DEVELOPMENT OF A MODIFIED BACTERIAL FORAGING OPTIMIZATION ALGORITHM BASED BLACK HOLE ATTACK MITIGATION MODEL FOR WIRELESS SENSOR NETWORKS

BY

ZAINAB MUKHTAR ABUBAKAR

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

FACULTY OF ENGINEERING

AHMADU BELLO UNIVERSITY,

ZARIA

JANUARY, 2017

DEVELOPMENT OF A MODIFIED BACTERIAL FORAGING OPTIMIZATION ALGORITHM BASED BLACK HOLE ATTACK MITIGATION MODEL FOR WIRELESS SENSOR NETWORKS

BY

Zainab Mukhtar ABUBAKAR, B.Eng (BUK) 2008

P13EGCM8024

zmukhtarabubakar@gmail.com

A DISSERTATION SUBMITTED TO THE SCHOOL OF POSTGRADUATE STUDIES, AHMADU BELLO UNIVERSITY, ZARIA IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF A MASTER OF SCIENCE (MSc) DEGREE IN COMPUTER ENGINEERING

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING
FACULTY OF ENGINEERING
AHMADU BELLO UNIVERSITY,
ZARIA, NIGERIA

JANUARY, 2017

Declaration

I declare that the work in this dissertation entitled “Development of a Modified Bacterial Foraging Optimization Algorithm Based Black Hole Attack Mitigation Model for Wireless Sensor Networks” has been carried out by me in the Department of Electrical and Computer Engineering. The information derived from the literature has been duly acknowledged in the text and a list of references provided. No part of this dissertation was previously presented for another degree or diploma at this or any other institution.

Zainab Mukhtar ABUBAKAR	------------------------------	-----------------------------
	Signature	Date

Certification

This dissertation entitled "DEVELOPMENT OF A MODIFIED BACTERIAL FORAGING OPTIMIZATION ALGORITHM BASED BLACK HOLE ATTACK MITIGATION MODEL FOR WIRELESS SENSOR NETWORKS" by Zainab Mukhtar ABUBAKAR meets the regulations governing the award of the degree of Master of Science (M.Sc) in Computer Engineering of the Ahmadu Bello University, and is approved for its contribution to knowledge and literary presentation.

Chairman, Supervisory Committee	--------------------------	-------------------------
(Prof. M. B. Mu'azu)	Signature	Date
Member, Supervisory Committee	--------------------------	-------------------------
(Dr. T. H. Sikiru)	Signature	Date
Head of Department	--------------------------	-------------------------
(Dr. Y. Jibril)	Signature	Date
Dean, School of Postgraduate Studies	--------------------------	--------------------------
(Prof. K. Bala)	Signature	Date

Dedication

This research work is dedicated to Almighty Allah (S.W.T) and my beloved parents.

Acknowledgement

All praises are due to Allah (SWT). Peace and blessings be upon His Prophet, Muhammad (SAW) and his companions.

My endless appreciation goes to the chairman of my supervisory committee, Prof. M.B Mu'azu, for his tireless effort, support, patience, constant guidance and encouragement from the beginning to the end of this dissertation. My appreciation also goes to my co-supervisor Dr. T.H Sikiru for his guidance and contributions. May Allah reward them abundantly. To all lecturers of Electrical and Computer Engineering namely, Prof. B. G. Bajoga, Prof B. Jimoh, Dr. Y. Jibril, Dr. S.M Sani, Dr. S. Garba, Dr. K.A. Abubilal, Dr. A.D Usman, Dr. A.M.S. Tekanyi, Dr. A.E Adewale, Dr. I.J. Umoh, Dr. H.A Abubakar, Dr. S. Man-yahaya, Engr. Z.M Abdullahi, Engr. A.S Musa, Engr. G.A Olarinoye, Engr. P.U Okorie, Engr A.I. Abdu, Engr. M.J. Muazu, Engr. J.O. Haruna, Engr A. Mohammed, Engr. S.T Ajayi, Engr. E. Okafor and those too numerous to mention, I thank you for the valuable contributions and encouragement.

Special thanks to my parents Alhaji Mukhtar Abubakar and Hajiya Rakiya Bello for their support and prayers. Words alone cannot express my appreciation.

My profound gratitude goes to my hubby Arc. Waziri Ibrahim for his understanding, care, kindness, patience and words of encouragement. I say thanks to my lovely son Ammar Ibrahim for his patience and understanding.

I appreciate my siblings; Nafisa, Aminu, Dayyaba, Bilkisu, Lukman, Aisha, Fatima, Hauwa and Asmau, my nieces; Ijlal, Nusaiba and Farha, my nephew; Ahmad-Mukhtar (Abba), my inlaws especially Amina Aminu, Alh. Q.A.Arabi, Lt. I.B Umar, Fatima Mailafia, Zubairu, Al-amin and Umar Waziri, for all the care and prayers.

I say thank you to Tijjani .A. Salawudeen for his guidance and immense contribution to the completion of this research work. I am grateful to my friends and my colleagues; Basira, Safiyya, Bukky, Jagila, Tukur Lawal, Shittu, Idris, B.O Sadiq, A.A. Saidu, E. Obi, E. Ochia, Atuman, Ajayi, Mal. Tukur Abdullahi, the entire members of Computer/Control research group and people of the Department of Electrical and Computer Engineering.

Zainab Mukhtar ABUBAKAR
January 2017

Abstract

This study is aimed at the development of a mechanism for black hole attack mitigation in Wireless Sensor Network (WSN) using a modified Bacterial Foraging Optimization Algorithm (BFOA). A total of 200 randomly generated bacterial sensor nodes with a communication range of 20*m* were deployed in a 100*m*x100*m* network coverage area, consisting of four base stations. The radii 20*m*, 30*m* and 40*m* were chosen for the black hole region. The algorithm was implemented in MATLAB R2015b. In all tests carried out, the results obtained at 40m radius showed the effect of the black hole attack better than those at 20*m* and 30*m*. Successful packet delivery probabilities of 83.52%, 95.78%, 97.26% and 99.78% respectively were achieved at 40m radius for one, two, three and four base stations respectively. Significant reduction in false positive was observed when the base stations were increased. A negligible value of about 0.003% false positive was observed with four base stations using 40*m* radius of black hole region. Average delivery times of 31sec, 37sec, 43sec and 49sec were achieved at 40*m* radius for one, two, three and four base stations respectively. The times indicated that the routing complexity increased as the number of base stations increased. The performance of the modified BFOA based method showed packet delivery probability improvement of 5.48%, 9.67%, 0.18% and 1.01% over the standard BFOA based method as the base stations were increased from one to four respectively. As the base stations were increased to five, six, seven and eight, successful packet delivery probabilities of 99.39%, 99.69%, 99.79% and 99.82% respectively were achieved using 40*m* radius of black hole region. The trend observed for packet delivery showed that optimum efficiency is achieved with quadrant placement of base stations. The results obtained indicated that optimal placement of the base stations minimized the effect of black hole attack and ensured successful packet delivery.

Table of Contents

CHAPTER TWO: LITERATURE REVIEW

CHAPTER THREE: MATERIALS AND METHODS

List of Figures

List of Tables

List of Abbreviations

Acronym	Definition
ABC	Artificial Bee Colony
ACO	Ant Colony Optimization
AFSA	Artificial Fish Swarm Algorithm
BFOA	Bacterial Foraging Optimization Algorithm
DoS	Denial of Service
IWD	Intelligent Water Drop
LEACH	Low Energy Adaptive Clustering Hierarchy
MANET	Mobile Adhoc Network
MATLAB	Matrix Laboratory
MTRP	Multicast Tree Assisted Random Propagation
PSO	Particle Swarm Optimization
RREP	Route Reply
RREQ	Route Request
WSN	Wireless Sensor Network

CHAPTER ONE

INTRODUCTION

1.1 Background

According to Annu and Chaudhary (2015), "Wireless Sensor Network (WSN) is an interconnection of a large number of nodes deployed for monitoring a system by means of measurement of its parameters". Due to its wide range of applications in both the military and civilian domain, it is emerging as a prevailing technology for the future. It is used in industrial process control and monitoring, healthcare monitoring, environment and habitat monitoring, disaster management, structural monitoring and lots more (Arya & Raina, 2014). Wireless Sensor Networks are prone to security attacks due to severe constraints such as broadcast nature of transmission medium, limited battery power, small memory and susceptibility to physical capture because they are deployed in hostile and physically non-protected areas. As such, security is a major concern in WSN. There are many possible attacks on sensor networks such as Denial of Service (DoS) attack, selective forwarding, sybil attack, sink hole attack, black hole attack, hello flood attack and wormhole attack (Sharma & Thakur, 2014; Sharma & Ghose, 2010).

One of the most severe attacks is the black hole attack, which drops the entire packet. It is a type of routing attack whereby an intruder captures and reprograms a set of sensor nodes in a network so that they do not transmit the generated or received data packets to their original destinations.Black hole attack prevention techniques proposed in literature either use neighbourhood interactions and message overhearing (Karakehayov, 2005; Roy *et al.*, 2008), or secret sharing and path diversity (Ketel *et al.*, 2005; Lou & Kwon, 2006).

It is assumed that based on neighbourhood message interactions and overhearing, a sensor node close to a black hole node can monitor and report the black hole node. Though, when many sensor nodes within the same location are compromised, they can conspire and make the technique ineffective. A much better technique is that based on path diversity and secret sharing, though it is still not very effective (Arya & Raina, 2014; Dighe & Vaidya, 2013; Misra *et al.*, 2011; Sheela & Mahadevan, 2012). In this technique, messages are transformed into multiple shares by secret sharing schemes and then the shares are delivered via multiple independent paths to the destination.Figure 1.1 shows a sensor node, which transmits data to the base station using four node disjoint paths. The figure indicates that none of the packets passing through the four paths reaches the base station due to the deliberate placement of the black hole region, despite the path diversity. This indicates that routing based on multiple paths can perform badly in the presence of black hole attacks (Dighe & Vaidya, 2013; Misra *et al.*, 2011).

Successful packet delivery to the destination, which is the base station, without loss of data in a WSN, is more essential than the requirement of prevention of data being captured by an adversary (Arya & Raina, 2014; Misra *et al.*, 2011). As such, this research developed a mechanism for successful packet delivery using multiple base stations in the presence of black hole nodes in a WSN. Modified bacterial foraging optimization technique was used to determine the optimal location to deploy the base stations in this work and in the mitigation of the black hole nodes.

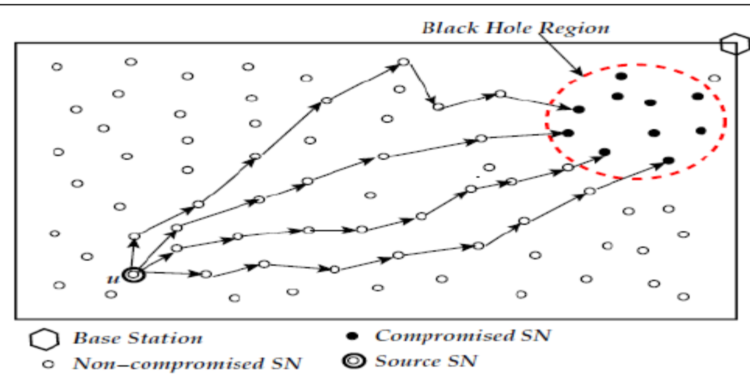

Figure 1.1: Data Delivery and Black Hole Region in a WSN with One Base Station (Misra *et al.*, 2011)

BFOA proposed by Passino (2002) based on the social foraging behaviour of a bacteria Escherichia *Coli* is a new member in the coveted domain of swarm intelligence that has drawn the attention of researchers in different fields of knowledge with regards to its biological motivation and graceful structure (Das *et al*., 2009). It has been applied for solving practical engineering problems like optimal control, harmonic estimation, machine learning, channel equalization and others (Annu & Chaudhary, 2015; Latiff *et al.*, 2007). The algorithm is based on imitating the searching behaviour of the bacteria in terms of positioning, handling and food ingestion. It has advantages such as high computational efficiency, high precision, fast convergence and global optimization (Kandasamy *et al.*, 2014; Li *et al.*, 2014). This research is aimed at the development of a black hole mitigation (detection and elimination) based model using an intelligent bio-inspired optimization algorithm (BFOA). A lot of research have been conducted on the area of malicious beacons detection using swarm intelligence techniques such as: Intelligent Water Drop (IWD) (Qureshi *et al.*, 2011), Ant Colony Optimization (ACO) (Iftikhar & Fraz, 2013), Particle Swarm Optimization (Khan & Iftikhar, 2013) and Bacterial Foraging Optimization (Arya & Raina, 2014). In this research, a modified BFOA was employed for the black hole mitigation and optimal deployment of base stations. This is due to its robustness, fast convergence and good optimization results.

One of the shortcomings associated with the standard BFOA is the constant step size which results in unnecessary chemotactic steps as the bacteria is close to the optimal values (Dasgupta *et al.*, 2009). With a large step size, the accuracy of the final solution is reduced because the bacteria are not able to reach the optimum location. Likewise, with a small step size, the convergence speed is affected because it will reach the optimum location but at a slow speed (Nasir *et al.*, 2015). However, since the BFOA is a meta-heuristic optimization technique, which is regeneration dependent, any constant parameter at the initial stage may have significant influence on the performance of the algorithm at the later stages. This led to the development of an improved BFOA with regards to the step size by some researchers (Dasgupta *et al.*, 2009: Supriyono & Tokhi, 2011; Nasir *et al.*, 2015;Niu *et al.*, 2015). The nature of their improvements was based on introduction of step size variation through adaptation of mathematical formulation and fuzzy logic approach by establishing a relationship between bacteria step size and nutrient value. However, linear value iteration based behaviour was introduced and discussed in this work to address the shortcoming of the standard BFOA.

1.2 Statement of Problem

Most routing protocols used in Wireless Sensor Networks do not consider message security due to resource constraints (Ngai *et al.*, 2006), which create opportunities for attackers. Among such attacks against wireless sensor network, a black hole attack has the ability to undermine the effectiveness of the network by partitioning the network to prevent useful information from reaching the base stations. Since successful delivery of data to the base station is more essential than prevention of data captured by an adversary because it can be countered with cryptographic techniques, several techniques based on secret sharing and multi-path routing were proposed in literature to overcome black hole attacks in the network but seems not to be very effective (Ketel *et al.*, 2005; Lou & Kwon, 2006). Thus, this research developed an algorithm to mitigate black hole attack in wireless sensor network based on modified BFO using optimized multiple base stations. Also the number of base stations was increased from four (4) to eight (8) to determine the increasing trend of packet delivery as the number of base stations is increased beyond four and to investigate the effectiveness of the modified BFOA as the complexity of wireless sensor network environment increases.

1.3 Motivation

Due to the distributed nature and deployment environment of WSNs, they are vulnerable to numerous security threats that can adversely affect their performance. Therefore to ensure the effectiveness and efficient functionality of WSNs, security is a major concern. The WSN mechanisms cannot at present ensure that an attack would not be launched and existing security mechanisms are inadequate. As such,there is need to develop new efficient and more reliable methods to protect the network from such attacks.

1.4 Aim and Objectives

The aim of this research is to develop an intelligent optimization based wireless sensor model capable of ensuring a successful packet delivery in the presence of black hole attack.

In order to achieve this aim, the following objectives were set:

1) Replication of the standard BFOA and development of the modified BFOA for the optimization of the multiple base station locations and detection of the black hole using MATLAB programming language.
2) Evaluation of the performance of the developed algorithm through simulation and comparison with existing algorithm to determine its effectiveness and accuracy in ensuring successful delivery of packets. Packet delivery success and failure, false positive and convergence speed were used as performance metrics.
3) Investigation of the effect of increasing the number of base stations from four (4) to eight (8) in order to find the trend of packet delivery as the number of base stations is increased beyond four (4).

1.5 Justification

Security attacks such as black hole poses great challenges to wireless sensor networks by hindering delivery of message to the destination. Since the sensor nodes that are responsible for sensing and forwarding the data to the base station for analysis and compilation can be compromised by attacks (such as: black hole, sink hole, hello flood, sybil, etc). There is need to optimally place the base stations to reduce the effect of these attacks. This has led to optimal placement of multiple base stations using robust intelligent meta-heuristic technique, which is capable of reducing the effect of attacks on WSNs. But the results of the mitigation technique can be improved to enhance accuracy and searching speed of the algorithm. Also, the number of base stations is a critical factor of the sensor network architecture that significantly affects the network performance. Therefore, the need to determine the optimal number of base stations for a particular coverage area.

