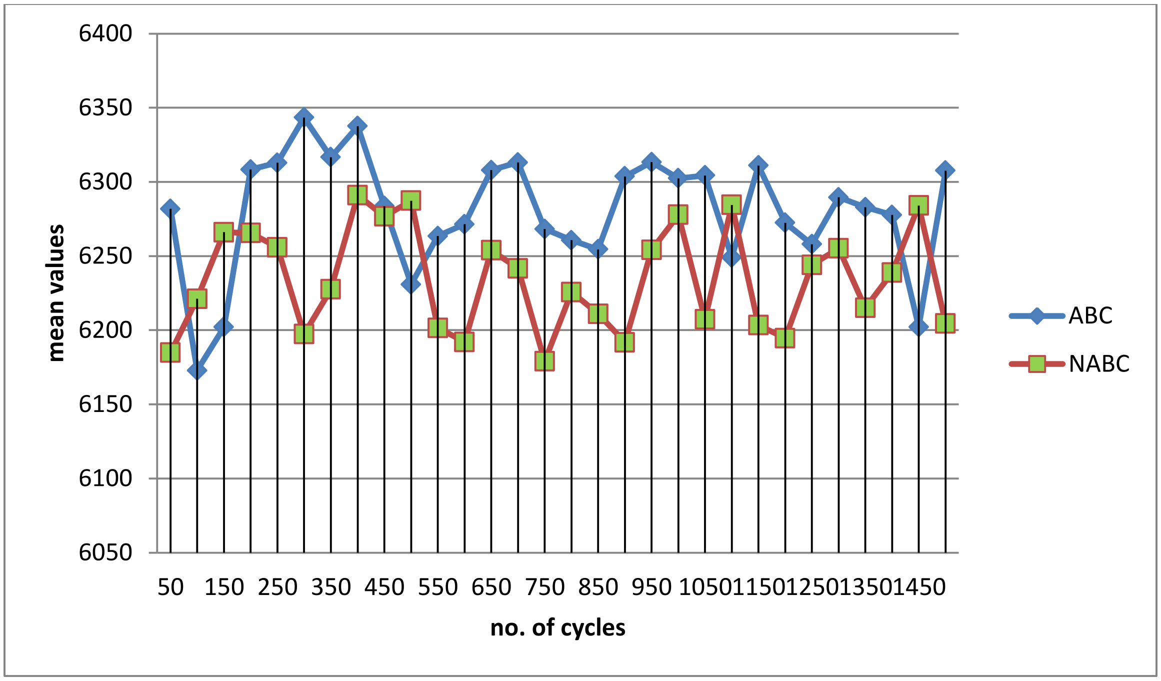

Fig.4.11 Performance Evaluation- NABC

As shown in the above figures 4.7-4.11, we took the value of lifetime after every 50 cycles for every run and took mean of it which represented in these figures which explains that irrespective of the Maxcycle value, all algorithms are performing better than ABC for maxcyclenumber=1500.