1.6 Methodology

The methodology adopted in this research is as follows:

1) Initialization of BFOA parameters and the network parameters.
    a) Number of Bacteria
    b) Problem dimension
    c) Chemotaxis step
    d) Swarm limit
    e) Reproduction step
    f) Elimination probability
    g) Attractant
2) Replication of the standard BFOA using MATLAB programming language for:

    a) Chemotaxis and swarming step
    b) Reproduction step
    c) Elimination and dispersal step
3) Development of the modified BFOA using MATLAB programming language for:
    a) Adaptive step size
    b) Steps a) to c) of item 2) are implemented
4) Deployment of the randomly generated sensor nodes using the standard and modified BFOA for the four base stations.
5) Application of model developed in 3) using scenario of 4) to detect the black hole nodes in the network.
6) Performance evaluation of the developed model using Packet Delivery Success and Failure, False Positive and Convergence Speed as performance metrics.
7) Comparison of the developed algorithm with the Standard algorithm.
8) Increasing the base station topology to eight and deployment of the randomly generated nodes.
9) Application of model developed in 3) using scenario of 8) to detect the black hole nodes in the network and evaluating the delivery performance.

1.7 Dissertation Organization

The general introduction has been presented in Chapter One. The rest of the chapters are structured as follows: Firstly, a detailed review of related literature and relevant fundamental concepts about wireless sensor networks, its constraints and characteristics, security goals, types of attack in wireless sensor networks, black hole attack and standard BFOA are carried out in Chapter Two. Secondly, an indepth approach and relevant mathematical models describing the development of the modified bacterial foraging optimization algorithm are presented in Chapter Three. Thirdly, the analysis, performance and discussion of the result are shown in Chapter Four. Finally, conclusion and recommendations of further work makes up the Chapter Five. The list of cited references and MATLAB codes in the appendices are provided at the end of this dissertation.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

This chapter consists of a review on the fundamental concepts and the review of similar works, which are relevant to the scope of this research.

2.2 Review of Fundamental Concepts

In this section, concepts fundamental to the research such as wireless sensor networks, its constraints and characteristics, security goals, types of attack in wireless sensor networks, black hole attack, standard and modified BFOA amongst others are reviewed.

2.2.1 Wireless sensor networks

A Wireless Sensor Network is a network consisting of several low power and cost nodes deployed to monitor physical and environmental conditions like temperature, pressure, humidity etc., at different locations. Secure routing in sensor networks presents challenges due to several factors like low processing capability, limited memory, restricted bandwidth, and above all, limited energy of sensor nodes (Tubaishat *et al.*, 2004). Figure 2.1 illustrates an example of wireless sensor network comprising of sensor nodes, a base station and a remote server.

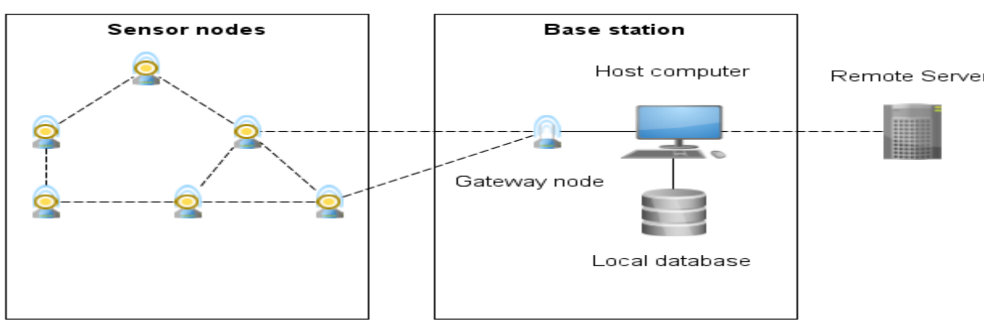

Figure 2.1: Typical Wireless Sensor Network (Michalik, 2013)

The sensor node gathers sensed data and forwards them wirelessly via other sensor nodes to a base station. The base station can store, visualize and analyse the sensed data, and then forward them to a remote server via the internet. In the context of Wireless Sensor Networks, a base station is an entry point to the wireless sensor networks whose primary goal is to gather sensed data from sensor nodes in wireless sensor network (Michalik, 2013).

In Wireless Sensor Networks, the routing protocols used are susceptible to routing attack that can disrupt the network connectivity. This is because most of them do not consider security due to resource constraints such as low processing power, limited memory and communication range (Martins & Guyennet, 2010).

The insertion of false information by a compromised node within the sensor network is a major cause of attack against security in the network. As such there is need to provide a means for detecting false reports (Pathan *et al.*, 2006). Some studies have suggested secure mechanisms such as cryptographic methods for protecting network, however they are mostly localized, or there is a high computation and also a requirement for time synchronization among the nodes (Salehi *et al.*, 2013). In the encryption scheme the transmission of extra bits is required which exhaust the memory, battery power and requires extra processing that are vital resources for the sensor nodes. It may also affect the quality of service parameters like delay, jitter and packet loss (Saleh & Khatib, 2005).

2.2.2 Characteristics and constraints of wireless sensor network

Some characteristics and constraints associated with wireless sensor networks are outlined as follows (Singh *et al.*, 2010):

1) **Dense sensor node deployment:** The sensor nodes are usually deployed densely.

2) **Battery-powered sensor nodes:** The sensor nodes are usually powered by battery and placed in a harsh environment where it is very difficult to recharge or change the batteries.
3) **Self-configurable:** The sensor nodes are usually randomly deployed and they configure autonomously into a communication network.
4) **Storage, computation and severe energy constraints:** Sensor nodes are having limited storage, energy and computation capabilities.
5) **Unreliable sensor nodes:** The deployment of the sensor nodes in harsh or hostile environment exposes them to physical damage or failures, thus making them unreliable.
6) **Application specific:** The sensor networks are application specific because they are designed and deployed to perform a specific task. The design requirements changes with its application.
7) **Many-to-one traffic pattern:** The network is considered as having a many-to-one traffic pattern because in most cases the data flow from multiple sensor nodes to a particular sink.
8) **Frequent topology change:** The topology of the network changes frequently due to node failure, energy depletion, addition or channel fading.

2.2.3 Security goals

The objectives for ensuring security in wireless sensor networks focus on the following security services (Kumar *et al.*, 2014; Pawar & Anuradha, 2015;Soni *et al.*, 2013).

1) **Confidentiality:** Ensure that data in transit is kept secret from eavesdroppers.
2) **Integrity:** Ensure that an adversary does not alter the received data in transit.
3) **Authentication:** Enables a sensor node to ensure the identity of the peer it is communicating with.
4) **Availability:** It is required that the service should always be available.

5) **Data freshness:** Ensure that old messages are not replayed. The data should be recent.

6) **Authorization:** It ensures that access to network services or resources is only through authorized node.

2.2.4 Types of attack in wireless sensor networks

In wireless sensor network, attacks can be broadly classified into two different levels: one is the attack against the security mechanism and the other against the basic mechanism like routing mechanism. Major attacks on wireless sensor networks are as follows (Singh & Malik, 2011; Soni *et al.*, 2013):

1) **Denial of Service (DoS):** In this type of attack, the resource of the victim node is exhausted by sending extra unnecessary packets, so as to prevent the network from accessing services. Thereby causing unintentional failure of the nodes. In wireless sensor network, several types of DoS attacks in different layers might be performed (Pathan *et al.*, 2006).

2) **Attacks on information in transit:** In wireless sensor network, the sensors are the main monitors. They monitor changes in parameter and immediately forward the report in accordance with the requirement. The information in transit can be altered, corrupted or varnished during forwarding report. Any attacker can monitor the flow of traffic and forge, intercept, change or interfere with the packets, as wireless sensor network is vulnerable to eavesdropping. Basically the nodes have limited transmission range. An attacker that has large processing capability and communication range has high tendency of attacking several sensors simultaneously (Kumar *et al.*, 2006).

3) **Sybil attack:** In this form of attack a single node forges the identity of multiple nodes. That is, multiple identities will be presented to the network by a malicious node. In geographic routing

protocols, this type of attack can be confusing because the adversary will appear to be in many locations at the same time (Soni *et al.*, 2013).

4) **Sinkhole attack:** In this form of attack all the traffic from a particular area are lured through a compromised node. The compromised node deceives other nodes with respect to the routing information and then alters the data or forward packets selectively (Karlof & Wagner, 2003).

5) **Hello flood attack:** In this form of attack, an attacker sends HELLO packets to a large number of sensor nodes, so as to convince them that they are neighbours (Pathan *et al.*, 2006). Therefore in trying to forward a packet to the base station, they forward through the attacker node, thereby ultimately spoofed by the attacker.

6) **Warmhole attack:** In this attack, packets at one location are recorded by an attacker and tunnelled to another location. The tunnelling could be done selectively. This attack is a serious threat to wireless sensor networks because it can take place at the initial phase during discovering of neighbourhood information by the sensor nodes (Pathan *et al.*, 2006; Singh & Malik, 2011).

7) **Blackhole attack:** In this attack, a black hole represents a malicious node, which attracts all the traffic in the sensor networks and then drops them instead of forwarding. The malicious node advertises that it has the shortest path to the destination. It advertises the presence of fresh routes irrespective of checking its routing table. Thus the Route Reply (RREP) from the malicious node will always reach the sender first as compared to other normal node that check their routing table entries for availability of fresh routes before replying to the sender of Route Request (RREQ) packet. This results in setup of a forged route between the sender and the attacker nodes. After acquiring the data packets, the attacker node then drops the packet (Arya & Raina, 2014; Kumar & Sangwan, 2015; Yadav *et al.,* 2012). This attack can be individual or

cooperative in nature (Kumar, 2015). In individual black hole attack, there is only one malicious node attack on the route. When the source node broadcast RREQ message, then the malicious node responds with high sequence number to its preceding node in the path. Thus the source node assumes malicious node as a destination node and start the process of data forwarding. The malicious node then drops all the packets received. In the case of a cooperative attack, the number of malicious node in the network is more than one. The overall result of a cooperative attack is a complete decrease in the throughput and increase in the packet drop ratio in the network.

This research focuses on cooperative black hole attack, which is a serious attack against wireless sensor networks because it hinders successful packet delivery to the destination by dropping all the packets (Kumar & Sangwan, 2015; Yadav *et al.*, 2012).

2.2.5 Bacterial foraging optimization algorithm (BFOA)

Bacterial foraging optimization algorithm is a new bionic optimization algorithm, which was proposed, based on foraging behaviour of *E. coli* found in human intestine (Passino, 2002). Foraging strategy is an optimization process of evolution. The assumption that animals search for and obtain nutrients in a way that maximizes their energy intake E per unit time T spent foraging is the basis of foraging theory (Passino, 2002).

The main idea of BFOA is mimicking the chemotactic movement of virtual bacteria in the problem search space (Das *et al.*, 2009). The individual bacterium communicates to others by sending signals. This technique is also inspired by the social foraging behaviour like ant colony and particle swarm optimization. It attracts the researchers due to its effectiveness in solving real world optimization problems and gives better results than conventional optimization methods of problem solving (Singh & Kaur, 2015). Figure 2.2 shows the trajectory movement of bacterial foraging algorithm.

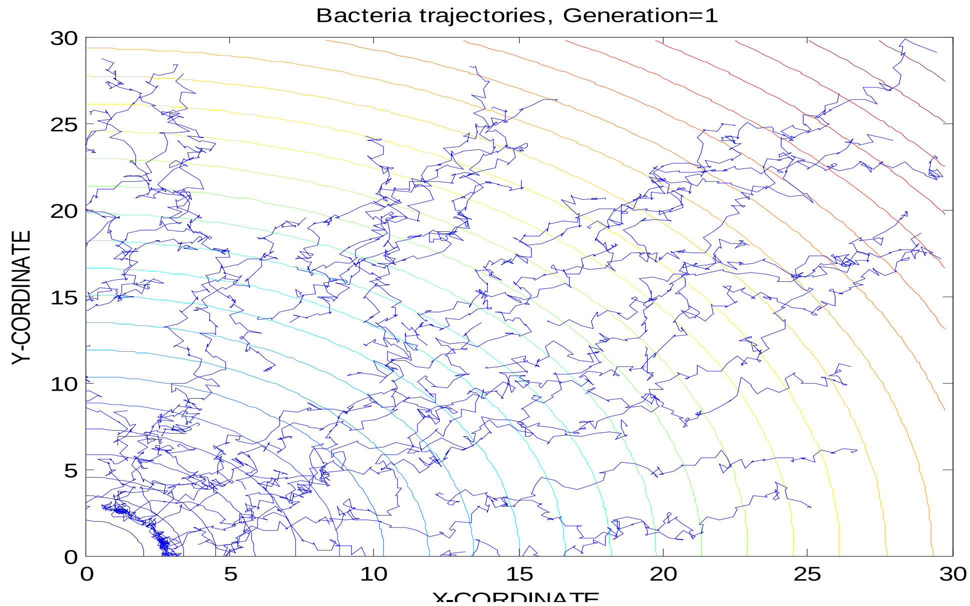

Figure 2.2: Trajectory Movement of E. Coli Bacterial

From Figure 2.2, it can be deduced that the trajectory movement of bacterial foraging optimization algorithm is similar to the routing path of sensor nodes deployment. Figure 2.2 was generated to enhance the visualization and understanding of the movement pattern of BFOA

a) Steps in BFO algorithm

The steps in Bacteria Foraging Optimization Algorithm are as follows:

1) **Chemotaxis:** This process simulates the movement of a bacterium *E. coli* cell through swimming and tumbling via a set of rigid flagella. The bacterium can move in two different ways (Dasgupta *et al.*, 2010). In the same direction, it can swim for a period of time or in an alternate direction during tumbling. The movement of the bacterium is represented by

$$\theta^i(j+1,k,l) = \theta^i(j,k,l) + C(i)\frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}} \quad i= \quad 1,2,\ldots\ldots\ldots.,S \qquad (2.1)$$

where:

jis the index for the chemotaxis step;

k is the index for the reproductive step;

l is the index for the elimination-dispersal step;

θi (j, k, l) represents the position of i-th bacterium at j-th chemotaxis step, k-th reproductive step and l-th elimination-dispersal step;

C (i) is the step size taken in the random direction;

andΔ indicate a vector in the random direction.

2) **Swarming:** During swarming, the bacterium congregates and moves in group towards location with high density nutrients. As a bacterium reaches the best location, it tries to attract other bacteria, so as to reach the desired location quickly. An attraction signal is produced for the communication between the cells in the *E. Coli* bacteria. Another repellant signal is also produced for noxious reserve. This process helps them to increase the bacterial density at the identified food position in the chemotaxis. The attraction signal is represented as follows:

$$J_{cc}(\theta,\theta(j,k,l)) = \sum_{i=1}^{S} J_{cc}(\theta,\theta^i(j,k,l)) = \sum_{i=1}^{S}\left[-d_{attrac\tan t}\exp\left(-\omega_{attrac\tan t}\sum_{m=1}^{p}(\theta_m-\theta_m^i)^2\right)\right] + \sum_{i=1}^{S} h_{repellant}\exp\left(-\omega_{repellant}\sum_{m=1}^{p}(\theta_m-\theta_m^i)^2\right) \qquad (2.2)$$

where:

J_{CC} (θ, P (j, k, l)) is the objective function value to be added to the actual objective function (to be minimized) to present a time varying objective function;

S is the total number of bacteria;

p is the number of variables to be optimized, which are present in each bacterium;

$d_{attractant}$is the depth of attractant signal released by a bacteria;

$w_{attractant}$ is the width of attractant signal;

$h_{repellant}$ is the height of repellant signals between bacterium;

$w_{repellant}$ is the weight of repellant signals between bacterium;

$\theta = [\theta_1, \theta_2 \ldots, \theta p]T$ is a point in the p-dimensional search domain;

and θm is the mth parameter of the global optimum parameter (Kandasamy *et al.*, 2014).

3) **Reproduction:** During reproduction, those bacterium that are least healthy die after some time while the healthier ones divide into two, and are placed in the same location. Thus, after reproduction the swarm size is maintained constant (Das *et al.*, 2009).

4) **Elimination and dispersal:** Sudden changes in the location where a bacterium population lives may occur due to various reasons e.g. abrupt temperature rise may kill a group of bacteria that are currently in a region with a high nutrient concentration (Das *et al.*, 2009). Events can disperse all the bacteria in a region. To simulate this phenomenon in BFOA some bacteria with a very small probability are eliminated and new bacteria is randomly initiated for replacement (Bermejo *et al.*, 2013).