4.4For maxcycleno= 2000

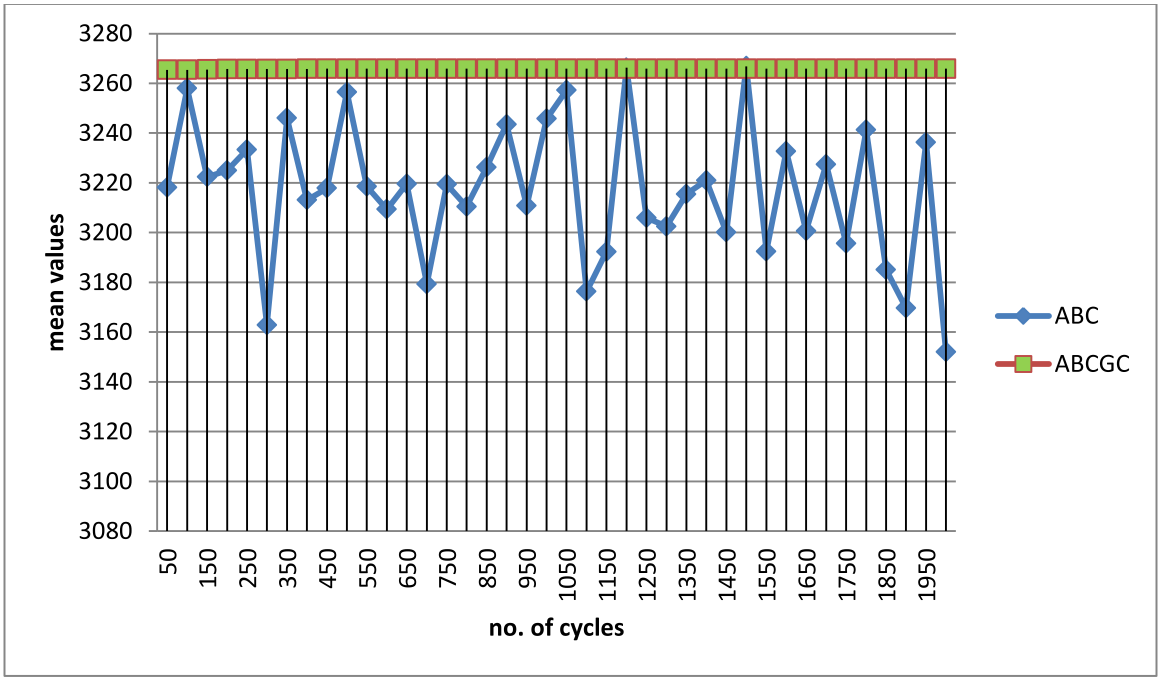

Fig.4.12 Performance Evaluation-ABCGC

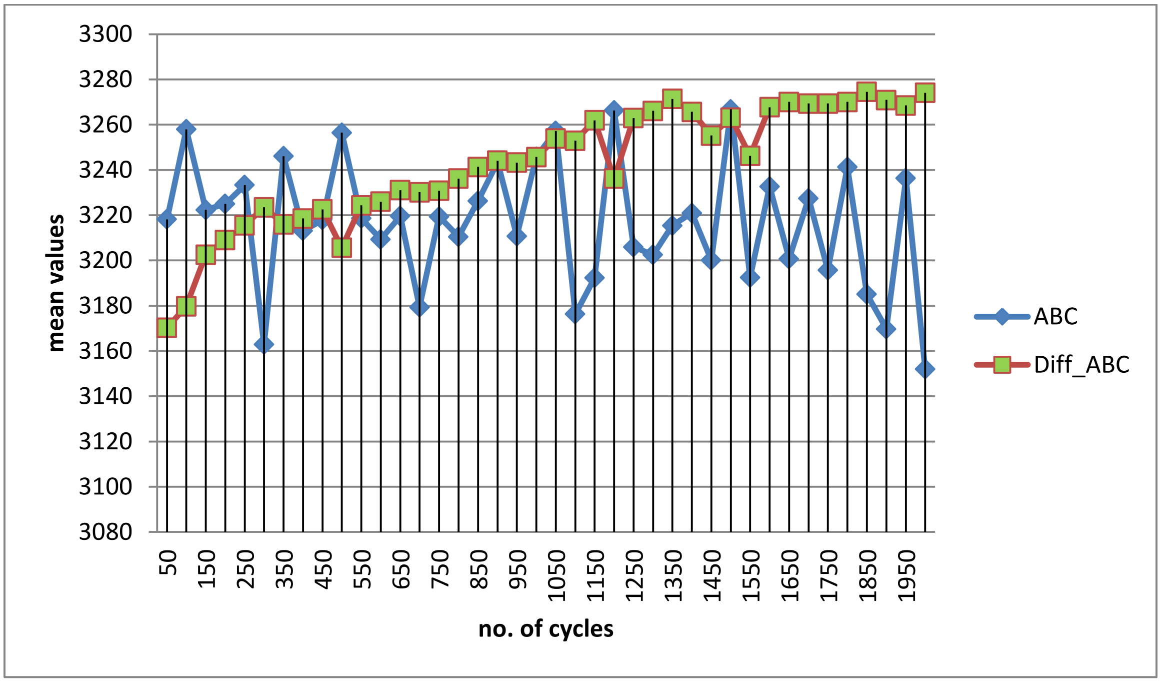

Fig.4.13 Performance Evaluation-Different ABC

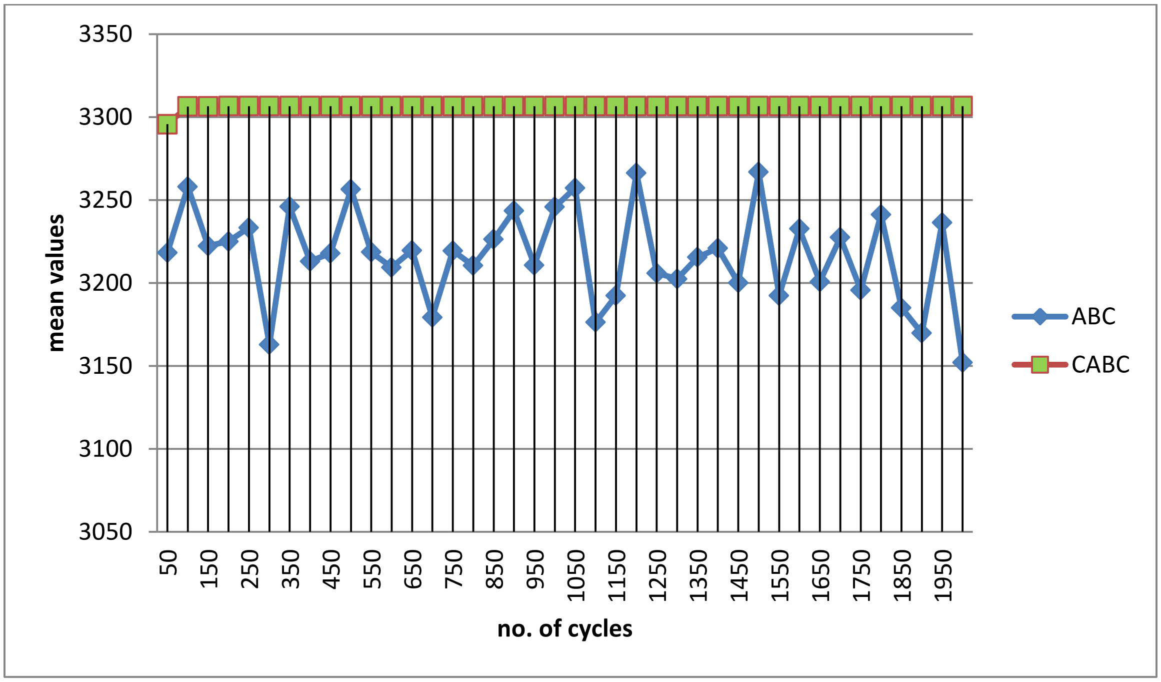

Fig.4.14 Performance Evaluation-CABC

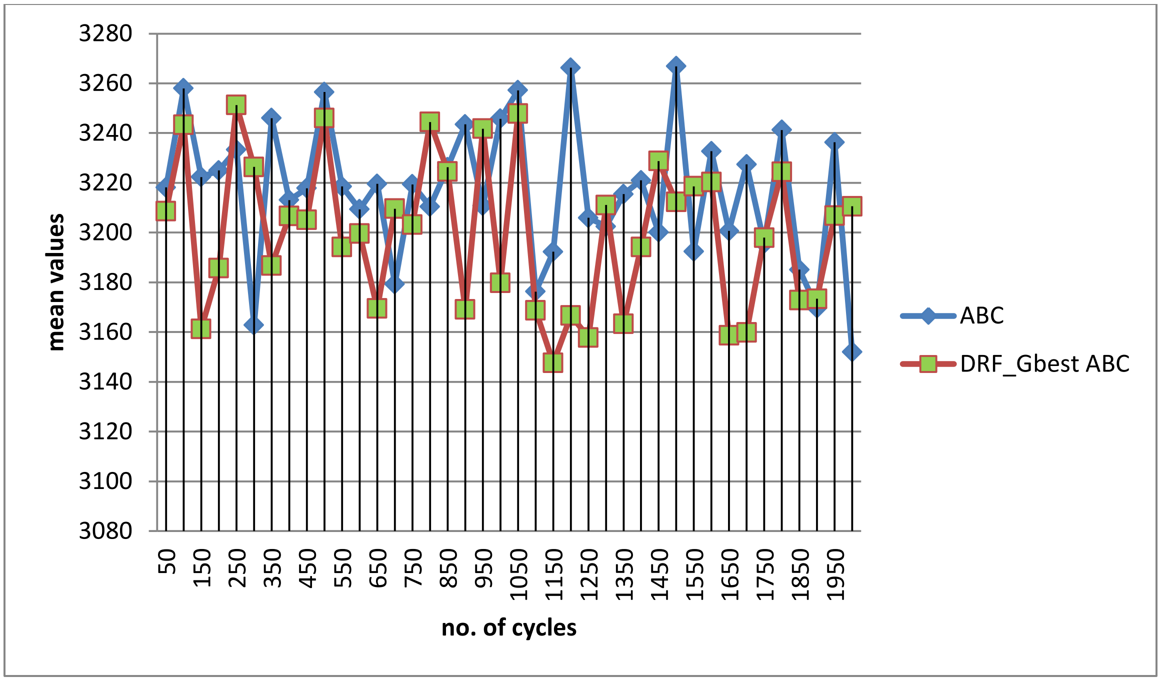

Fig.4.15 Performance Evaluation-DRF_GBest ABC

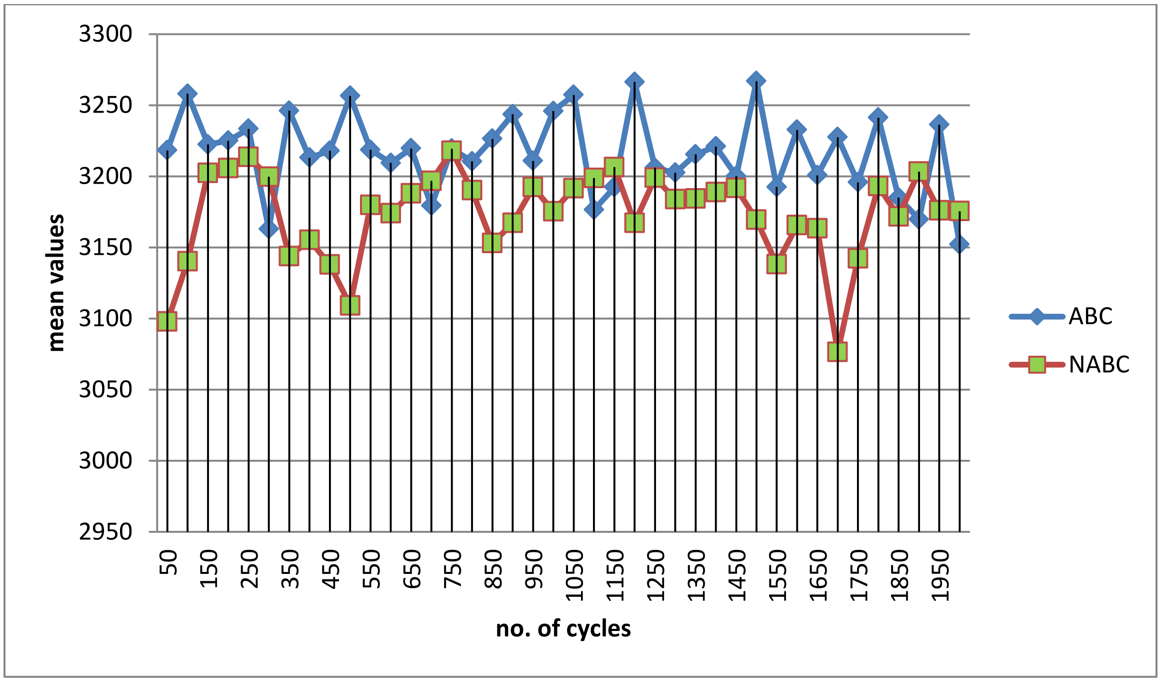

Fig.4.16 Performance Evaluation- NABC

As shown in the above figures4.12-4.16, we took the value of lifetime after every 50 cycles for every run and took mean of it which represented in these figures which explains that irrespective of the Maxcycle value, all algorithms are performing better than ABC for maxcycleno.=2000.

4.5 For maxcyclenumber= 2500

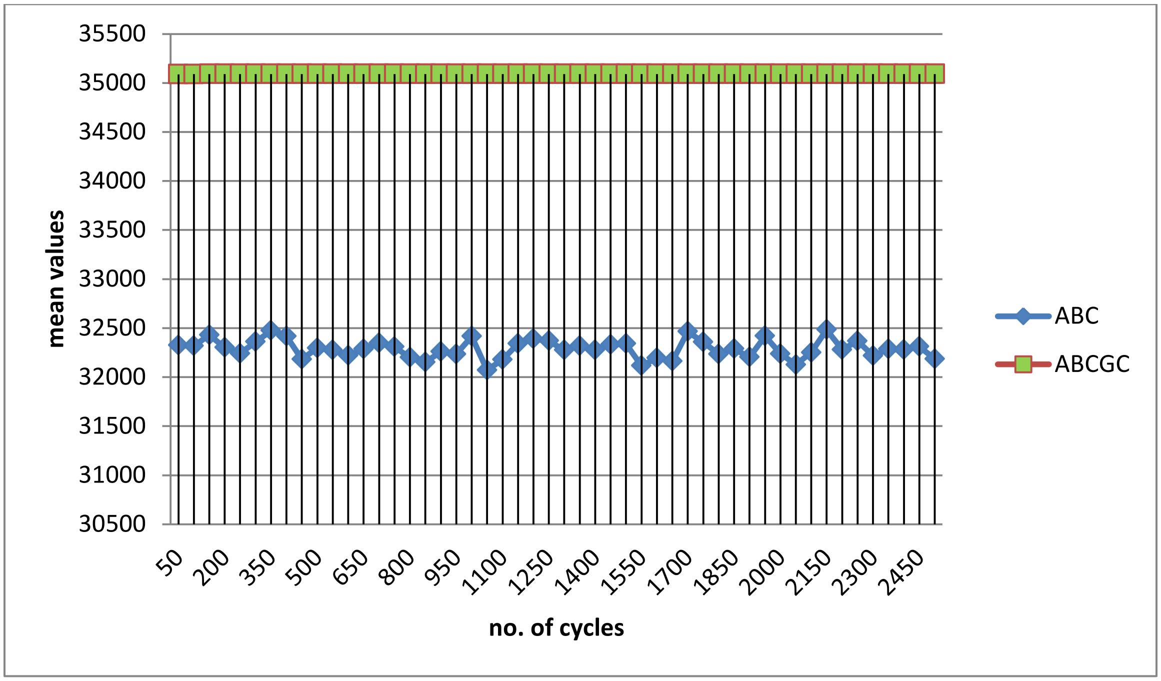

Fig.4.17 Performance Evaluation-ABCGC

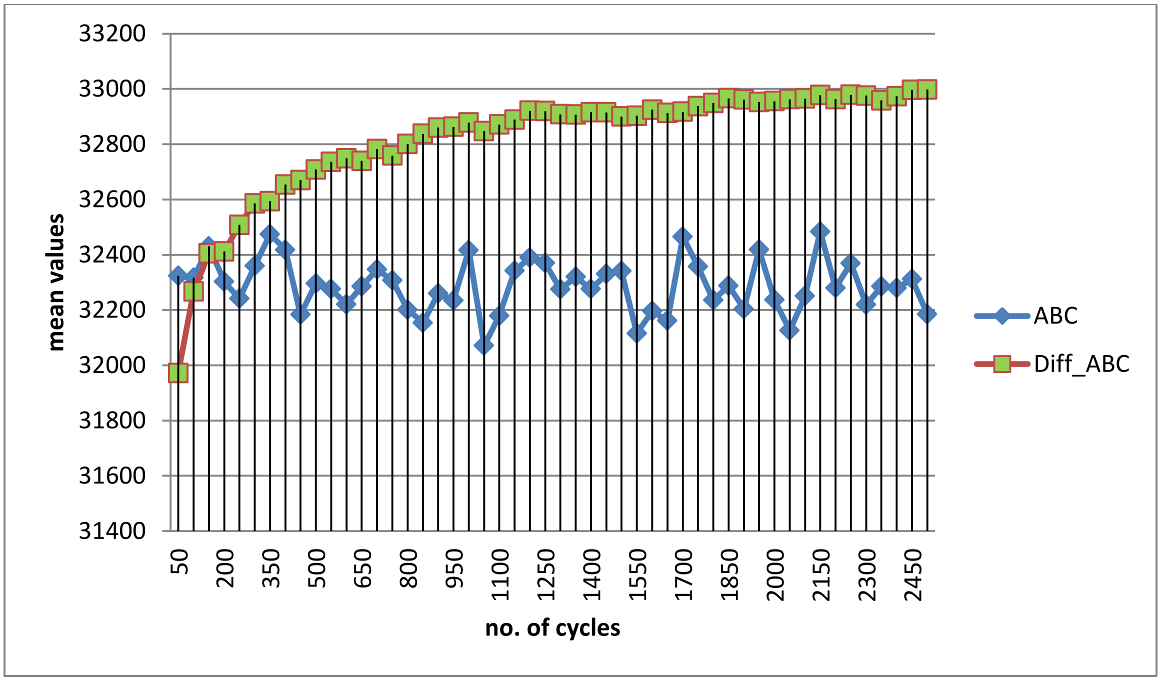

Fig.4.18 Performance Evaluation-Different ABC

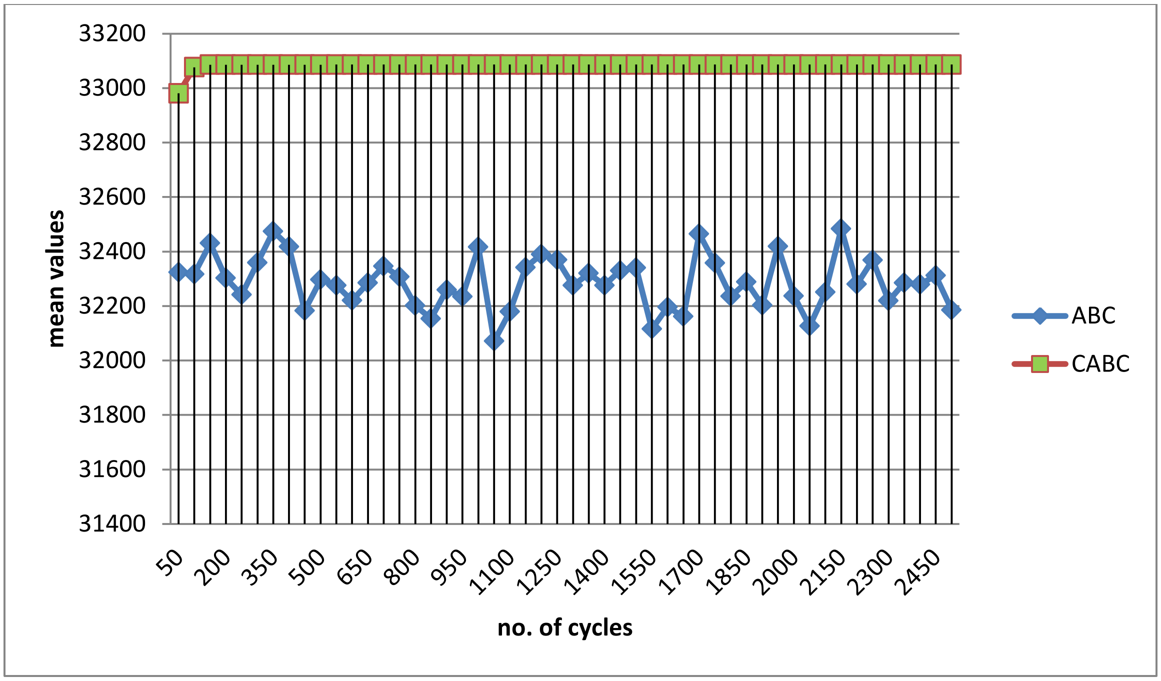

Fig.4.19 Performance Evaluation-CABC

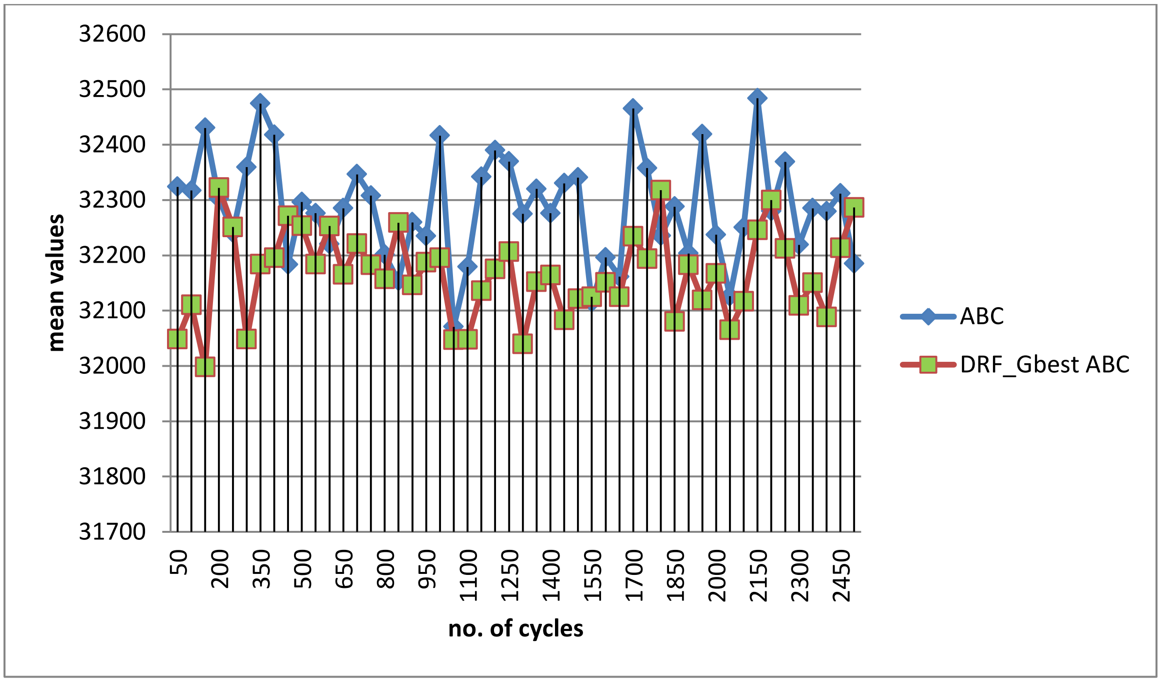

Fig.4.20 Performance Evaluation-DRF_GBest ABC