In the standard BFOA, the chemotaxis step length C is a constant. This constant step size makes it difficult to balance between the global and local search ability thereby affecting the accuracy and searching speed (Niu *et al.*, 2015). Based on the standard BFOA given in (Das *et al.*, 2009) the flowchart for the BFOA is shown in Figure 2.3:

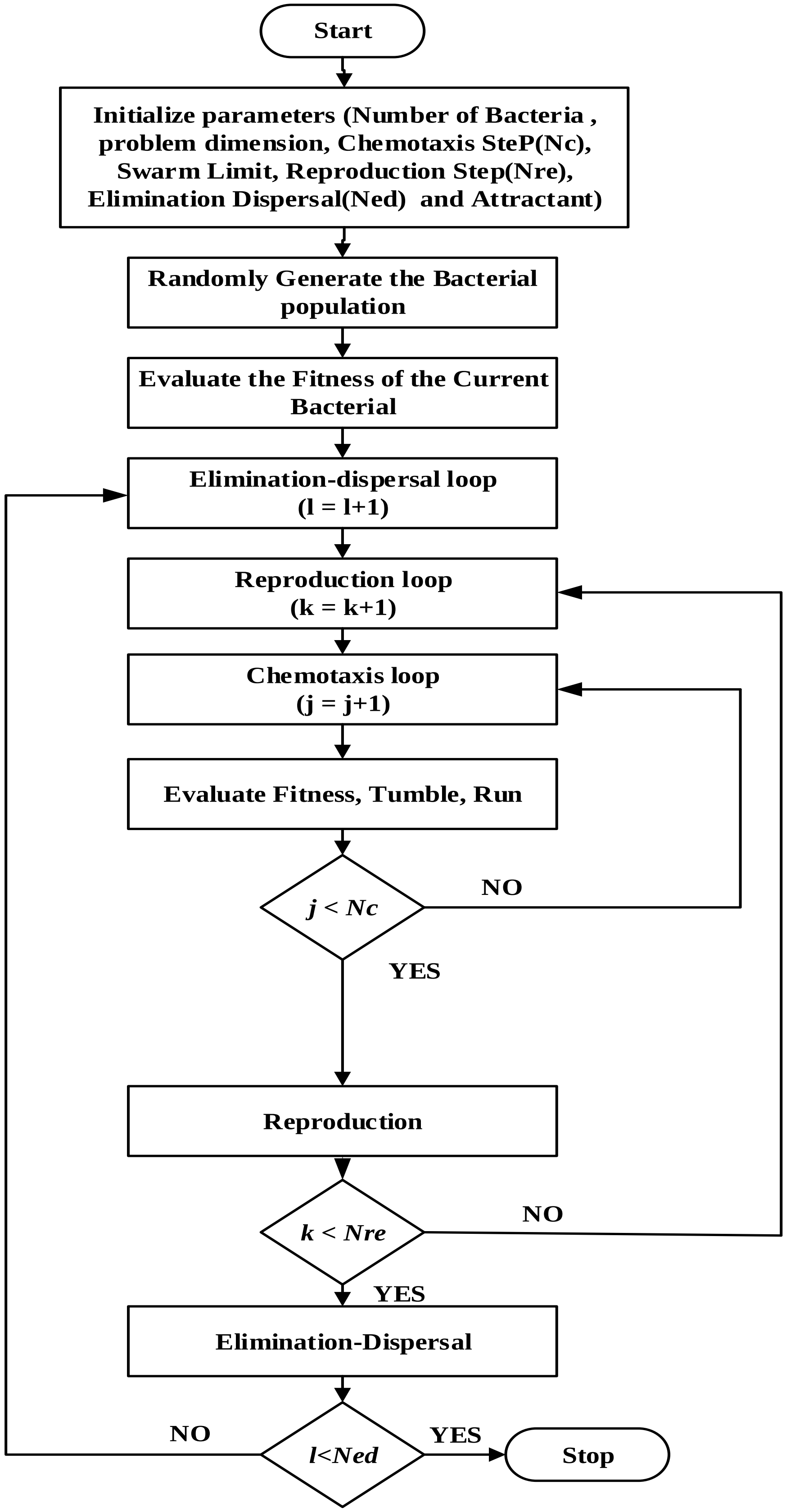
Start
Initialize parameters (Number of Bacteria , problem dimension, Chemotaxis SteP(Nc), Swarm Limit, Reproduction Step(Nre), Elimination Dispersal(Ned) and Attractant)
Randomly Generate the Bacterial population
Evaluate the Fitness of the Current Bacterial
Elimination-dispersal loop (l = l+1)
Reproduction loop (k = k+1)
Chemotaxis loop (j = j+1)
Evaluate Fitness, Tumble, Run
j < Nc
NO
YES
Reproduction
k < Nre
NO
YES
Elimination-Dispersal
NO
l<Ned
YES
Stop

Figure 2.3: Standard BFOA Flowchart (Das *et al*., 2009)

Figure 2.3 shows the flow chart implementation of the step-by-step procedure in BFOA given in section 2.2.5 a)

Where:

Nc is the number of chemotaxis step;

Nre is the number of reproductive step;

and Ned is the number of elimination-dispersal step.

2.2.6 Modified step size

The main shortcoming of the standard BFOA is the constant step size which results in unnecessary chemotactic steps as the bacteria is close to the optimal values (Dasgupta *et al.*, 2009). In order to avoid this, a refined variant was proposed by (Dasgupta *et al.*, 2009)in which the step size was adapted as follows:

$$C = \frac{|J(\theta)|}{|J(\theta)| + \lambda} = \frac{1}{1 + \lambda / |J(\theta)|} \qquad (2.3)$$

Where:

C is the step size;

λis a positive constant;

*θ*is the position of the bacterium;

and *J (θ)* represents the combined effects of attractants and repellants signals

Thus, when *J (θ)* is large the step size is also increased (C → 1), and once the bacteria approach the global optima the step size will be decreased. This increase and decrease in step size may also have significant influence on the performance of BFOA. This is because some optimization problemsmay require step size to be increased adaptively due to variation in search space.

Since BFOA is an evolutionary computational algorithm, constant parameters at the initial stage will have a significant influence on the performance of the algorithm. Therefore, iterative behavior based parameters need to be incorporated such that every generation can have its own suitable step size. In order to address this, a linear value iteration weight proposed for Fish swarm algorithmin Mu'azu et al. (2015)was incorporated to the adaptive step size given in equation (2.3), so as to determine the step size trends to local and global solution. The linear value iterative weight is given as (Mua'zu *et al.*, 2015):

$$w = \left(\frac{itr_{\max} - itr}{itr_{\max}} \right)^n \qquad (2.4)$$

Where:

w is the iterative weight;

$Itr_{\max}, Itr \text{ and } n$ are the total number of iteration, current iteration and total bacterium respectively.

In this research, the dimension of search space for the BFOA (P) is the WSN area of deployment and the total number of bacterial in the population (S) represent the total number of sensor nodes. This was used to develop the technique based on the foraging behaviour of BFOA.

2.2.7 Objective function formulation

For every swarm algorithm, a fitness value whose optimal value is to be determined needs to be formulated. Since the focus of this research is minimization of the effect of black hole attack and ensuring successful packet delivery through optimal base station placement, the objective function to be optimized was formulated based on the density of the nodes. Therefore for every bacterial (WSN nodes) deployed randomly in the network, the fitness value, which is based on the density of the nodes was determined. The step by step procedure of BFOA was performed and the fitness value for every generation was computed. The optimal value of this fitness was employed for the base station location. The fitness (objective function) was formulated as follows

$$Fitness = \sum_{i=1}^{n} D_i \qquad (2.5)$$

Where D_i is a logical vector, which is the Euclidean distance representing the density of nodes present within the neighbourhood. D_i was obtained using the following (Arya & Raina, 2014):

$$D_i = \sqrt{(X - L_i)^2 - (Y - L_j)^2} \qquad (2.6)$$

Where:

L_i and L_j represents the current and next bacterial sensor node location respectively;

X and Y are the coordinate locations of the nodes.

2.2.8 Performance metrics

Since the sensor nodes transmit data packet to the base station, packet delivery success has been employed as the most important metric by several researchers (Arya & Raina, 2014; Misra *et al.*, 2011)to evaluate the performance of wireless sensor network algorithms. The performance metrics to be used for evaluating the performance of the developed algorithm are:

(a) Packet Delivery Success: It represents the ability of the developed algorithm to correctly identify the black hole and ensure maximum throughput of data. If the packet delivery success probability is 1 (i.e. 100%), it signifies that all the attack nodes were successfully identified and maximum data packets were successfully delivered. This is probabilistic in nature as proposed in (Misra *et al.*, 2011). Assuming that the deployments of sensor nodes are uniform and random, the number of sensor nodes present in the black hole region is determined as follows:

$$n_b = \frac{n \times \pi \times R_b^2}{A} \tag{2.7}$$

Where:

n is the deployed sensor nodes

n_b is the number of sensor nodes in the black hole region

R_b is the radius of the black hole region

A is the coverage area

The probability that the path taken by the packet from the sensor node to one of the base station does not contain any black hole is determined using the following equation:

$$p_{ni} = 1 - p_{ni}^b \tag{2.8}$$

p_{ni}^b is the probability that at least one sensor node in the path is a black hole node, and is determined as follows:

$$p_{ni}^b = \frac{\binom{n_b}{1}\binom{n-n_b}{h_a-1} + \ldots + \binom{n_b}{\min\ n_b, h_a} \cdot \binom{n-n_b}{h_a - \min\ n_b, h_a}}{\binom{n}{h_a}} \tag{2.9}$$

From equation (2.9) if the path p_{ni} contains only one sensor node from the black hole, then this sensor node is chosen from the black hole nodes in $\binom{n_b}{1}$ way. The rest of the hop count nodes ($h_a - 1$) can be chosen in $\binom{n-n_b}{h_a-1}$ ways. This selection can be made in $\binom{n_b}{1} \cdot \binom{n-n_b}{h_a-1}$ ways.

Note:

h_a is the hop count, which constitute the black hole portion of the path between source and destination

n_b is the number of black hole nodes.

Similarly, since each path taken by the sensor node to the base station $B_i \in B$ is independent of all the other paths, then the event of individual paths p_{ni} , $i \mid B_i \in B$ having at least one sensor node from the black hole is independent of the same event for another path p_{nk} , $k \mid B_k \in B \setminus B_i$. Therefore, the combined probability of the individual path having at least one black hole node is given by:

$$P_{bh} = \prod_{B_i \in B} P_{ni}^{b} \tag{2.10}$$

Hence, the probability that at least one path from sensor node to the base station ($B_i \in B$) does not contain any black hole node is as follows:

$$P_{ni} = 1 - P_{bh} \tag{2.11}$$

Where:

P_{ni} is the probability of the path taken by the packet.

P_{ni}^{b} is the probability that at least one sensor node in the path is a black hole.

Note that each path p_{ni} originating from the sensor node n to the base station $B_i \in B_n$ does not depend on all other paths. Similarly, the event of each path p_{ni}, $i \mid B_i \in B_n$ having one sensor nodes from the black hole does not depend on the same event for another path p_{nk}, $k \mid B_k \in B_n \mid B_i$.

(b) False Positive: False positive is used to determine the inability of the model to determine the existence of an intruder. If N_i denote a set of sensor nodes identified by a base station B_i as black hole nodes, then all the sensor nodes are initially added to the set N_i. If B_i receives a data packet through a particular path p_n, then all the sensor node in the path to B_i are removed from the set N_i since they are involved in data forwarding. After a total time of T seconds, the base station create a global black hole set using the following

$$N = \bigcap_{i=1}^{|B|} N_i \qquad (2.12)$$

Equation (2.12) represents the sensor nodes that none of the base stations got any data packets from.

(c) Packet Delivery Failure: This represents the inability of the algorithm to correctly identify the black hole nodes, thereby reducing the probability of the packet getting to the base station. Assuming the packet delivery success probability is denoted as P_{ds}, then the packet delivery failure probability is determined as follows:

$$P_{df} = 1 - P_{ds} \qquad (2.13)$$

Where P_{df} denotes the packet delivery failure.

(d) Convergence Speed: Convergence speed is used to determine how fast and efficient the packets are delivered to its destinations. This will be determined using the computational energy of the algorithm. For example, the time it takes the packet to be delivered from its source to its destination.

2.3 Review of Similar Works

Considerable work has been done by many researchers to develop an algorithm capable of detecting and preventing black hole attack in wireless sensor networks. Black hole attack has been studied in the wired network, agent based network, mobile ad hoc network (MANET) and wireless sensor networks. Due to high computation and storage requirements for the mechanisms applied for black hole attack in non-wireless sensor networks, it does not apply to the black hole problems in wireless sensor networks (Sheela *et al.*, 2012). The significant works, which are directly related to this research,are reviewed as follows:

Karakehayov (2005)proposed a technique for black hole attack detection in wireless sensor networks. In this method, the transmitting sensor node transmits a packet to several sensor nodes towards the direction of the base station. If a sensor node fails to forward a packet, its next hop neighbour identifies the incidence and reports the sensor node as a black hole. However, the method was expensive for a network with large number of black hole nodes because each original message requires extra messages. Also a technique to eliminate the black hole after its detection was not considered.

Lou & Kwon (2006)developed a scheme for reliable and secure collection of data in wireless sensor networks based on secret-sharing and multi-path routing. The scheme was based on a distributed N-to-1 multipath discovery protocol, which periodically or on demand initiated a route update at the base station. For every sensor node, a set of node-disjoint paths back to the base station was found at the end of each discovery process. Then based on the availability of the multiple paths at each node, they proposed a hybrid multipath scheme for secure and reliable collection of data. However the scheme considered only one base station and has several setbacks because strategic position of the black hole nodes was not considered and also routes are prone to black hole attack when directed towards a single base station.

Atakli *et al* (2008) proposed a novel scheme based on weighted-trust evaluation to detect malicious nodes in wireless sensor networks. In this scheme, forwarding nodes gave trust values to each of the nodes in the cluster. When a node has been compromised, the forwarding node then lowers the node's trust value. Through simulation, the correctness and efficiency of the detection scheme were verified. However, the type of attack that led to the node being compromised was not specified and the approach was based on the assumption that majority of the sensor nodes are working properly, which is usually not so in all cases.

Shu *et al* (2010) proposed an enhanced technique Multicast Tree Assisted Random Propagation (MTRP) to mitigate black hole attacks in wireless sensor network based on multi-path routing. It was considered more effective and instead of using deterministic multi-path routes to transmit data from the sensor nodes to the base stations, it proposed the use of randomized routes. Multiple paths were computed in a randomized way, each time a packet is to be sent, thereforethe routes were as dispersive as possible, thereby reducing chances of passing through a black hole. But the technique was only effective if there were a small number of black holes in the network. Also a single base station was used and routes are prone to black hole attack when directed towards a single base station.

Misra *et al (*2011) designed an efficient technique namely BAMBi (Black hole Attacks Mitigation with Multiple Base stations in wireless sensor networks) that uses multiple base stations to counter the impact of black holes on data transmission in the network. In this technique, the base stations were deployed at fixed location. The approach was very effective and required less computation and message exchanges, thereby saving the energy of the sensor nodes. With one base station it had high false positive value because many non-black hole nodes that were not able to reach the base station as a result of black hole nodes disruption were identified as black hole nodes. The false positive value reduced with two or more base stations. However, the optimal number of base stations and their positions were not identified.

Sheela & Mahadevan (2012) developed a technique to defend against cloning attack, sink hole attack and black hole attack in wireless sensor networks using mobile agents with several base stations. The role of mobile agent was to prepare and update the global information table at every sensor node. Routing through multiple base stations algorithm was activated only when there was a chance of black hole attack. Mobile agents find the probability of the presence of black hole nodes. The technique had low communication overhead and the performance of the algorithm was examined through simulation. However, the scheme did not consider optimizing the base station, hence the computational efficiency of the scheme was very low.

Sheela *et al* (2012) proposed a secure routing algorithm against black hole attacks with four base stations deployed in the network by using mobile agents. In this technique, routing through multiple base stations algorithm was activated only when there was a chance of black hole attack in the network, which is detected by mobile agent. If it doubted that a particular node was a black hole node, it then triggered the routing process algorithm through multiple base stations for a given time. If the node was a black hole node, then it revokes the node and triggers a routing process algorithm through the nearest base station. The technique was efficient and ensures high data rate delivery. However, the use of mobile agents to detect the probability of presence of black hole nodes in the network might be prone to error, because the current positions of the black hole might be outside the range of nodes.

Dighe & Vaidya (2013) developed a technique to effectively mitigate the adverse effects of black hole attacks on wireless sensor networks using multiple base stations. This technique was based on the deployment of multiple base stations in the network and routing of copies of data packets to the base stations. The technique was highly effective and required very little computation. However, the position of the base stations was not optimized.

Gondwal & Diwaker (2013) proposed a technique to detect black hole attack using check agent with four base stations. The technique reduced the extra overhead from the network as check agents played a vital role in the detection of black hole in the network. Also the technique was fast, energy efficient and reduces message complexity, as it was a software program implemented using java language. It was designed to move from node to node checking the presence of black hole nodes in the network. The results were compared with and without the check agents. However, the base stations position was not optimized.

Arya & Raina (2014) presented an efficient technique to counter the effect of black hole in data transmission using BFOA based optimized multiple base stations deployed randomly in the network. Simulation result showed that their proposed method increased the packet delivery success with a significantly reduced false positive. However, the oscillatory effect of step size associated with BFOA and eliminating the black hole after its detection was not considered.

Sharma *et al (*2014) developed a mechanism for black hole attack detection in LEACH based wireless sensor networks. This mechanism created clusters from the sensor nodes on the basis of signal strength and the protocol elects the cluster head for each round. Identification of the cluster head was maintained by the base stations at each round and a repeated election of a cluster head meant the network was under a black hole attack that triggers the base station to send alert packets to the sensor nodes. A successful data transmission was only possible if no repeated election occurred. The drawback of this model was its inability to detect sensor nodes as a black hole node but could tell if the cluster head was the black hole node or not. Therefore, prevention of the black wireless sensor network nodes was not considered.

Tomar & Tak (2014) designed an algorithm that prevent black hole attack and successfully deliver data by optimizing the position of multiple base stations using genetic algorithm. In this technique, genetic algorithm is applied to the number of base stations to calculate the optimized position of the base station after dividing the area into parts equal to number of base stations. Fitness value computation was then carried out and those with the highest value were selected as the optimized position. This approach achieved high data delivery rate with large number of base stations, but did not identify the number of black hole nodes in the network.

Kalaiselvan & Singh (2015)designed a method to detect and isolate black hole attack in wireless sensor networks. Their approach was to identify the black hole node by monitoring fake packet replies that are being transmitted by the nodes and then discarded from the network. Sensor nodes were deployed in the field randomly and the sensor network performance was analysed for the presence of black hole nodes. Black hole nodes are present if the network performance was lower than the threshold. Otherwise it is not. Route request messages are broadcasted to the other node in the network by the sensor node. It then checks for reply packets to identify the black hole node and broadcast to other nodes that a particular node is the black hole node. This results to complete isolation of the black hole node from the network and all reply messages from it are simply discarded by other sensor nodes. However, this technique only considered a single base station and did not detect collaborative black hole attack.

It is obvious from literature reviewed that a lot of research work has been done on the area of black hole mitigation in wireless sensor networks. Successfully delivering packet from the source to the destination without any loss of data still calls for more research in wireless sensor networks. In order to address some of the issues associated with most of the reviewed literature, this research developed black hole detection mechanism in wireless sensor network using a modified BFOA. The modified BFOA is expected to overcome the oscillatory effect of step size associated with the standard BFOA with a view to improving the packet delivery success whilst reducing the false positive and the packet delivery failure rates. The number of base stations was increased from four to eight to investigate the increasing trend of packet delivery as the number of base stations was increased beyond four and its efficiency as the complexity of wireless sensor environment increases.

CHAPTER THREE

MATERIALS AND METHODS

3.1 Introduction

In this chapter, the methods, materials and procedures employed for the successful completion of this research are discussed. The standard BFOA and the modified BFOA were replicated and developed respectively.

3.2 Initialization of BFOA and Network Parameters

As discussed in subsection 2.2.5, the performance of BFOA depends on the appropriate selection of its control parameters (swarm length, chemotaxis step, reproduction step, attractant condition, elimination step and elimination probability). The appropriate values selected for these parameters are presented in Table 3.1

Table 3.1: BFOA Simulation Parameters

S/No	Simulation Parameter	Value
1	Population	200
2	Length of Swarm	4
3	Chemotaxis Step	100
4	Reproduction Step	4
5	Elimination-Dispersal Step	2
6	Elimination Probability	0.25
7	Attractant Condition	2
8	Number of Iteration	1000

The simulation parameters presented in Table 3.1 were selected because BFOA has been reported to perform optimally within this range of values (Passino, 2002). The MATLAB script for the modified BFOA algorithm can be found in Appendix A.

The network model used in this research consists of a number of randomly generated sensor nodes and a set of base stations. The sensor nodes were considered as bacterial sensors, performing the foraging behaviours of BFOA described in subsection 2.2.5 a) The network consists of a randomly deployed sensor nodes $S = \{S_1, S_2 \dots, S_N\}$ and a set of base stations $B = \{b_1, b_2, \dots, b_m\}$, which are connected to each other and are stronger than the sensor nodes. The sensor nodes were deployed randomly in a square coverage of dimension $100m \times 100m$ and each sensor node has a communication range of $20m$. Table 3.2 shows the detailed description of the network model settings.

Table 3.2: Assumed Network Parameters for Simulation

S/No	Parameters	Value
1	Number of Sensor nodes	200
2	Transmission Range	20m
3	Coverage Area	$100m \times 100m$
4	Radius of Black hole Region	20m, 30m & 40m
5	Black hole position	Random

To ensure efficient packet delivery in the presence of black hole, the following model assumptions were made:

1) The network consists of randomly deployed sensor nodes and base stations.
2) The base stations are more powerful than the sensor nodes and are connected to a replenished power.
3) The sensors are deployed in a square field area of $10000m^2$
4) The sensor nodes are connected to each other and the density of the wireless sensor network is

high enough to allow adequate connectivity.

5) The sensor nodes have a maximum transmission range of *20m.*

6) When the base stations are communicating with the sensor nodes, they are assumed to have the same transmission range in order to ensure adequate symmetric communication.

7) The data information control packet was assumed to be encrypted.

3.3 Standard Bacterial Foraging Optimization Algorithm

In this research, the standard BFOA was first replicated as described in Figure 2.3. The Chemotaxis, Reproduction and Elimination implementations are described as follows.

3.3.1 Chemotaxis and swarming Step

In the chemotaxis and swarm movement, the flagella are initialized and the number of chemotactic movement length is also initialized. The bacterium movement represented in equation (2.1) is then simulated through the cell-to-cell attractant signalling. A snippet of the MATLAB script for the cell to cell attractant is given as follows;

```
function [Jar]=attract_func(x,theta,S,flag)
If flag==2 % Test to see if main program indicated cell-cell attraction
    Jar=0;
return
end
```

The complete MATLAB script is given in Appendix B2.

3.3.2 Reproduction step

As discussed in section 2.2.5 a), the healthier bacterium is split asexually into two children bacterium in the same location. This was done to maintain constant size of the swarm. The reproduction step is implemented as described in subsection 2.2.5.

3.3.3 Elimination and dispersal step

In order to simulate BFOA effectively, some unhealthy bacteria are eliminated with an elimination probability of 0.25 while a new bacterial is randomly generated over the search space as a replacement. The elimination probability of 0.25 was chosen in this research as stated in Table 3.1. The pseudo code implementation of the BFOA is given as follows:

```
Start BFOA
initializeparameters (population, dimension, chemotaxis step, reproduction step, elimination
step and number of iteration)
randomlygenerate bacterial position
eliminationloop (l=l+1)
reproductionloop (k=k+1)
chemotaxisloop (j=j+1)
fori=1:S
computefitness function
simulateswarming and tumble; generate a new bacterium
if j<Nc%Nc is the Chemotaxis step
goto chemotaxis step%Continue chemotaxis step until bacterial life is over
end
fori=1:S
evaluatefitness function
performreproduction (Discard the worst and split the best result into two)
if k<Nre%Nre is the Reproduction step
goto reproduction step%Continue reproduction step until bacterial life is over
end
fori=1:S
        Evaluate fitness
eliminateand disperse each bacterium
if l<Ned%Ned is the Elimination-dispersal step
goto elimination and dispersal step%Continue elimination and dispersal step until bacterial life is
over
end
end
end
```

end
End BFOA

3.4 Development of the Modified Bacterial Foraging Algorithm

As stated earlier, the main shortcoming of the standard BFOA is the constant step size, which results in unnecessary chemotactic steps as the bacteria moves closer to the optimal value. In this research, a linear value iteration weight step size given in equation (3.1) was introduced in order to improve the convergence speed and optimal performance of the standard BFOA.

$$C = \frac{1}{1+ \lambda / \left|J(\theta)\right|} \times \left(\frac{itr_{\max} - itr}{itr_{\max}} \right)^{n} \tag{3.1}$$

Based on equation (3.1) the solution search movement of the BFOA given in equation (2.1) is modified as follows:

$$\theta_i(j+1, k, l) = \theta_i(j,k,l) + \frac{\left[\frac{1}{1+ \lambda / \left|J(\theta)\right|} \times \left(\frac{itr_{\max} - itr}{itr_{\max}} \right)^{n} \right](i)\,\Delta(i)}{\sqrt{\Delta^{T}(i)\,\Delta(i)}} \tag{3.2}$$

Where, Δ is a randomly generated vector in the range of $[-1,1]$.

λ is a positive constant

$J(\theta)$ represents the combined effects of attractant and repellants and θ is the position of the bacterium.

$Itr_{\max}, Itr$ *and* n are the total number of iteration, current iteration and total bacteria respectively.

Equation (3.1) is expected to overcome the oscillatory effect of the step size and improve the precision of the BFOA with respect to convergence speed and optimization efficiency. The pseudo code description of the modified algorithm is given as follows:

```
%Pseudo Code Representation BFOA
Start
Initialize all parameters
Generate Initial bacterial randomly
Evaluate the fitness of each bacterium
While iteration
for (Elimination & dispersal loop)
for (Reproduction loop)
for (Chemotaxis loop)
performTumble,Run
evaluate fitness
end%Chemotaxis loop
end%Reproduction loop
end%Elimination& dispersal loop
select adaptive step size
end
Update & output
Stop
```

Based on the pseudo code given above, the flow chart implementation is presented in Figure 3.1:

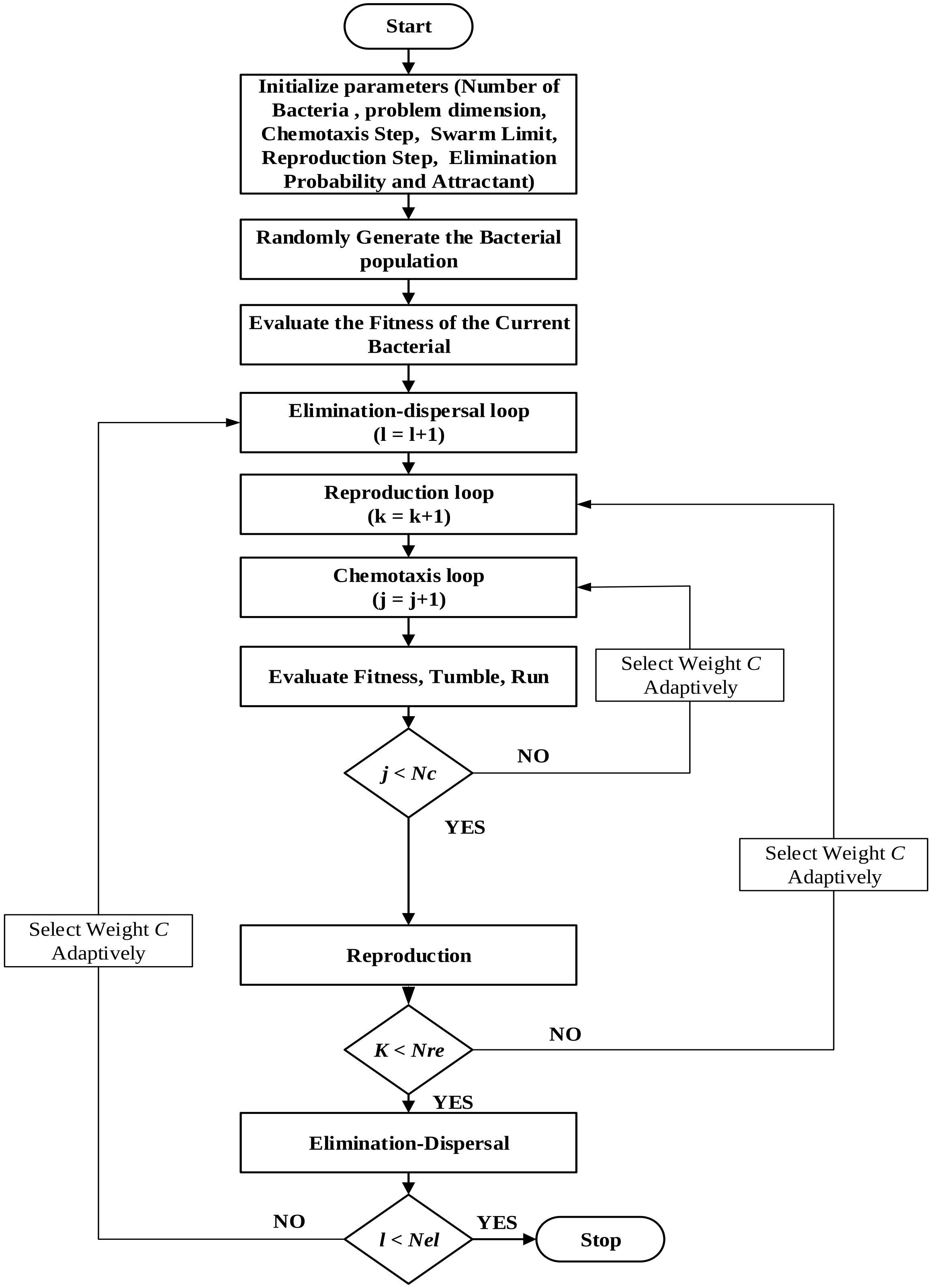

Figure 3.1: Modified BFOA Flowchart

Figure 3.1 shows the flowchart for the modified BFOA. From Figure 3.1, it can be observed that the step size of the BFOA is selected adaptively as the chemotaxis, reproduction and elimination - dispersal loops through the simulation process. This is expected to overcome the unnecessary oscillation of the standard BFOA given in Figure 2.2.

3.5 Optimized Positioning of Base Station and Detection of Black Hole

For the set of $S = \{S_1, S_2 \dots, S_N\}$ bacterial nodes, the fitness value $Y = f(S)$ were computed. The BFOA step by step approach presented in subsection 2.2.5 a) were implemented according to the modified step size (3.2) and the attractant signal represented in equation (2.2), and then the fitness of the new bacterial node location was evaluated. The fitness (objective function) was formulated using equations (2.5) and (2.6).If the current (new) location of the bacterial sensor nodes are within the search limits of the base station location, then the new location are stored as $S_i^{(t+1)}$ otherwise the previous location stored as S_i^{t} are considered. The fitness value of the current bacterial nodes stored in memory is evaluated and the selection of the best sensor node is done based on the fitness values. The whole process repeats itself through the total number of iterations until the remaining parent bacterial nodes are exhausted. At the termination of the algorithm, the bacterial node with the best fitness value is chosen as the best location for the deployment of the base station. The step-by-step algorithmic pseudo code implementation of the proposed model is given as follows:

```
Start%Start the Algorithm
%Pseudo code Representation of the Developed Technique
Initialize all parameters
Randomly generate the initial positions (Xi) of the bacterial sensor nodes
while iteration
fori=1:number of nodes
    Evaluate the Fitness (Yi)
end%End for
cleari
fori=1:total number of iteration
```

```
for j=1:number of nodes
        Select a new position (Xj) of the bacterial node
based on the modified bacterial attractant signal
ifXj>x-coordinate search limit &&Xj<y-coordinate search limit
          Select Xj (New Position) and update the memory
else
          Select Xi (Previous Position) and update the memory
end%End if
       Evaluate the Fitness (Yj)
End%End for
fori=1:number of nodes
if Fitness (Yj)>Fitness (Yi)
          Replace Xi with Xj
          Replace Yi with Yj
end%End if
end%End for
end%End for
    Select the position that correspond to the best fitness
end%End While
end%End Algorithm
```

Appendix B1 presents the MATLAB script used for the objective formulation.