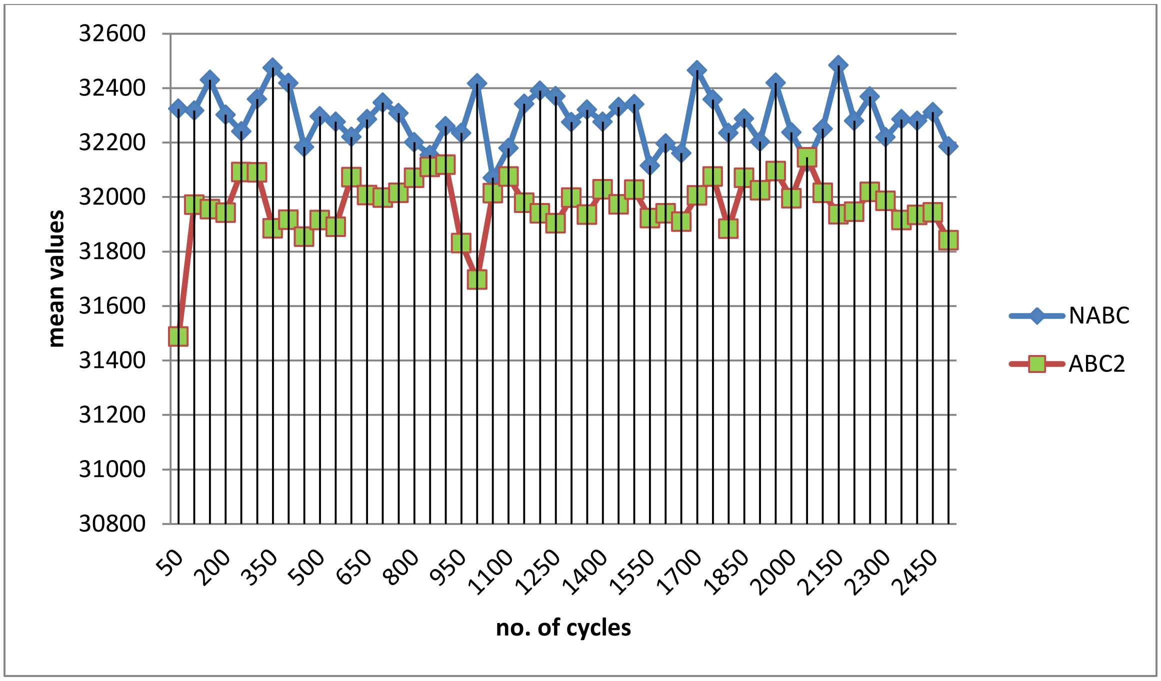

Fig.4.21 Performance Evaluation-NABC

As shown in the above figures4.17-4.21, we took the value of lifetime after every 50 cycles for every run and took mean of it which represented in these figures which explains that irrespective of the Maxcycle value, all algorithms are performing better than ABC for maxcyclenumber.=2500.