3.6 Performance Evaluation

The performance of the developed standard and modified Bacterial Foraging Optimization algorithm was evaluated using performance metrics listed in subsection 2.2.8

3.6.1Packet delivery success and failure

Since the sensor nodes are transmitting data packet to all the base stations, more than one copy of data may reach the base stations, which implies successful packet delivery. This means that the sensor node path should not pass through the black hole region. This is achieved using the BFOA as discussed previously. The probability that the path taken by the packet from the sensor node to the base station does not contain any black hole, which implies successful packet delivery, was determined using equation (2.11). Otherwise, it is known as packet delivery failure and is obtained using equation (2.13).

3.6.2False positive

The base stations create a global black hole set of all the nodes that were not involved in data forwarding using equation (2.12). That is, those sensor nodes that did not forward data packet to any base station. However, there is possibility that non-black hole node may be identified as black hole due to base stations not receiving packets as a result of the sensor node being obstructed by black hole region. Though this is expected to be very minimal since the position of the base station was determined through optimal deployment of the sensor networks using BFOA.

3.6.3 Convergence speed

The MATLAB command *tic* which record the beginning time of a running program and the MATLAB command *toc*, which record the end time of a running program, were used. The difference between the *tic* and *toc* value was used to determine the computational time of the model.

3.7 Comparison of the Results obtained fromthe Developed Modified BFOA with the Results of the Standard BFOA

The results obtained from the developed modified algorithm were compared with the results of the standard BFOA used in the work of (Arya & Raina, 2014). The MATLAB code for the comparison is found in Appendix D.

3.8 Extended Eight Base Stations

To investigate the increasing trend of packet delivery as the number of base stations goes beyond four, the number of base stations was increased to eight and they were optimally positioned using the developed modified BFOA. The MATLAB code for the extended base stations is found in Appendix B3.

CHAPTER FOUR

RESULTS AND DISCUSSION

4.1 Introduction

In this section, the performance of the standard BFOA and the modified BFOA were evaluated using performance metrics: packet delivery success and failure, false positive and convergence speed. The results obtained for the increase in number of base stations are also discussed.

4.2 Packet Delivery SuccessResults of Modified BFOA

The results for the percentage of packets successfully delivered to the base station using one, two, three or four base stations are shown in Figure 4.1.

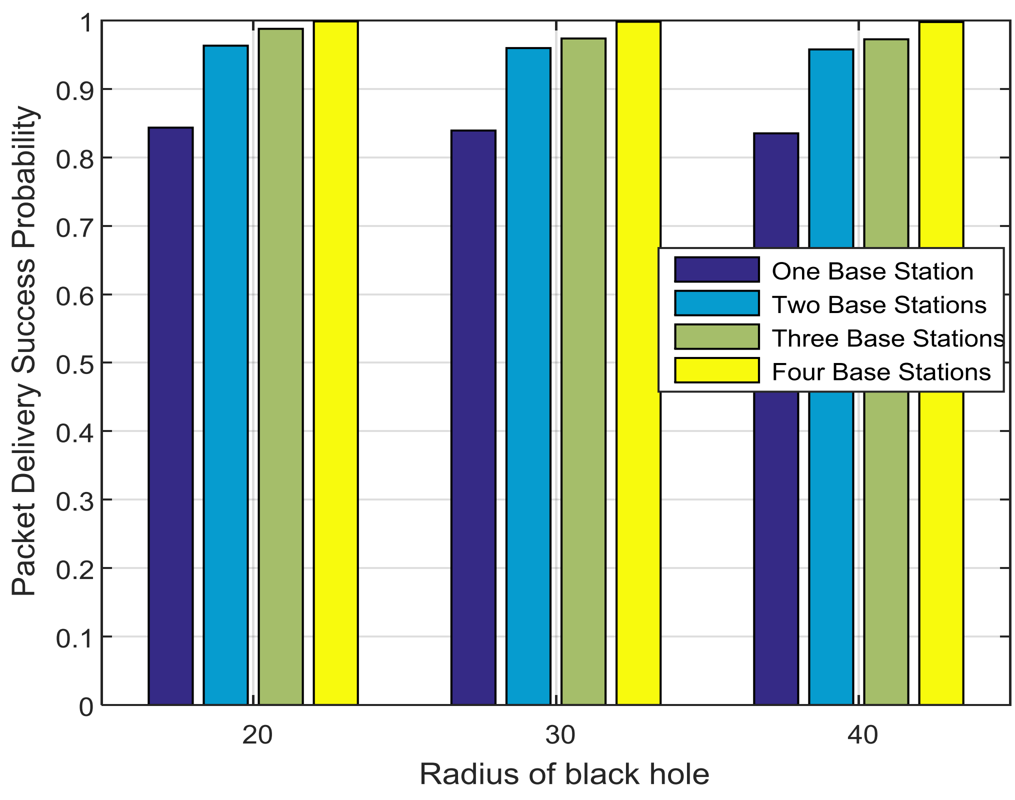

Figure 4.1: Packet Delivery Success

Figure 4.1 shows a slightly linear increase in the percentage of packet successfully delivered to the base station in the network. In general, the performance of the technique increases as the number of base station was increased. Packet delivery success of 84.35%, 96.33%, 98.79% and 99.87% was attained for one, two, three and four base stations respectively when the radius of black hole region was 20*m*. When the radius of black hole region was increased to 30*m*, the packet delivery success was 83.93%, 95.98%, 97.38% and 99.82% for one, two, three and four base stations respectively. Similarly, when the radius of black hole region was 40*m*, the packet delivery success was 83.52%, 95.78%, 97.26% and 99.78% for one, two, three and four base stations respectively. This indicates that increasing the radius of black hole region from 20*m* to 30*m* and 40*m*, the performance of the technique slightly decreases with respect to packet delivery success as shown in Figure 4.2.

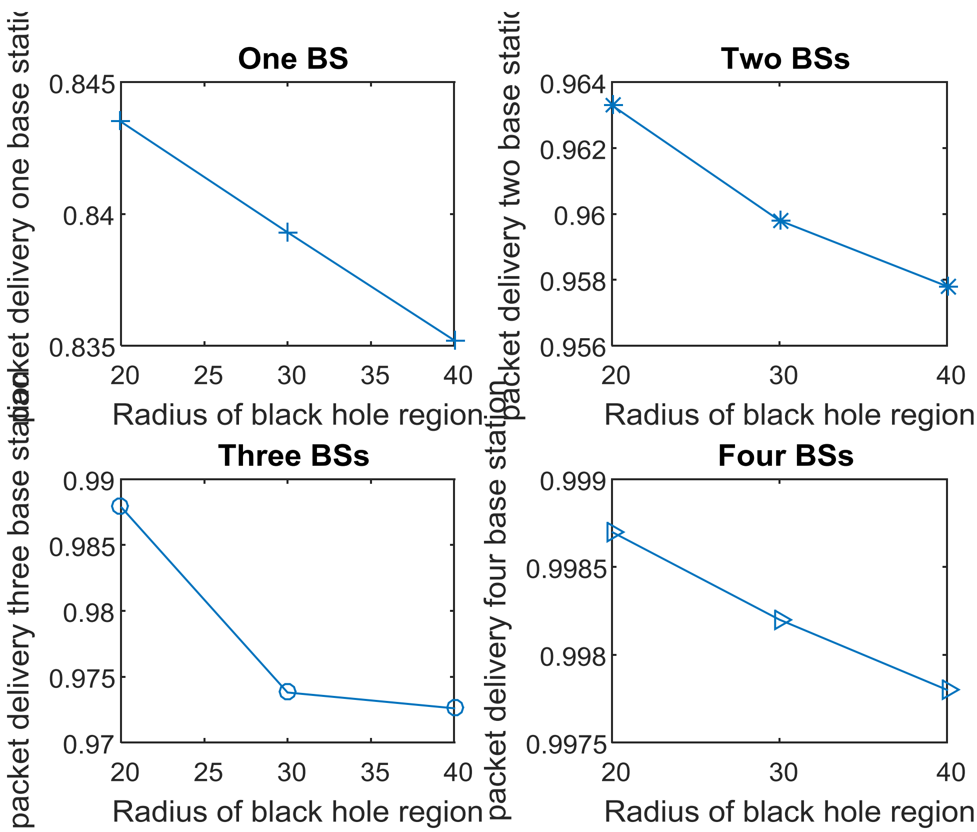

Figure 4.2: Relationship between Packet Delivery and Radius of Black Hole

As expected, the packet delivery success decreased when the radius of the black hole region becomes larger. However, a successful delivery probability of 99.78% was achieved when the position of the four base stations deployed in the network were optimized with 40*m* radius of black hole region. For three, two and one base stations, a packet delivery probability of 97.26%, 95.78% and 83.52% was achieved for black hole region with radius of 40*m*. This indicates that the success rate of packet delivery reduces slightly with an increase in the radius of black hole region for models with lesser number of base stations.

4.3 Packet Delivery Failure Results of Modified BFOA

As stated earlier, packet delivery failure represents the inability of the algorithm to correctly identify the black hole nodes, thereby reducing the probability of the packet getting to the base station. The bar chart showing the packet delivery failure for the entire base station configuration is presented in Figure 4.3.

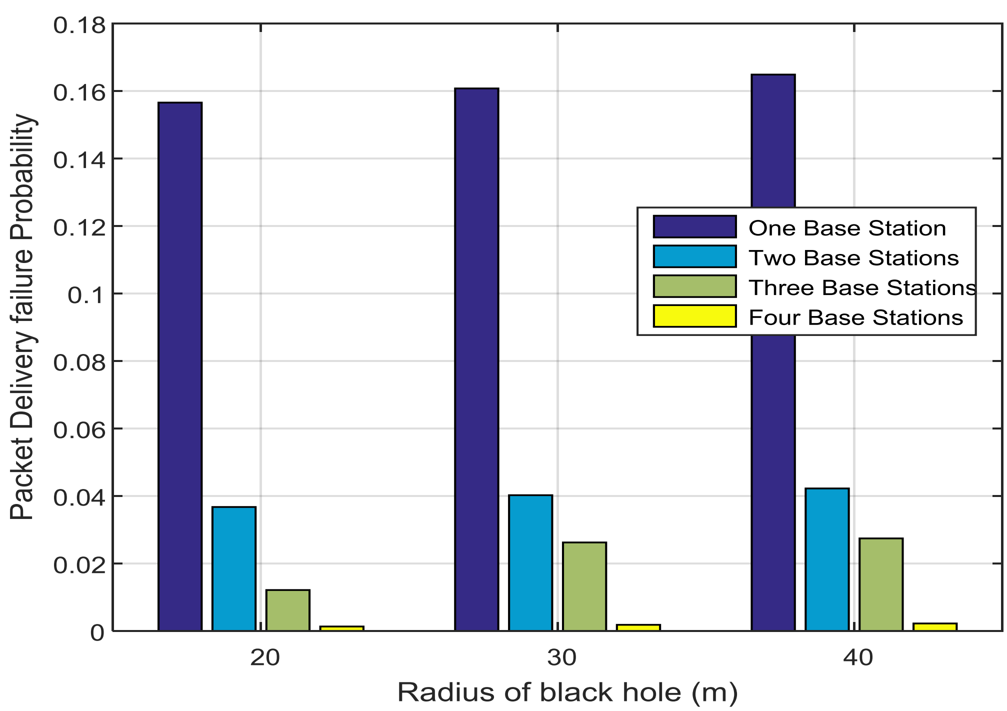

Figure 4.3: Packet Delivery Failure

Figure 4.3 illustrates the percentage of packet delivery failure of the transmitted packet. As expected, when the radius of the black hole region was increased, the packet delivery failure increased. Packet delivery failure of 15.65%, 3.67%, 1.21% and 0.13% was obtained for one, two, three and four base stations respectively when the radius of black hole region was 20*m*. When the radius of black hole region was increased to 30*m*, the packet delivery failure was 16.07%, 4.02%, 2.62% and 0.18% for one, two, three and four base stations respectively. Similarly, when the radius of black hole region was 40*m*, the packet delivery failure was 16.48%, 4.22%, 2.74% and 0.22% for one, two, three and four base stations respectively. It was observed that when the number of base station was increased to four (4) and the radius of black hole region was increased to 40*m*, the percentage failure was less than 1%. But when a single base station was deployed in a strategic position with black hole radius of 40*m*, packet delivery failure percentage observed was more than 16%. This indicates that using multiple base stations packet delivery failure may be eliminated completely even in the presence of black hole attack using the developed technique. The packet delivery failure is generally high with an increase in the black hole region with one base station. This is due to the routing path of the one base station, which may be within the region of the black hole radius.

4.4 False Positive Results of Modified BFOA

The bar chart showing the inability of the developed technique to detect the presence of black hole nodes is shown in Figure 4.4.

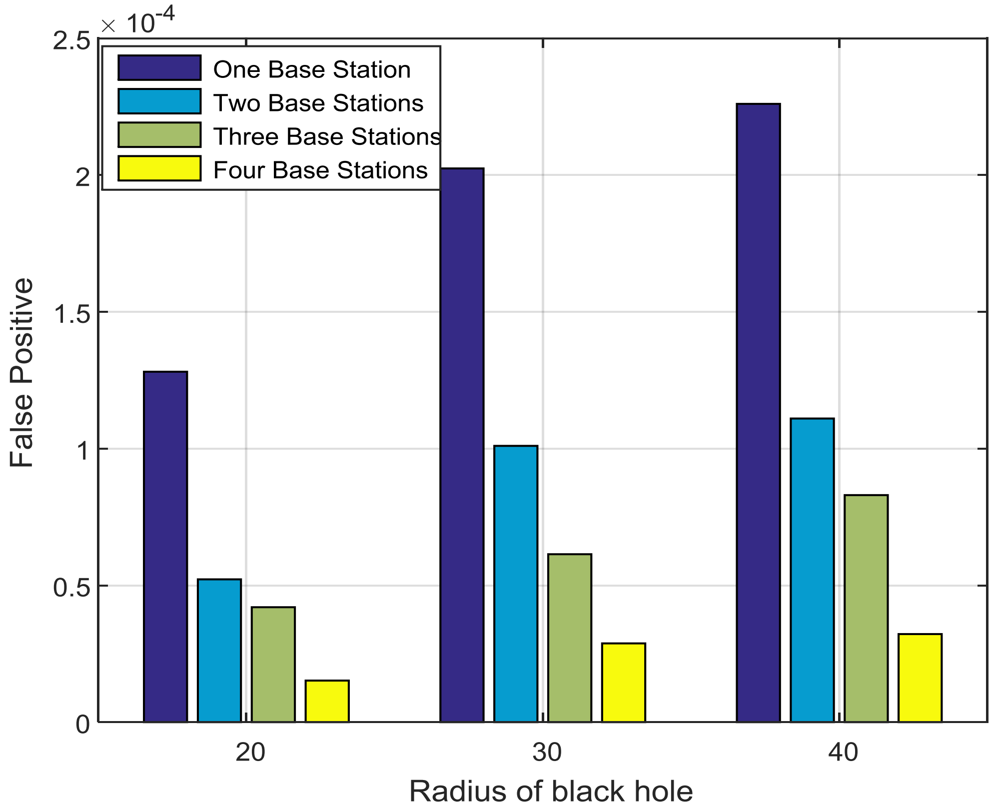

Figure 4.4: False Positive

Figure 4.4 shows the percentage of false positive. False positive value of 0.0128%, 0.0052%, 0.0042% and 0.0015% was obtained for one, two, three and four base stations respectively when the radius of black hole region was 20*m*. When the radius of black hole region was increased to 30*m*, false positive value of 0.0202%, 0.0101%, 0.0061% and 0.0029% was obtained for one, two, three and four base stations respectively. Similarly, when the radius of black hole region was 40*m*, the false positive value was 0.0226%, 0.0111%, 0.0083% and 0.0032% for one, two, three and four base stations respectively. It was observed that the percentage of false positive is generally higher for one base station. This is because a large number of sensor nodes that are unable to reach the base station due to obstruction by the black hole region are usually identified as black hole. Significant reduction in false positive was observed when the number of base stations was increased. With a 40*m* radius of black hole region, only a negligible value of about 0.003% false positive was observed with four base stations. With two or three base stations, the false positive was reduced to a value less than 0.01%. This indicates the efficiency of the approach used in this research.

4.5 Convergence Speed Results of Modified BFOA

The convergence speed is used to determine the performance of the network with respect to base station. The time of packet delivery for one, two, three and four base stations using 20*m*, 30*m* and 40*m* radius of black hole region was determined as shown in Figure 4.5.

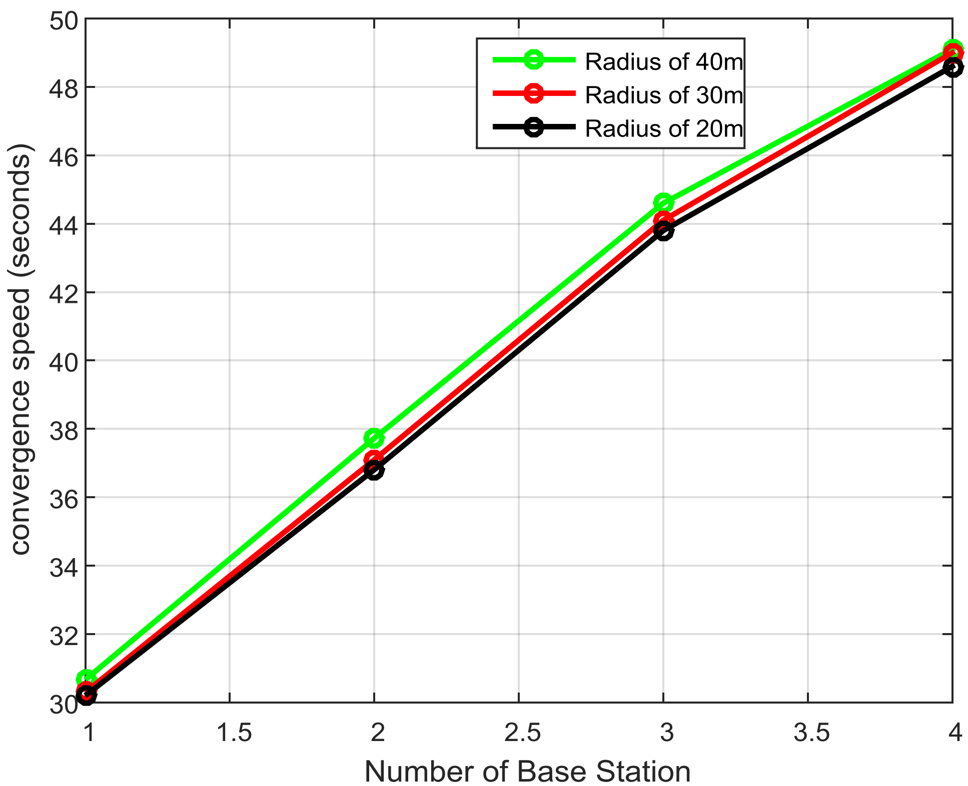

Figure 4.5: Time of Packet Delivery

From Figure 4.5, it was observed that the time of packet delivery increased as the number of base stationswere increased. This was expected, since sensor nodes route its packet through different paths depending on the number of base stations. With 20*m* radius of black hole region, it took an average sensor node deployment of about 30.2s, 36.8s, 43.8s and 48.6s before the packet was delivered using one, two, three and four base stations respectively. Similarly, with 30*m* radius of black hole region, it took an average sensor node deployment of about 30.3s, 37.1s, 44.1s and 49s before the packet was delivered using one, two, three and four base stations respectively. Also, with 40*m* radius of black hole region, it took an average sensor node deployment of about 30.7s, 37.7s, 44.6s and 49.1s before the packet was delivered using one, two, three and four base stations respectively.These time differences are within a tolerable value compared to the routing complexity of increasing the base stations. From the results obtained, it can be observed that increasing radius of black hole region does not have much effect on the time of packet delivery. This is due to the increased routing path of the multiple base stations.

The MATLAB program used to generate all the response given in this section can be found in appendix C.

4.6 Comparison between the Results of the Modified BFOA and the StandardBFOA

The result of Packet delivery success obtained using the modified BFOA was compared with the standard BFOA proposed in (Arya & Raina, 2014) as shown in Figure 4.6**.** Notice that, 1Bx, 2Bx, 3Bx and 4Bx implies one, two, three and four base stations respectively.

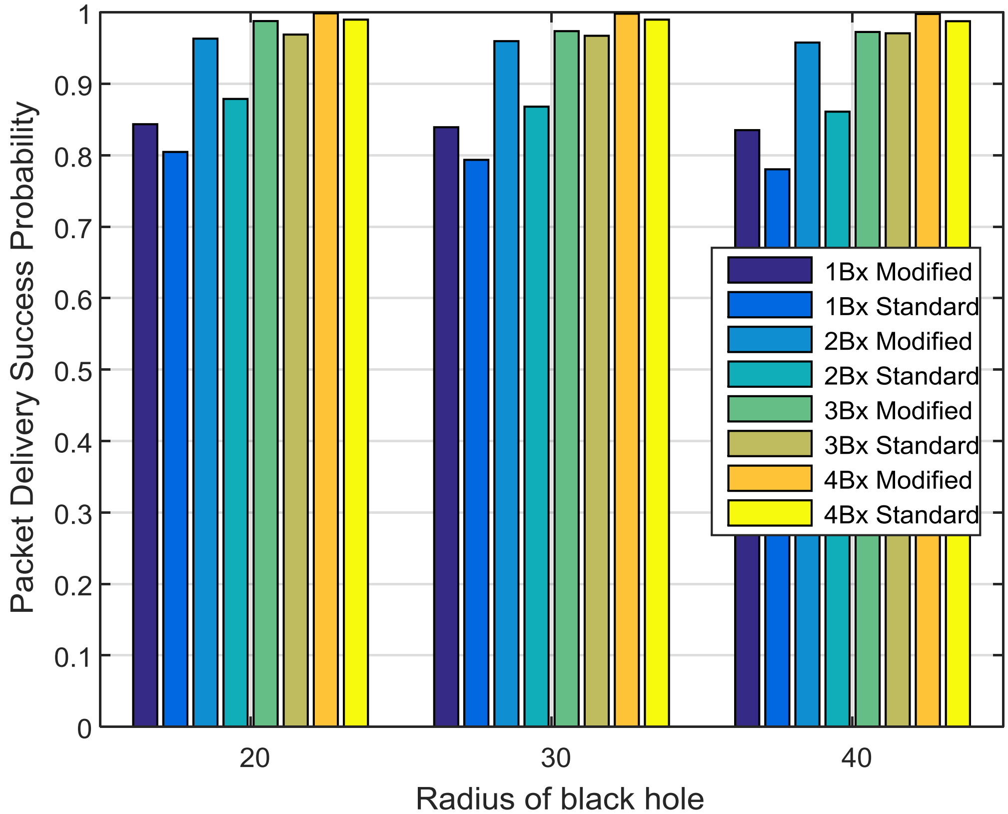

Figure 4.6: Packet Delivery Success using the Modified BFOAand Standard BFOA

From Figure 4.6, it can be observed that the results obtained using the modified BFOA was better than that obtained using the standard BFOA in terms of packet delivery success.

With 20*m* radius of black hole region using one, two, three and four base stations, a packet delivery success of 84.35%, 96.33%, 98.79% and 99.87% respectivelywas obtained using the modified BFOA. But using the standard BFOA, a packet delivery success of 80.47%, 87.89%, 96.90% and 98.99% was obtained for one, two, three and four base stations respectively. With 30*m* radius of black hole region using one, two, three and four base stations, a packet delivery success of 83.93%, 95.98%, 97.38% and 99.82% respectivelywas obtained using the modified BFOA. But using the standard BFOA, a packet delivery success of 79.36%, 86.81%, 96.73% and 98.99% was obtained for one, two, three and four base stations respectively. With 40*m* radius of black hole region using one, two, three and four base stations, a packet delivery success of 83.52%, 95.78%, 97.26% and 99.78% respectively was obtained using the modified BFOA. But using the standard BFOA, a packet delivery success of 78.04%, 86.11%, 97.08% and 98.77% was obtained for one, two, three and four base stations respectively. This indicates that there is a marginal improvement in the effectiveness of the performance of the modified BFOA compared to the standard BFOA.

The Packet delivery failure obtained using the modified BFOA and the standard BFOA are shown in Figure 4.7.

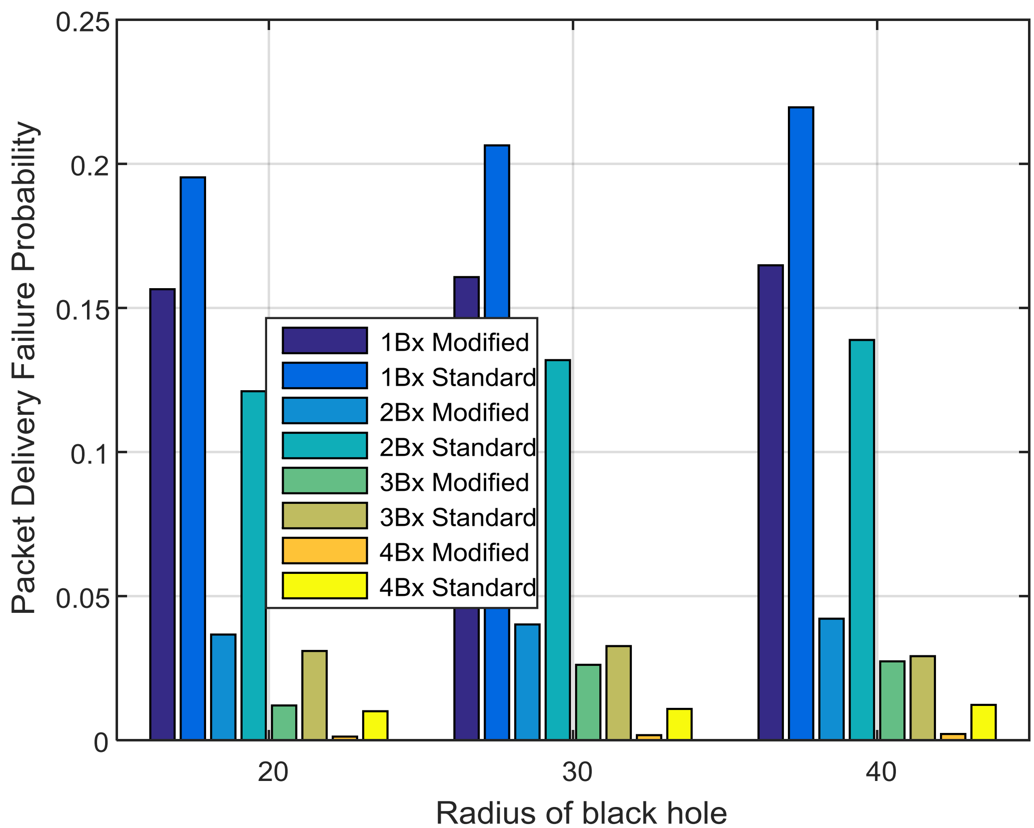

Figure 4.7: Packet Delivery Failure using the Modified BFOA and Standard BFOA

From Figure 4.7, it can be observed that the results of Packet delivery failure obtained using the modified BFOA were less in all simulation scenarios than that obtained using the standard BFOA.

With 20*m* radius of black hole region using one, two, three and four base stations, a packet delivery failure of 15.65%, 3.67%, 1.21% and 0.13% respectivelywas obtained using the modified BFOA. But using the standard BFOA, a packet delivery failure of 19.53%, 12.11%, 3.10% and 1.09% was obtained for one, two, three and four base stations respectively. With 30*m* radius of black hole region using one, two, three and four base stations, a packet delivery failure of 16.07%, 4.02%, 2.62% and 0.18% respectively was obtained using the modified BFOA. But using the standard BFOA, a packet delivery failure of 20.64%, 13.19%, 3.27% and 1.01% was obtained for one, two, three and four base stations respectively. With 40*m* radius of black hole region using one, two, three and four base stations, a packet delivery failure of 16.48%, 4.22%, 2.74% and 0.22% respectively was obtained using the modified BFOA. But using the standard BFOA, a packet delivery failure of 21.96%, 13.89%, 2.92% and 1.23% was obtained for one, two, three and four base stations respectively.

The False positive obtained using the developed method and the standard BFOA are shown in Figure 4.8.

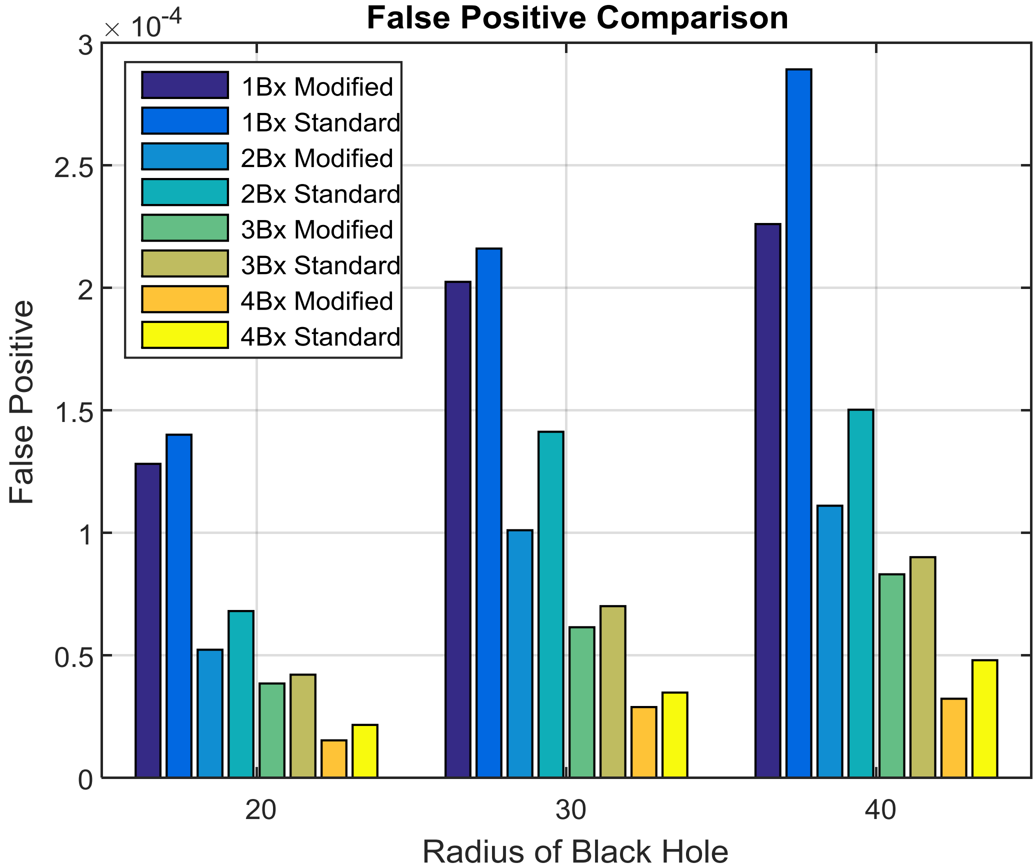

Figure 4.8: False Positive using the Modified BFOA and Standard BFOA

From Figure 4.8, it was observed that the modified BFOA obtained less false positive as the radius and number of base station was increased compared to that obtained using the standard BFOA. This is expected, due to the improved behaviour of the developed method. With 20*m* radius of black hole region using one, two, three and four base stations, a false positive value of 0.0128%, 0.0052%, 0.0042% and 0.0015% respectivelywas obtained using the modified BFOA. But using the standard BFOA, a false positive value of 0.014%, 0.0068%, 0.0048% and 0.0021% was obtained for one, two, three and four base stations respectively. With 30*m* radius of black hole region using one, two, three and four base stations, a false positive value of 0.0202%, 0.0101%, 0.0061% and 0.0029% respectively was obtained using the modified BFOA. But using the standard BFOA, a false positive value of 0.0216%, 0.0141%, 0.007% and 0.0035% was obtained for one, two, three and four base stations respectively. With 40*m* radius of black hole region using one, two, three and four base stations, a false positive value of 0.0226%, 0.0111%, 0.0083% and 0.0032% respectivelywas obtained using the modified BFOA. But using the standard BFOA, a false positive value of 0.0289%, 0.0502%, 0.009% and 0.0048% was obtained for one, two, three and four base stations respectively.

The Convergence speed obtained using the modified BFOA and the standard BFOA are shown in Figure 4.9.