4.6For Maxcyclenumber = 3000

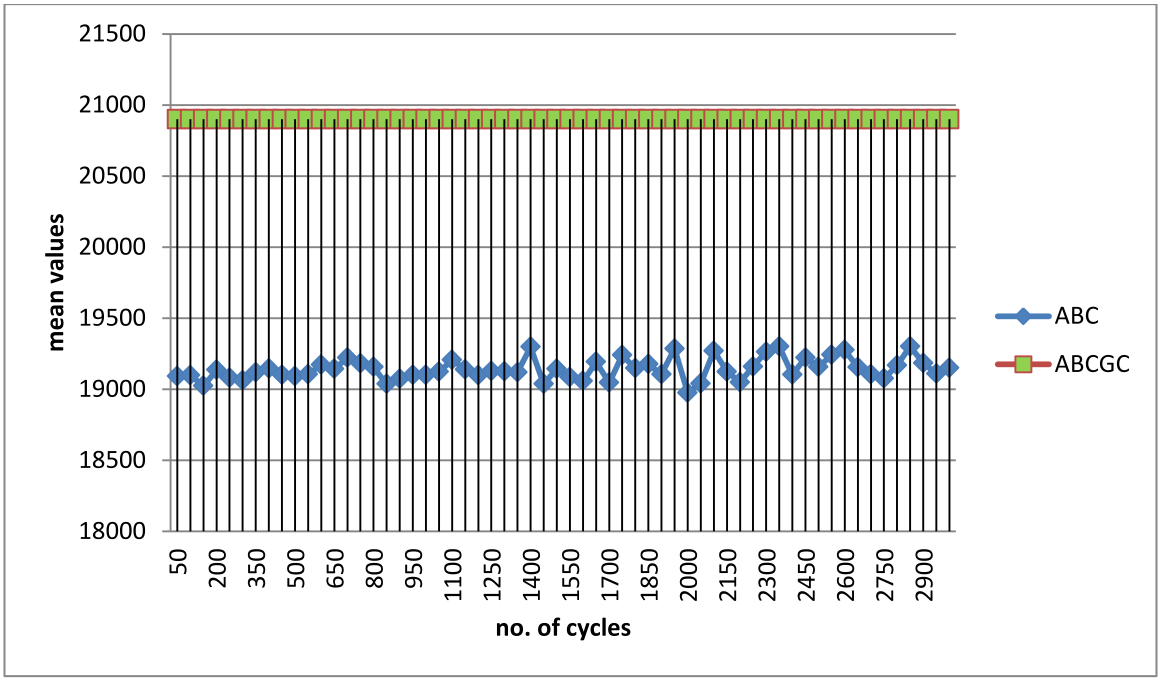

Fig.4.22 Performance Evaluation-ABCGC

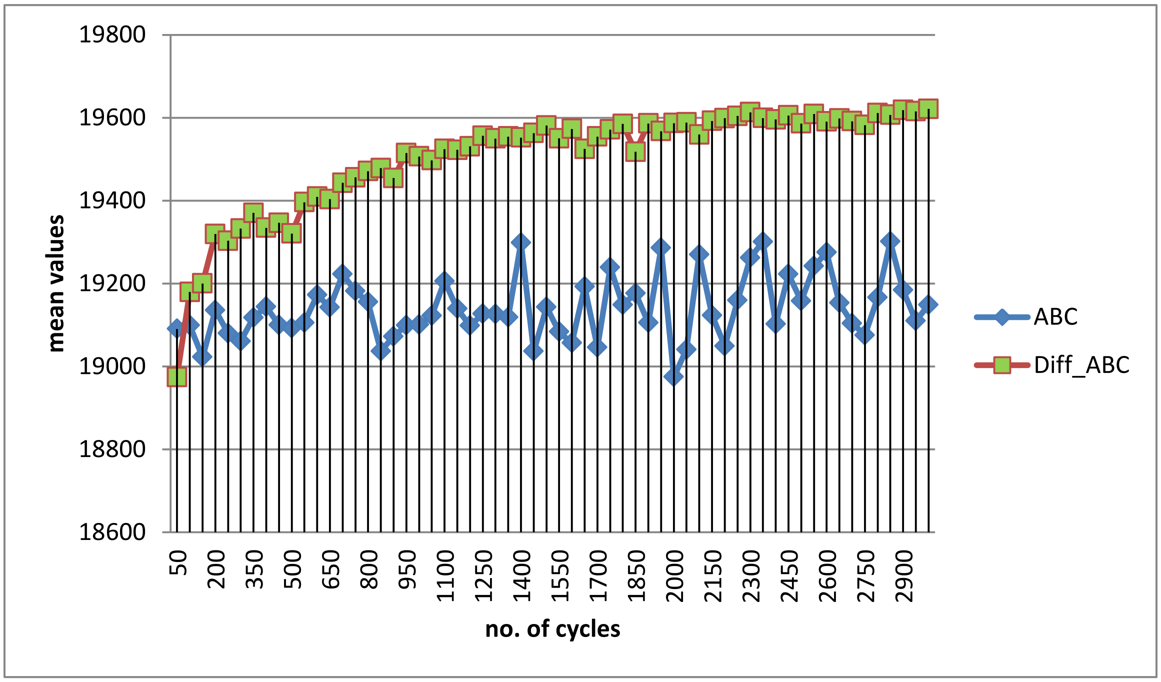

Fig.4.23 Performance Evaluation-Different ABC

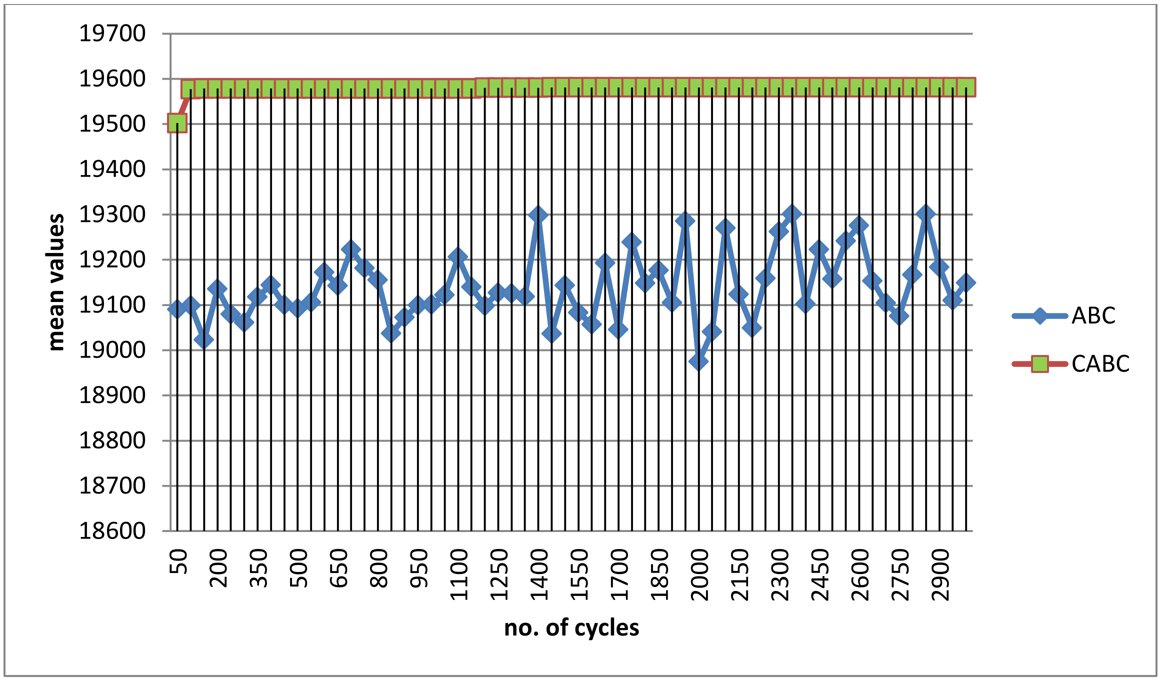

Fig.4.24 Performance Evaluation-CABC

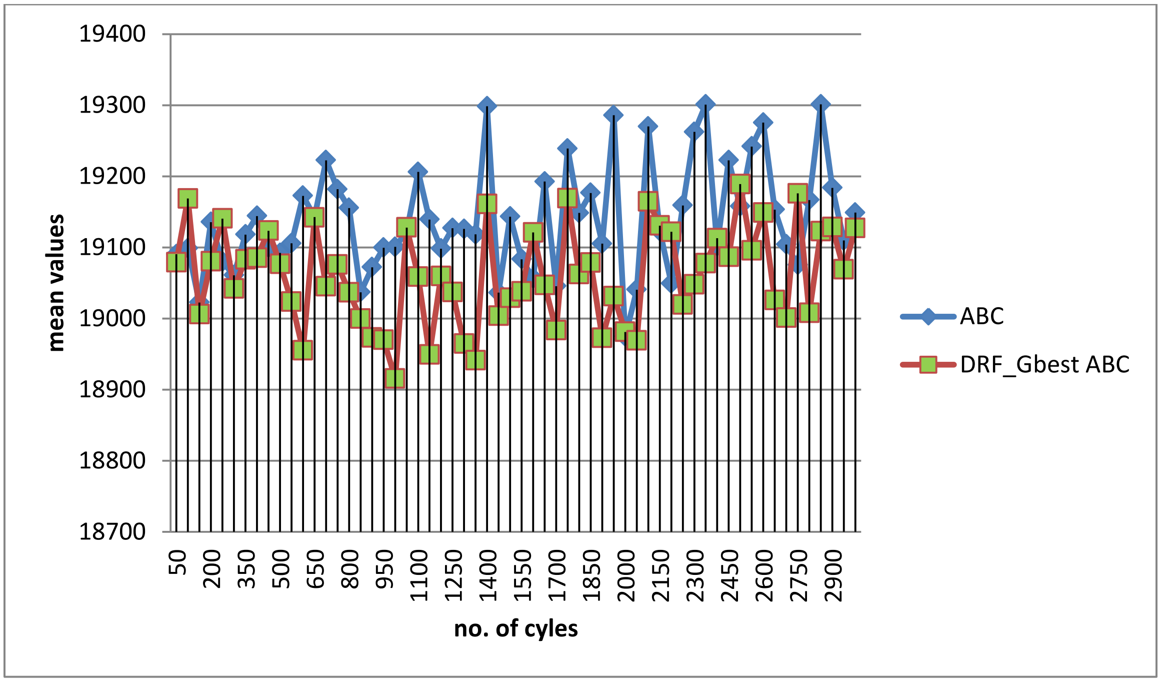

Fig.4.25 Performance Evaluation-DRF_GBest ABC

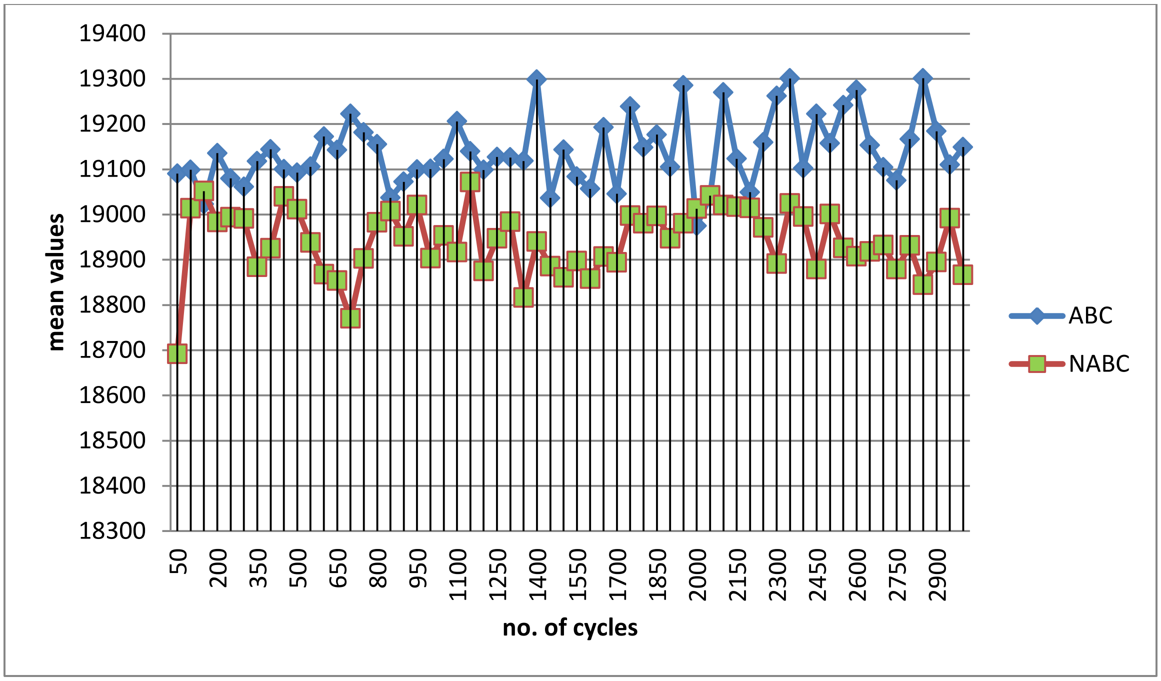

Fig.4.26 Performance Evaluation-NABC

As shown in the above figures4.22-4.26, we took the value of lifetime after every 50 cycles for every run and took mean of it which represented in these figures which explains that irrespective of the Maxcycle value, all algorithms are performing better than ABC for maxcyclenumber=3000.

4.7 Avg Mean values comparision

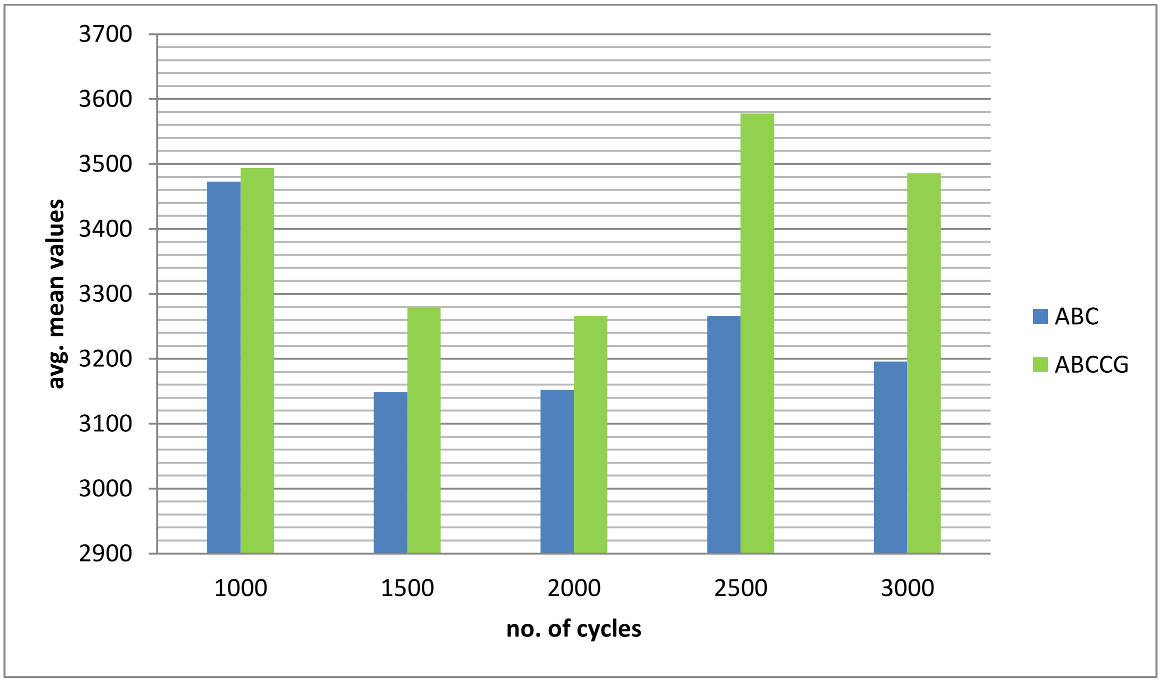

Fig.4.27 Mean values of ABC and ABCCG

Fig.4.27 represents the mean value of lifetime returned by both ABC and ABCGC for different values of MaxCycles, and it is clearly visible that ABCGC performs better than ABC. The mean difference between ABC and ABCGC is 5.39% on average according to values in fig.4.27.

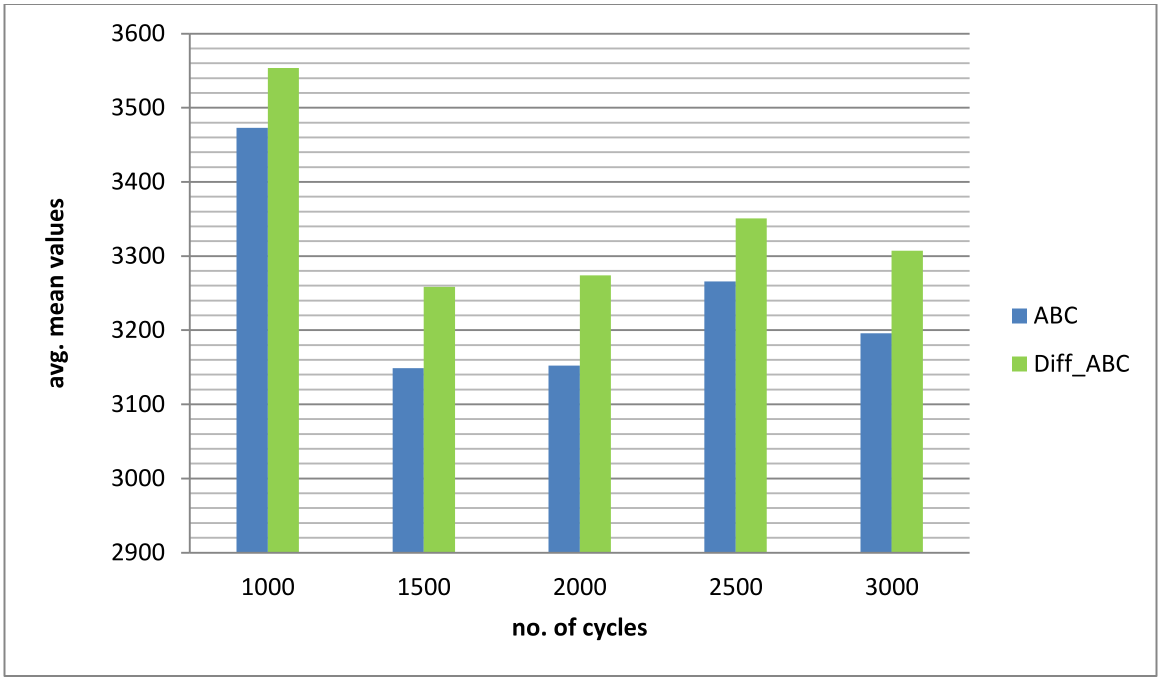

Fig.4.28 Mean values of ABC and Diff_ABC

Fig.4.28 represents the mean value of lifetime returned by both ABC and Diff_ABC for different values of MaxCycles, and it is clearly visible that Diff_ABC performs better than ABC. The mean difference between ABC and Diff_ABC is 1.577% on average according to values in fig.4.28.

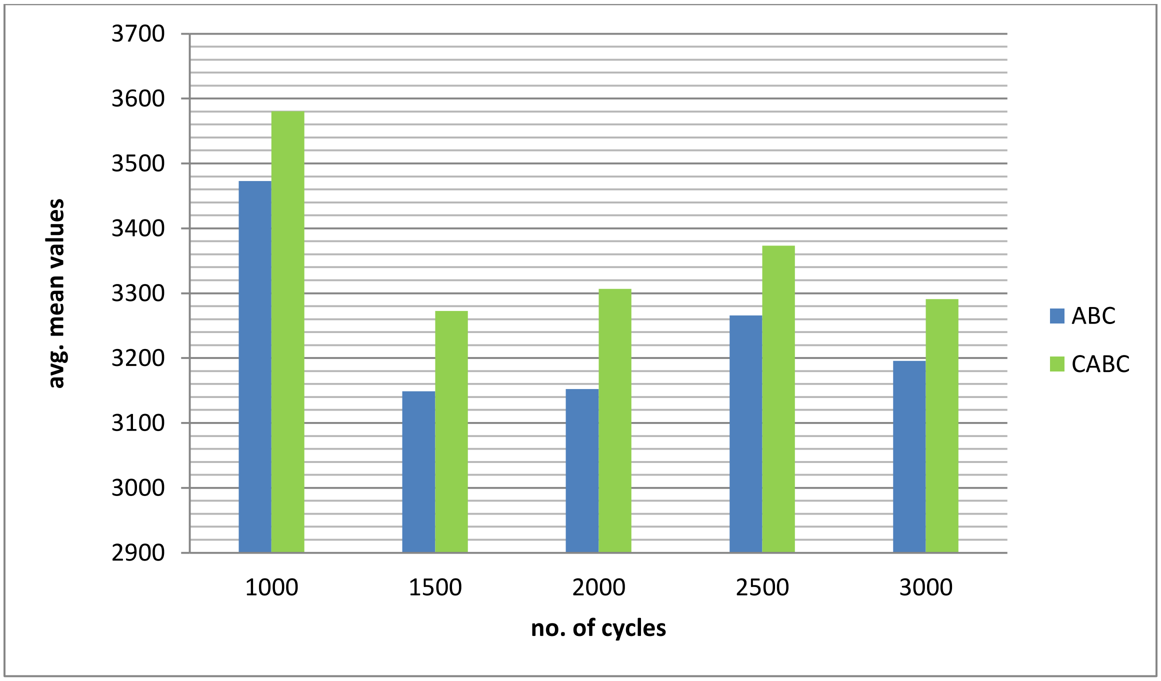

Fig.4.29 Mean values of ABC and CABC

Fig.4.29 represents the mean value of lifetime returned by both ABC and CABC for different values of MaxCycles, and it is clearly visible that CABC performs better than ABC. The mean difference between ABC and CABC is 1.819% on average according to values in fig.4.29.