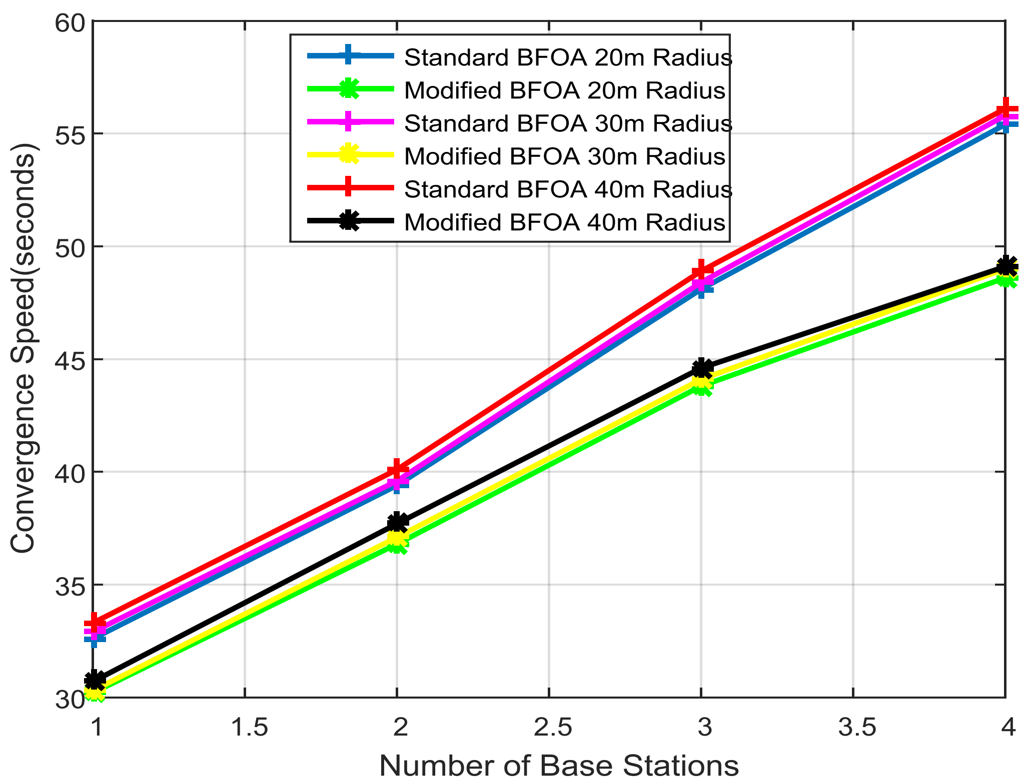

Figure 4.9: Convergence Speed using the Modified BFOA and Standard BFOA

From Figure 4.9 it was observed that the time of Packet delivery using the modified BFOA was less than the time using the standard BFOA. For one, two, three and four base stations, a delivery time of 32.6s, 39.4s, 48.1s and 55.4s respectively was obtained for 20*m* radius of black hole regionusing the standard BFOA. But using the modified method, a delivery time of 30.2s, 36.8s, 43.8s and 48.6s for one, two, three and four base stations respectively were obtained. Similarly, with one, two, three and four base stations, a delivery time of 32.9s, 39.6s, 48.4s and 55.8s respectively was obtained for 30*m* radius of black hole region using the standard BFOA. But using the modified method, a delivery time of 30.3s, 37.1s, 44.1s and 49.0s for one, two, three and four base stations respectively were obtained. Also, with one, two, three and four base stations, a delivery time of 33.3s, 40.1s, 48.9s and 56.1s respectively was obtained for 40*m* radius of black hole region using the standard BFOA. But using the modified method, a delivery time of 30.7s, 37.7s, 44.6s and 49.1s for one, two, three and four base stations respectively were obtained. These results indicated thatthe improvement in the modified BFOA has an effect on how fast the algorithm converges. Appendix D shows the MATLAB program for generating these results.

4.7 Result ofExtendedBase Stations

In order to further evaluate the performance of the model, the number of base stations were extended to eight under the same network configurations. The bar chart showing the percentage of delivery probability for five, six, seven and eight base stations is shown in Figure 4.10.

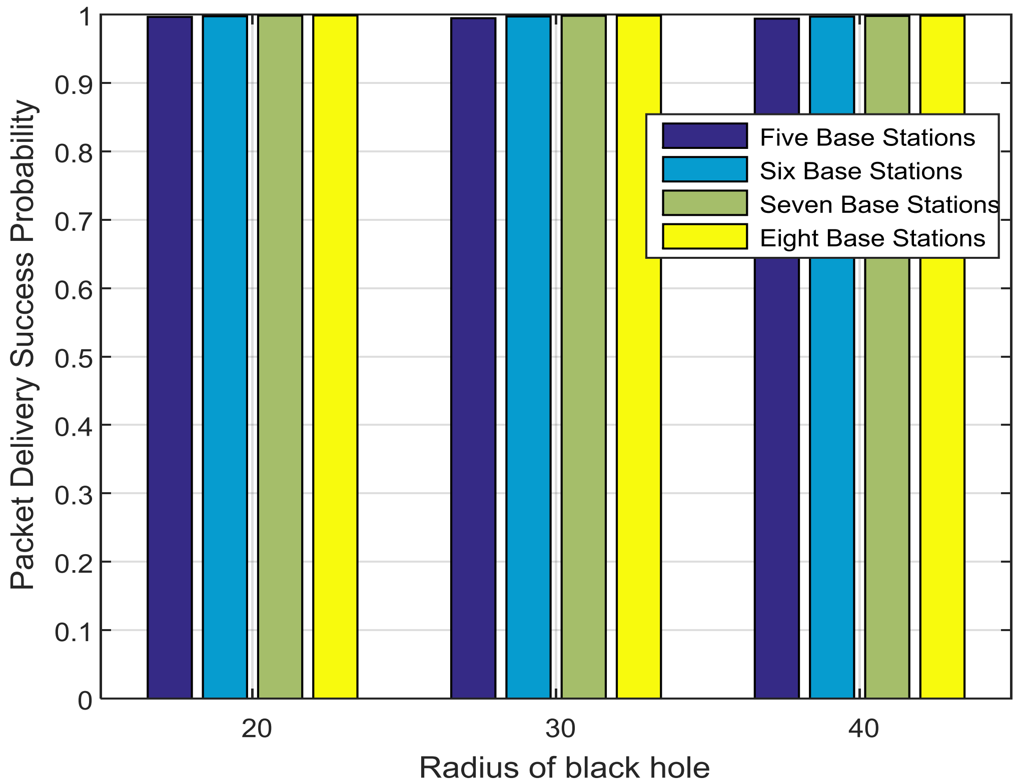

Figure 4.10: Packet Delivery for Five, Six, Seven and Eight Base Stations

Packet delivery success of 99.65%, 99.74%, 99.83% and 99.86% was obtained for five, six, seven and eight base stations respectively when the radius of black hole region was 20*m*. When the radius of black hole region was increased to 30*m*, the packet delivery success was 99.46%, 99.71%, 99.81% and 99.84% for five, six, seven and eight base stations respectively. Similarly, when the radius of black hole region was 40*m*, the packet delivery success was 99.39%, 99.69%, 99.79% and 99.82% for five, six, seven and eight base stations respectively. The bar chart showing the packet delivery probability for all the base stations (1-8) is shown in Figure 4.11.

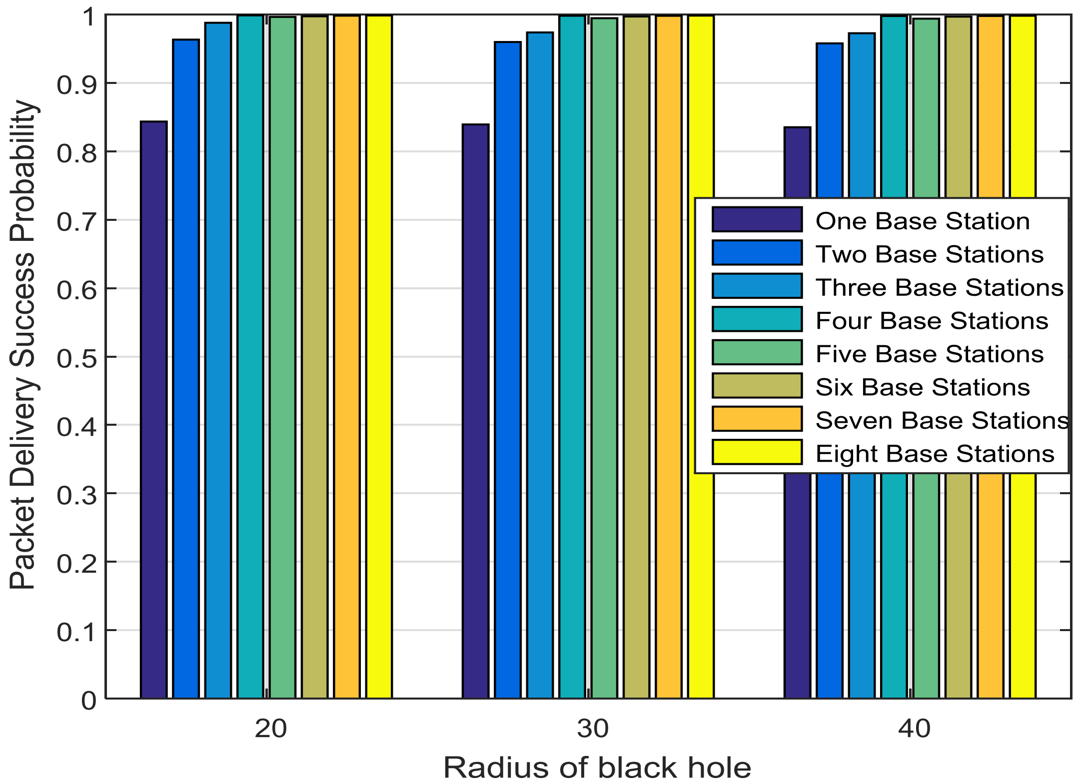

Figure 4.11: Packet Delivery for all the Base Stations

It was observed from Figure 4.11 that as the number of base stations was increased from four to five, the efficiency in packet delivery drops slightly and rose again as it was increased to six, seven and eight base stations.

This shows that, for the configuration used in this research, optimum efficiency in delivery was achieved with quadrant placement of base stations. Also the efficiency in packet delivery decreased with increase in radius of black hole region for all the number of base stations. The behaviour of the network as the number of base station was increased on the different radius of the network is shown in Figure 4.12.

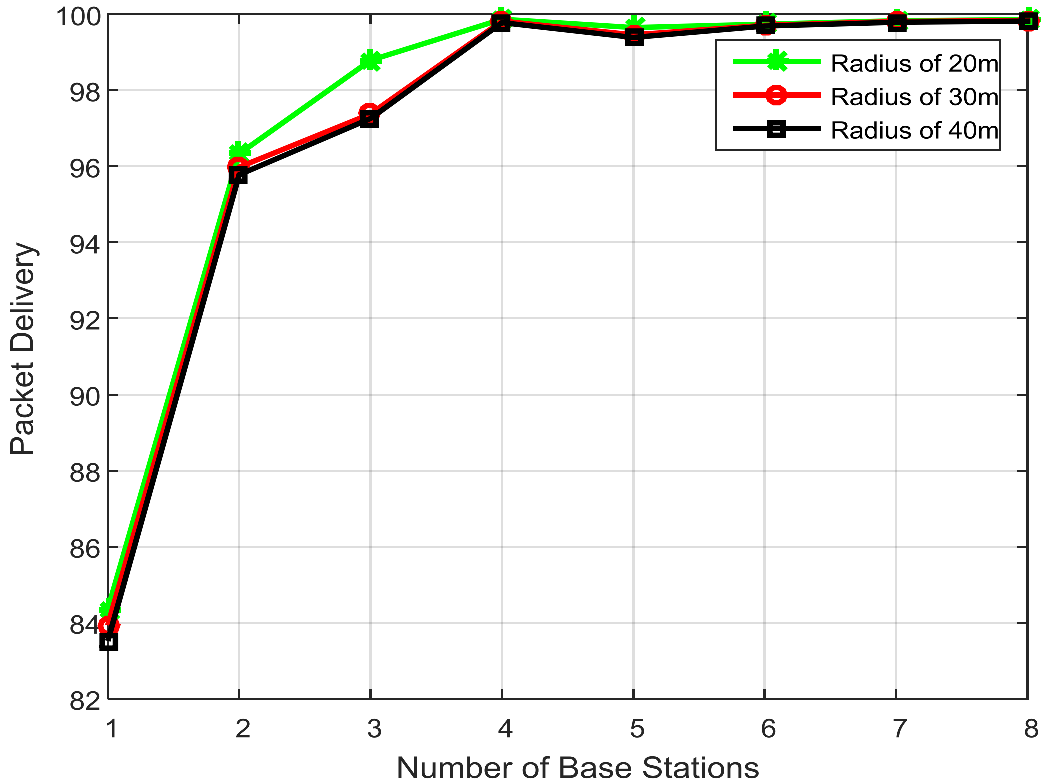

Figure 4.12: Behaviour of the Base Station with Different Radii

From Figure 4.12, it was observed that as the number of base station was increased beyond four (4) base stations, the packet delivery of the network appears to be approximately the same. This is an indication that, for the network configuration considered in this research work, optimal placement of four (4) base stations is enough to guarantee an efficient packet delivery.

CHAPTER FIVE

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

A Black hole attack mitigation model for wireless sensor network using a modified BFOA has been developed. The modified BFOA was used to randomly deploy multiple base stations in the network, which was aimed at improving the packet delivery success and countering the impact of black hole attack. Simulation results showedthat, the developed approach at 20*m* radius obtained a packet delivery success of 84.35%, 96.33%, 98.79% and 99.87% for one, two, three and four base stations respectively. Similarly, at 30*m* and 40*m* the results were as follows: 83.93%, 95.98%, 97.38% and 99.82%; and 83.52%, 95.78%, 97.26% and 99.78% respectively. This is an indication that, optimal placement of multiple base station using an intelligent optimization algorithm can significantly reduce the effect of wireless sensor network attackers while ensuring successful packet delivery. Validation was carried out by comparing the developed method with the standard BFOA used in the work of Arya and Raina (2014). Results showed that, the developed method had an improved performance over the standard BFOA. For the standard BFOA, at 20*m* radius a packet delivery success of 80.47%, 87.89%, 96.90% and 98.99% were obtained for one, two, three and four base stations respectively. Similarly, at 30*m* and 40*m* the results were as follows: 79.36%, 86.81%, 96.73% and 98.99%; and 78.04%, 86.11%, 97.08% and 98.77% respectively. The modified BFOA for the extended eight base stations obtained a delivery success of 99.65%, 99.74%, 99.83% and 99.86% for five, six, seven and eight base stations respectively using 20*m* radius of black hole region. Similarly, at 30*m* and 40*m* the results were as follows: 99.46%, 99.71%, 99.81%, and 99.84%; and 99.39%, 99.69%, 99.79% and 99.82% respectively. For the configuration used in this research, it can be concluded that optimum efficiency in delivery was achieved with quadrant placement of base stations. Also, this research has confirmed that quadrant placement of base stations achieved a better delivery performance. After each quadrant (four base stations), which is the optimum number for the configuration used in this research, it falls for the next base station (fifth base station) and then continue rising up to the eight base stations.

5.2 Limitations

Although the aim of this research work was successfully achieved, but some of the limitations of this research work are highlighted as follows:

i) This research work is limited to a network coverage area of 100 x 100 square meters.

ii) The research work is limited to only network of mobile nodes, which may not be the case in all scenarios.

5.3 Significant Contributions

The significant contributions of this work are as follows:

i) Development of an improved Bacterial foraging optimization algorithm (BFOA) with optimal control parameters and tolerance value. This has added adaptive capabilities to the existing standard BFOA, thus improving its convergence speed and accuracy.

ii) The developed modified BFOA showed a packet delivery improvement of 3.88%, 8.44%, 1.89% and 0.88% over the standard method as the number of base stations was increased from one to four respectively for 20*m* radius of black hole region. Similarly, for 30*m* and 40*m* radius, the delivery improvements were as follows: 4.57%, 9.17%, 0.65% and 0.83%; and 5.48%, 9.67%, 0.18% and 1.01% respectively.

5.4 Recommendations for Further Works

The following possible areas of further works are recommended for further research:

i) The modified BFOA for the eight base stations can be tested on an increased coverage area, so as to determine its efficiency as the complexity of wireless sensor network environment increases.

ii) The performance of the model developed in this work can be compared with model developed using similar optimization technique like the ACO, AFSA, ABC etc.

iii) The modified BFO can be applied in other combinatorial optimization problem.

REFERENCES

Annu, S., & Chaudhary, J. (2015). A review of BFOA applications to WSN. *International Journal of Advanced Foundation and Research in Computer (IJAFRC), 2*(5), 1-15.

Arya, M., & Raina, E. J. P. S. (2014). BFO based optimized positioning for black hole attack mitigation in WSN. *International Journal of Engineering Trends and Technology (IJETT), 14*(1), 29-34.