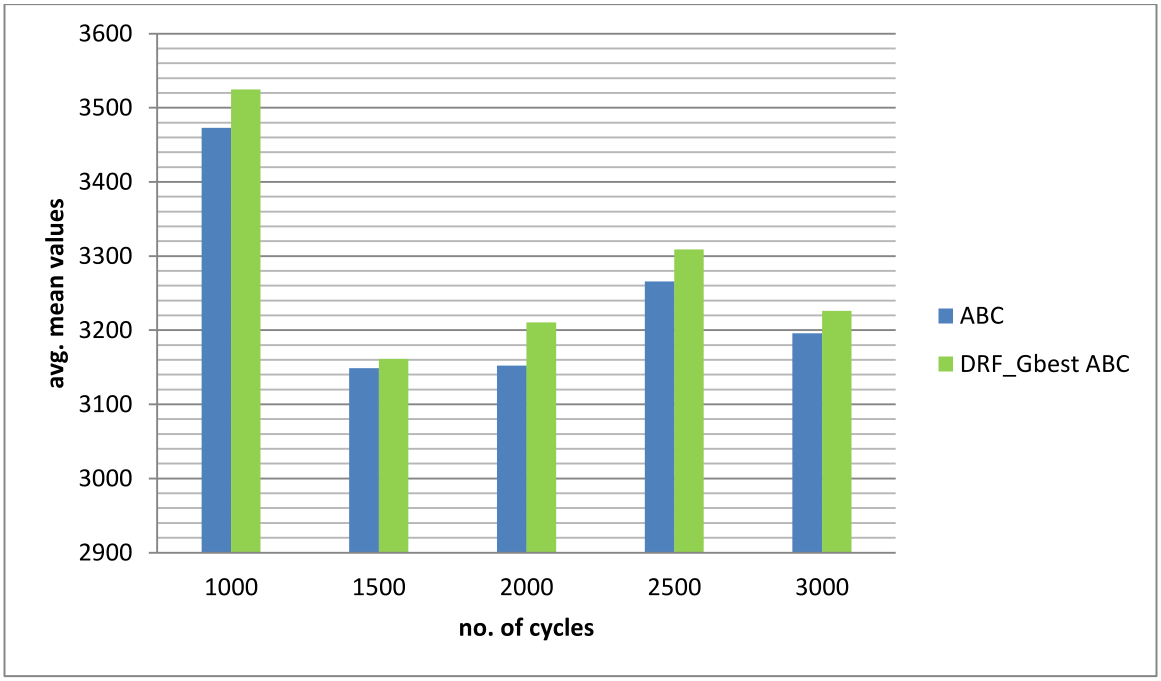

Fig.4.30 Mean values of ABC and DRF_GBest ABC

Fig.4.30 represents the mean value of lifetime returned by both ABC and DRF_GBest ABC for different values of MaxCycles, and it is clearly visible that DRF_GBest ABC performs better than ABC. The mean difference between ABC and DRF_GBest ABC is 6.020% on average according to values in fig.4.30

.

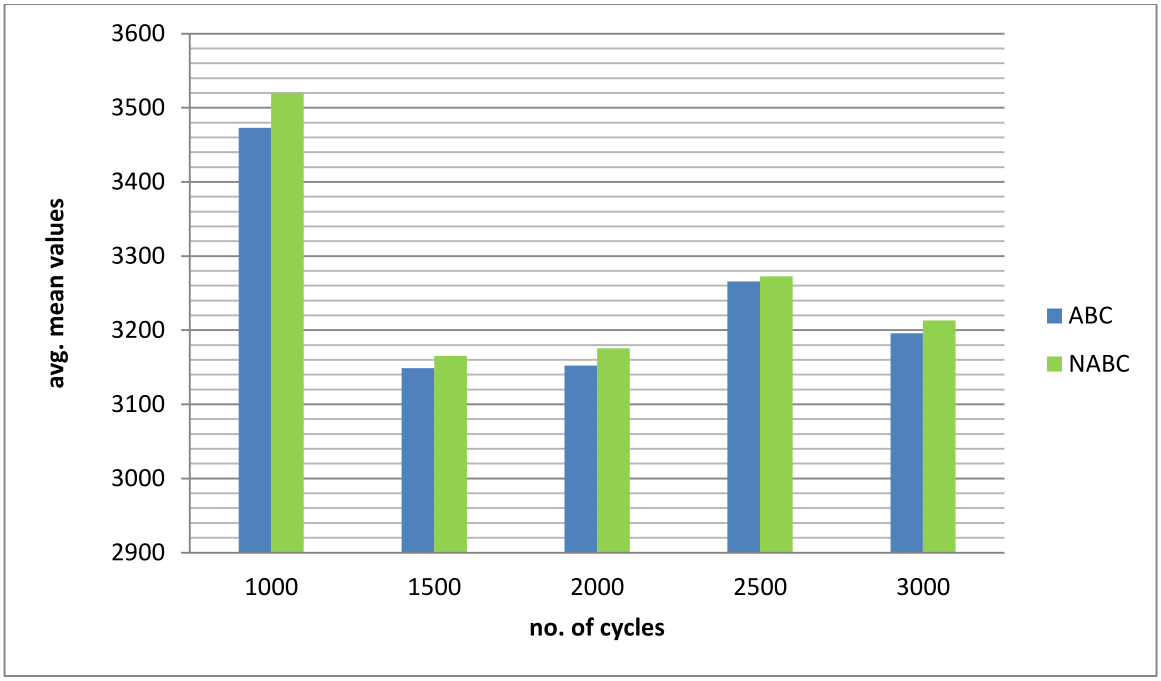

Fig.4.31 Mean values of ABC and NABC

Fig.4.31 represents the mean value of lifetime returned by both ABC and NABC for different values of MaxCycles, and it is clearly visible that NABC performs better than ABC. The mean difference between ABC and NABC is 3.363% on average according to values in fig.4.31.

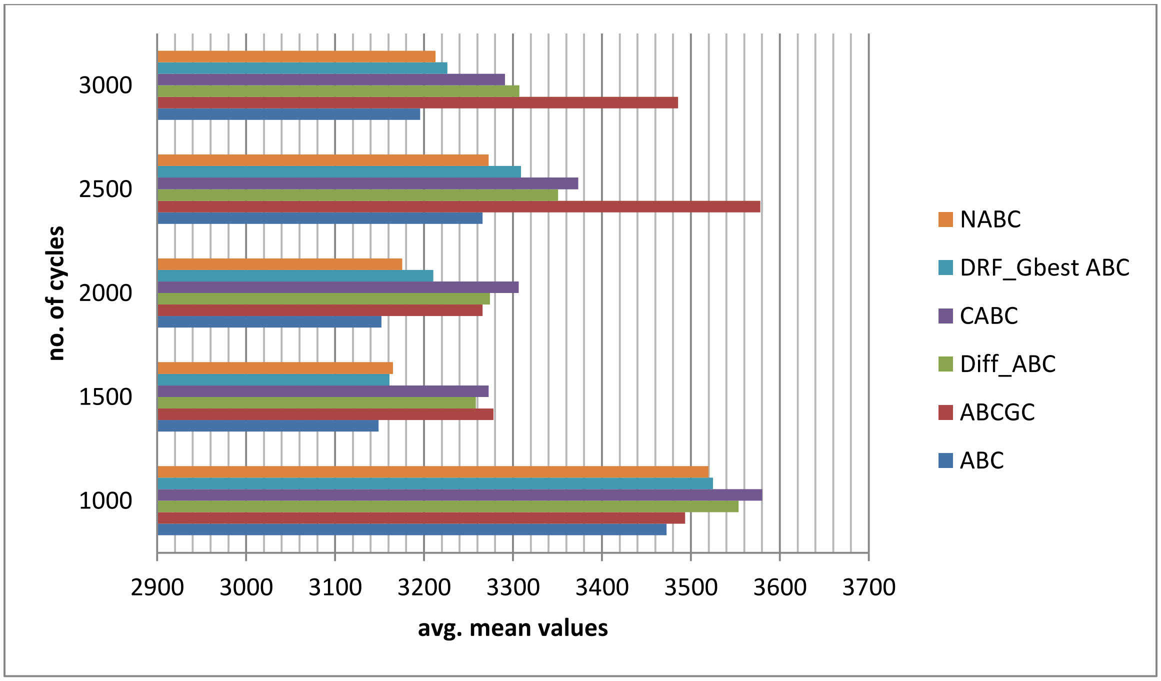

Fig.4.32 Cumulative comparision of all algorithms for different number of cycles.

Fig.4.32 represents the mean value of lifetime returned by both different algorithms for different values of MaxCycles, and it is clearly visible that all performs better than ABC.

CHAPTER 5

CONCLUSION AND FUTURE SCOPE

We dedicated our work to base station positioning problem in wireless sensor network for maximizing the network lifetime. We used ABC algorithm for solving the base station positioning problem which consists of two main features i.e. exploitation and exploration. For finding out the optimal results we need to find the perfect balance between these two features. We tried to find the balance via different modified algorithms discussed in the previous chapters. As shown in the previous results chapter, modified algorithms were successful in finding better solutions to our base station positioning problem. DRF_GBest ABC provide us the best result in form of 6.020% increase in network lifetime with respect to conventional ABC algorithm. Apart from DRF_GBest ABC, ABCGC also increased the network lifetime by approximately 5.39%. Thus the proposed approaches are observed to be more efficient than the conventional ABC approach and successfully address the objective that we undertook as part of this work.

Though the proposed approaches performed better than the conventional ABC algorithm, there is room for improvement in base station positioning problem field as well as ABC algorithm. We took the assumption for the network model of our problem that every sensor node can transmit its data to the base station in a single hop irrespective of the location of base station and its own location too. We could use a network model in which there is possibility of multi hop transmission. We can incorporate the multi hop network by managing the intermediate nodes before calculating the lifetime of sensor node for particular location of base station.

Another viable future scope of work is incorporating levy distribution for finding the step size used in the local search equation of the ABC algorithm. Levy distribution will increase the exploitation of our search by generating variable step size. Another modification that can be used in local search phase of ABC algorithm is Differential Evolution and Polynomial Mutation technique in which we do mutation and crossover which is used in the local search in the form of multiplication component of the search equation.

Apart from levy distribution and Differential Evolution and Polynomial Mutation technique, we could use a modification on ABC called as adaptive ABC. In adaptive ABC, we analyze the performance of our food sources after every cycle and on the basis of the quality of food sources we decide the number of employee bees and number of onlooker bees for the next cycle. In this way, we can analyze the performance of our algorithm at runtime and tweak it according to the requirement for exploration or exploitation.

Another scope of work in this field is to incorporate dynamic network. By dynamic network we mean that, the sensor nodes can move in the network and as they move, the previously found base station position may become sub-optimal. As discussed above, all our modifications outperforms conventional ABC by increasing the network lifetime, so this approach can be used in finding the optimal base station for the wireless sensor network or can be used as initial step for solving more complex problems like routing, or clustering.

REFERENCE

[1] A. Efrat, S. Har-Peled, J.S.B. Mitchell, “Approximation algorithms for two optimal location problems in sensor networks”, Proceedings of the 3rd International Conference on Broadband Communications, Networks and Systems (Broadnets‟05), Boston, Massachusetts , October 2005.

[2] Tzung-Pei Hong, Guo-Neng Shiu, “Allocating multiple base stations under general power consumption by the particle swarm optimization”, proceedings of the 2007 IEEE Swarm Intelligence Symposium (SIS 2007).

[3] Amir Mollanejad, Leili Mohammad Khanli, Mohammad Zeynali, “DBSR:Dynamic base station repositioning using genetic algorithm in wireless sensor network”, IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 2, No 2, March 2010.

[4] Dervis Karaboga, Bahriye Akay, “A comparative study of artificial bee colony algorithm”, Applied Mathematics and Computation 214 (2009) 108–132.

[5] A. Akkaya, M. Younis, W. Youssef, "Positioning of Base Stations in Wireless Sensor Networks", IEEE Communications Magazine, Vol. 45, No.4, pp. 96-102, April 2007.

[6] J. Pan , Y. T. Hou , L. Cai , Y. Shi and S. X. Shen "Locating base-stations for video sensor networks", Proc. IEEE Vehicular Technology Conf., 2003

[7] Ivan Vilovic, Niksa Burum, Zvonimir Sipus, “Ant colony approach in optimization of base station position”, Proceedings of the European Conference on Antennas and Propagation (EuCAP '09), pp. 2882–2886, Berlin, Germany, March 2009.

[8] M. Dorigo, G. Di Caro, The ant colony optimization meta-heuristic, New Ideas in Optimization, McGraw-Hill, London, 1999, pp. 11–32.

[9] Yunyue Lin, QishiWu, Xiaoshan Cai, Xiaojiang Du, and Ki-Hyeon Kwon, “On

Deployment of Multiple Base Stations for Energy-Efficient Communication in Wireless Sensor Networks" Hindawi Publishing Corporation International Journal of Distributed Sensor Networks Volume 2010, Article ID 563156, 8 pages doi:10.1155/2010/563156.

[10] A. Bogdanov E. Maneva, S. Riesenfeld, Power-aware base station positioning for sensor networks, in: Proceedings of the 23rd International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM"04), Hong Kong, March 2004.

[11] J. Kennedy and R. Eberhart, "Particle swarm optimization." Proc. IEEE International Conf. on Neural Networks (Perth, Australia), IEEE Service Center, Piscataway, NJ, 1995 (in press).

[12] P.D. Hossein Zadeh, C. Schlegel, and M.H. MacGregor, "Distributed optimal dynamic base station positioning in wireless sensor networks", Computer Networks, 56(1):34-49, January 2012.

[13] Rajiv Kr. Tripathi, Sateeshkrishna Dhuli, Y. N. Singh, Nishchal K. Verma, "Analysis of Weights for Optimal Positioning of Base Station in a Wireless Sensor Network" Communications (NCC), 2014 Twentieth National Conference on, 1-4

[14] Dorottya Vass , Attila Vidács, Positioning mobile base station to prolong wireless sensor network lifetime, Proceedings of the 2005 ACM conference on Emerging network experiment and technology, October 24-27, 2005, Toulouse, France

[16] Choi, Y. S., Kim, K. S., Kim, N., 2007, in IFIP International Federation for Information Processing, Volume 245, Personal Wireless Communications, eds. Simak, B., Bestak, R., Kozowska, E., (Boston: Springer), pp. 217-229.

[17]Soo Kim, Jeong-Gil Ko, Jongwon Yoon, Heejo Lee, “Multiple-Objective Metric for Placing Multiple Base Stations in Wireless Sensor Networks”, In Proceedings of the International Symposium on Wireless Pervasive Computing, pages 627-631, February 2007.

[18]S.R. Gandham, M. Dawande, R. Prakash, S. Venkatesan, Energy efficient schemes for wireless sensor networks with multiple mobile base stations, in: Proc. IEEE Globecom 2003, San Francisco, CA December 1–5, vol. 1, 2003, pp. 377–381.

[17] Mahmud, S., Wu, H., Xue, J.: Efficient Energy Balancing Aware Multiple Base Station Deployment for WSNs. In: Proceedings of the 8th European Conference on Wireless Sensor Networks, Bonn, Germany, pp. 179–194 (2011).

[18] M. A. Guvensan, Z. C. Taysi, and A. G. Yavuz, "Multiple base station positioning for sensor applications," in Proceedings of the European Computing Conference, ser. Lecture Notes in Electrical Engineering, N. Mastorakis, V. Mladenov, and V. T. Kontargyri, Eds. Springer US, 2009, vol. 27, pp. 619-629.

[19] Anurag Singh Tomar, Gaurav Kumar Tak, “Optimized Positioning Of Multiple Base Station for Black Hole Attack”, International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 3 Issue 8, August 2014.

[20] Y. Shi, Y.T. Hou, Approximation algorithm for base station placement in wireless sensor networks, in: Proceedings of the Fourth Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON "07),June 2007, pp. 512–519.

[23] A. Ahrari, M. Shariat-Panahi, and A. A. Atai, “GEM: a novel evolutionary optimization method with improved neighborhood search,” *Applied Mathematics and Computation,* vol. 210, no. 2, pp. 376–386, 2009.

[24]Garrison W. Greenwood, Senior Member, IEEE Shubham Chopra," A Modified artificial Bee Colony Algorithm for Solving Large Graph Theory Problems", 2013 IEEE Congress on Evolutionary Computation, June 20-23, Cancún, México

[25]Gaowei YAN †, Chuangqin LI," An Effective Refinement Artificial Bee Colony Optimization Algorithm Based On Chaotic Search and Application for PID Control Tuning", Journal of Computational Information Systems 7: 9 (2011) 3309-3316

[26]Jiuyuan Huo and Fanming Meng, Lanzhou, P.R. China," An Improved Gbest-guided Artificial Bee Colony Algorithm Based on Dynamic Regulatory Factor", 8th International Conference on Advanced Computational Intelligence Chiang Mai, Thailand; February 14-16, 2016

[27]Yanyu Zhang, Peng Zeng, Yang Wang, Baohui Zhu, Fangjun Kuang," Linear Weighted Gbest-guided Artificial Bee Colony Algorithm", 2012 Fifth International Symposium on Computational Intelligence and Design

[28] Yujiang Yi, Renjie He." A Novel Artificial Bee Colony Algorithm", 2014 Sixth International Conference on Intelligent Human-Machine Systems and Cybernetics