Atakli, I. M., Hu, H., Chen, Y., Ku, W. S., & Su, Z. (2008). Malicious node detection in wireless sensor networks using weighted trust evaluation.*Symposium on Simulation of Systems Security (SSSS '08)*, 836-843.

Bermejo, E., Cordón, O., Damas, S., & Santamaría, J. (2013). Quality time-of-flight range imaging for feature-based registration using bacterial foraging. *Journal of Applied Soft Computing (ASOC), 13*(6), 3178-3189.

Das, S., Biswas, A., Dasgupta, S., & Abraham, A. (2009). Bacterial foraging optimization algorithm: theoretical foundations, analysis, and applications. *In book: Foundations of Computational Intelligence, Springer, 3*, 23-55.

Dasgupta, S., Das, S., Abraham, A., & Biswas, A. (2009). Adaptive computational chemotaxis in bacterial foraging optimization: an analysis. *IEEE Transactions on Evolutionary Computation,, 13*(4), 919-941.

Dasgupta, S., Das, S., Biswas, A., & Abraham, A. (2010). Automatic circle detection on digital images with an adaptive bacterial foraging algorithm. *International Journal of Software Computing, 14*(11), 1151-1164.

Dighe, P. G., & Vaidya, M. B. (2013). Counter effects of black Hole attack on data transmission in wireless sensor network with multiple base stations. *International Journal of Engineering and Innovative Technology (IJEIT), 3*(5), 346-351.

Gondwal, N., & Diwaker, C. (2013). Detecting blackhole attack in WSN by check agent using multiple base stations. *American International Journal of Research in Science, Technology, Engineering & Mathematics, 3*(2), 149-152.

Iftikhar, M. S., & Fraz, M. R. (2013). A survey on application of swarm intelligence in network security. *Transactions on Machine Learning and Artificial Intelligence, 1*(1), 1-15.

Kalaiselvan, K., & Singh, G. (2015). Detection and isolation of blackhole attack in wireless sensor networks. *International journal of innovative research in science, engineering and technology (IJIRSET), 4*(5), 3516-3524.

Kandasamy, M., Vijayachitra, S., & Saravanan, K. (2014). Heuristic algorithm based controller for a real time pH neutralization process system. *International Journal of ChemTech research, 7*, 2320-2332.

Karakehayov, Z. (2005). Using REWARD to detect team black-hole attacks in wireless sensor networks. *In ACM Workshop on Real-World Wireless Sensor Networks (REALWSN'05). Retrieved from http://www.researchgate.net/publication/228628571*

Karlof, C., & Wagner, D. (2003). Secure routing in wireless sensor networks: attacks and countermeasures. *Journal of Ad hoc networks, 1*(2), 293-315.

Ketel, M., Dogan, N. S., & Homaifar, A. (2005). Distributed sensor networks based on mobile agents paradigm. *In proceedings of the 37th Southeastern Symposium on system theory (SSST'05)*, 411-414.

Khan, F., & Iftikhar, M. (2013). Optimized energy-efficient iterative distributed localization for wireless sensor networks.*IEEE 2013 IInternational Conference on Systems, Man, and Cybeernetics (SMC)*, 1407-1412.

Kumar, H., Sharma, D., & Kumar, A. (2006). Security threats in wireless sensor networks. *40th Annual IEEE International Carnahan Conference on Security Technology*, 243-451.

Kumar, N. (2015). Variation in wireless sensor network performance parameters under black hole attack and it's mitigation. *International Journal of Engineering Research and General Science, 3*(3), 479-485.

Kumar, S., & Sangwan, S. (2015). A survey of black hole detection techniques in WSNs. *International Journal of Advanced Research in Computer and Communication Engineering, 4*(5), 557-562.

Kumar, V., Jain, A., & Barwal, P. (2014). Wireless sensor networks: security issues, challenges and solutions. *International Journal of Information & Computation Technology, 4*(8), 859-868.

Latiff, N., Tsimenidis, C. C., & Sharif, B. S. (2007). Energy-aware clustering for wireless sensor networks using particle swarm optimization.*18th IEEE International Symposium on Personal, Indoor and Mobile Radio Communication*, 2713-2717.

Li, J., Dang, J., Bu, F., & Wang, J. (2014). Analysis and improvement of the bacterial foraging optimization algorithm. *Journal of Computing Science and Engineering (JCSE), 8*(1), 1-10.

Lou, W., & Kwon, Y. (2006). H-SPREAD: a hybrid multipath scheme for secure and reliable data collection in wireless sensor networks. *In proceedings of the IEEE Transactions on Vehicular Technology, 55*(4), 1320-1330.

Martins, D., & Guyennet, H. (2010). Wireless sensor network attacks and security mechanisms: a short survey. *13th IEEE International Conference on Network Based Information Systems, 313-320.*

Michalik, M. (2013). Base station for wireless sensor network. *A published diploma thesis, Retrieved from http://is.muni/czz/th/256522/fi_m/dp.pdf*.

Misra, S., Bhattarai, K., & Xue, G. (2011). BAMBi: blackhole attacks mitigation with multiple base stations in wireless sensor networks. *In proceedings of the IEEE International Conference on Communications, 1-5.*

Mua'zu, M.B., Salawudeen, A.T., Sikiru, T.H., Abdu, A., & Mohammad, A. (2015). Weighted artificial fish swarm algorithm with adaptive behaviour based linear controller design for nonlinear inverted pendulum. *Journal of Engineering Research, 20*(1), 1-12.

Nasir, A., Tokhi, M. O., & Ghani, N. A. (2015). Novel adaptive bacterial foraging algorithms for global optimisation with application to modelling of a TRS. *Journal of Expert Systems with Applications, 42*(3), 1513-1530.

Ngai, E. C., Liu, J., & Lyu, M. R. (2006). On the intruder detection for sinkhole attack in wireless sensor networks.*In proceedings of the IEEE International Conference on Communications (ICC'06),* 3383-3389.

Niu, B., Wang, J., & Wang, H. (2015). Bacterial-inspired algorithms for solving constrained optimization problems. *Journal of Neurocomputing, 148*, 54-62.

Passino, K. M. (2002). Biomimicry of bacterial foraging for distributed optimization and control. *IEEE Control Systems Magazine, 22*(3), 52-67.

Pathan, A.-S. K., Lee, H.-W., & Hong, C. S. (2006). Security in wireless sensor networks: issues and challenges. *8th International Conference on Advanced Communication Technology (ICACT), 2, 6-104.*

Pawar, M. V., & Anuradha, J. (2015). Network security and types of attacks in network. *International Conference on Intelligent Computing, Communication and Convergence (ICCC), Procedia computer science, 48*, 503-506.

Qureshi, S., Asar, A., Rehman, A., & Baseer, A. (2011). Swarm intelligence based detection of malicious beacon node for secure localization in wireless sensor networks. *Journal of Emerging Trends in Engineering and Applied Sciences, 4*(2), 664-672.

Roy, S. D., Singh, S. A., Choudhury, S., & Debnath, N. C. (2008). Countering sinkhole and black hole attacks on sensor networks using dynamic trust management. *IEEE Symposium on Computers and Communications, 537-542.*

Saleh, M., & Khatib, I. (2005). Throughput analysis of WEP security in ad hoc sensor networks.*Proceedings of the Second International Conference on Innovations in Information Technology (IIT'05)*, 25-28.

Salehi, S. A., Razzaque, M. A., Naraei, P., & Farrokhtala, A. (2013). Detection of sinkhole attack in wireless sensor networks. Proceedings of *the 2013 IEEE International Conference on Space Science and Communications (IconSpace)*, 361-365.

Sharma, A., & Thakur, J. (2014). An energy efficient network life time enhancement proposed clustering algorithm for Wireless Sensor Networks. *International Journal of Enhanced Research in Management & Computer Applications, 2*(7), 1-4.

Sharma, K., & Ghose, M. (2010). Wireless sensor networks: an overview on its security threats. *International Journal of Computer Applications (IJCA) Special Issue on "Mobile Ad-hoc Networks" MANETs,* 42-45.

Sharma, V., Patil, K., & Tiwari, A. (2014). Detection and suppression of blackhole attack in leach based sensor network. *International Journal of Computer Technology and Applications, 5*(6), 1873-1877.

Sheela, D., & Mahadevan, G. (2012). Mollifying the effect of cloning, sink hole and black hole attacks in wireless sensor networks using mobile agents with several base stations. *International Journal of Computer Applications, 55*(9), 34-41.

Sheela, D., Srividhya, V., Asma, B., & Chidanand, G. (2012). Detecting black hole attacks in wireless sensor networks using mobile agent. *International Conference on Artificial Intelligence and Embedded Systems (ICAIES), 45-48.*

Shu, T., Krunz, M., & Liu, S. (2010). Secure data collection in wireless sensor networks using randomized dispersive routes. *IEEE Transactions on Mobile Computing, 9*(7), 941-954.

Singh, G., & Kaur, S. (2015). A review of black hole in MANET. *International Journal of Scientific Research and Development, 3*(3), 2356-2358.

Singh, H., & Malik, G. (2011). Approaches to wireless sensor network: security protocols. *World of Computer Science and Information Technology Journal (WCSIT), 1*(7), 302-306.

Singh, S. K., Singh, M. P., & Singh, D. K. (2010). Routing protocols in wireless sensor networks - a survey. *International Journal of computer science and engineering survey (IJCSES), 1*(2), 63-83.

Soni, V., Modi, P., & Chaudhri, V. (2013). Detecting sinkhole attack in wireless sensor network. *International Journal of Application or Innovation in Engineering & Management, 2*(2), 29-32.

Supriyono, H., & Tokhi, M. (2011). Adaptation schemes of chemotactic step size of bacterial foraging algorithm for faster convergence. Journal of Artificial Intelligence, 4(4), 207-219.

Tomar, A. S., & Tak, G. K. (2014). Optimized positioning of multiple base station for black hole attack. *International Journal of Advanced Research in Computer Engineering & Technology,* 3(8*)*, 2767-2770.

Tubaishat, M., Yin, J., Panja, B., & Madria, S. (2004). A secure hierarchical model for sensor network. *In Newsletter of Association of Computing Machinery (ACM) Sigmod Record,* 33(1), 7-13.

Yadav, P., Gill, R. K., & Kumar, N. A. (2012). A fuzzy based approach to detect black hole attack. *International Journal of Soft Computing and Engineering (IJSCE),* 2(3), 388-391.

APPENDIX A

MATLAB script for the modified BFOA

```
functionmBFA
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%Authors: Zainab.M.A
%Prof. M.B. Mu'azu
%Dr. T.H Sikiru
%Post-Graduate Department of Computer & Control Engineering; DEPARTMENT OF
%ELECTRICAL AND COMPUTER ENGINEERING.
%AHMADU BELLO UNIVERSITY ZARIA NIGERIA....
%!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
%THIS MATLAB PROGRAM IS DEVELOPED FOR OPTIMIZED BASE STATION BLACK HOLE
%DETECTION IN WSN USING A MODIFIED BACTERIAL FORAGING ALGORITHM
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%
%The Chemotaxis, Swarming, Reproductive and Elimination/Dispersal behaviour of
%E. Coli bacterial is simulated for Base station optimization problem in WSN.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%
clc
closeall
clearall
%Bs=input('Please Provide the Number of Base Station (between 1 to 4):= ');
clc
%P=input('PLEASE PROVIDE THE PROBLEM DIMENSION:= ');% Dimension of the search space
clc
%r=input('Select the Radius of black hole region:= ');
clc
% if P==0
p=2;
% else
%     p=P;
% end
%s=input('PLEASE PROVIDE THE POPULATION OF BACTERIAL:= ');%Population of Bacterial
clc
% if s==0
S=200;
% else
%     S=s;
% end
%nC=input('PLEASE PROVIDE THE NUMBER OF CHEMOTAXIS STEP:= ');%chemotaxis
clc
% if nC==0
Nc=100; %This value may be different for every bacterial in the population
% else
%     Nc=nC;
% end
%nS=input('PLEASE ENTER THE LENGHT OF SWIM LIMI:= ');
clc
% if nS==0
Ns=4;% Limits the length of a swim
% else
%     Ns=nS;
% end
%NR=input('PLEASE ENTER THE VALUE OF REPRODUCTION STEP:= ');
clc
% if NR==0
Nre=4;   % The number of reproduction steps
% else
%     Nre=NR;
```

```
% end
Sr=S/2;  % The number of bacteria reproductions (splits) per generation (this
%ND=input('PLEASE ENTER THE NUMBER OF ELIMINATION/DISPASAL EVENTS:= ');
clc
% if ND==0
Ned=2; % The number of elimination-dispersal events (Nre reproduction steps in between each event)
% else
% Ned=ND;
% end
% PD=input('PLEASE ENTER THE PROBABILITY THAT EACH BACTERIAL WILL BEE ELIMINATED:= ');
% clc
% if PD==0
ped=0.25;
% else
%   ped=PD;
% end
Bs=[1,2,3,4];
ra=[20,30,40];
flag=2; % flag=0 indicates  nutrients and cell-cell attraction
% flag=1 indicates  no (zero) nutrients and only cell-cell attraction
% flag=2 indicates  nutrients and no cell-cell
% attraction
% Initialize population

tic
P(:,:,:,:,:)=0*0*ones(p,S,Nc,Nre,Ned);
for m=1:S
    P(:,m,1,1,1)=1*(15*((2*round(rand(p,1))-1).*rand(p,1))+[15;15]);
end
C=0*ones(S,Nre); % Allocate empty memory
rlengthunit=0.1;
C(:,1)=rlengthunit*ones(S,1);
% Allocate memory for cost function:
J=0*ones(S,Nc,Nre,Ned);
Jhealth=0*ones(S,1);
B=input('Please Provide the Number of Base Station (between 1 to 4):= ');
clc
R=input('Selet the Radius of black hole region:= ');
clc
ITR=2;%input('Please Provide the total number of iteration:= ');
iteration=0;
% Elimination-dispersal loop:
%Alocate memory for swarm
sw=ones*Nre/(Ns+0.8);
tw=ones*Ned/(p+0.35);
fl=ones*Ned/Ned-5^-2;
pw=ones*Nre/10000;
qw=ones*Ned/1000;
while iteration<ITR
iteration=iteration+1;
for el=1:Ned
% Reproduction loop:
for k=1:Nre
% Swim/tumble (chemotaxis) loop:
%---------------------------------
for j=1:Nc
```

```
fori=1:S  % For each bacterium
% Compute the Node Density at the current location of
% each bacterium
        J(i,j,k,el)=optimfunc(P(:,i,j,k,el),flag);
%       J(i,j,k,el)=optimfunc1(P(:,i,j,k,el),flag);
% Next, add the cell-cell attraction effect:
        J(i,j,k,el)=J(i,j,k,el)+bact_cellcell_attract_func(P(:,i,j,k,el),P(:,:,j,k,el),S,flag);
%-----------
% Tumble:
Jlast=J(i,j,k,el);
        Delta(:,i)=(2*round(rand(p,1))-1).*rand(p,1);
        P(:,i,j+1,k,el)=P(:,i,j,k,el)+C(i,k)*Delta(:,i)/sqrt(Delta(:,i)'*Delta(:,i));
        J(i,j+1,k,el)=optimfunc(P(:,i,j+1,k,el),flag); % Node concentration for each bacterium after
%       J(i,j+1,k,el)=optimfunc1(P(:,i,j+1,k,el),flag);
% Next, add the cell-cell attraction effect:
        J(i,j+1,k,el)=J(i,j+1,k,el)+bact_cellcell_attract_func(P(:,i,j+1,k,el),P(:,:,j+1,k,el),S,flag);
        m=0; % Initialize counter for swim length
while m<Ns
            m=m+1;
if J(i,j+1,k,el)<Jlast
Jlast=J(i,j+1,k,el);
                P(:,i,j+1,k,el)=P(:,i,j+1,k,el)+C(i,k)*Delta(:,i)/sqrt(Delta(:,i)'*Delta(:,i));
                J(i,j+1,k,el)=optimfunc(P(:,i,j+1,k,el),flag);
%               J(i,j+1,k,el)=optimfunc1(P(:,i,j+1,k,el),flag);
                J(i,j+1,k,el)=J(i,j+1,k,el)+bact_cellcell_attract_func(P(:,i,j+1,k,el),P(:,:,j+1,k,el),S,flag);
else
                m=Ns;
end
end
end
end
tl=ones*Nre/((Nre+0.20.*ones));
sl=ones*Nre/((Nre+0.044.*ones));
gl=ones*Ned/Ned-9.7^-2;
Jhealth=sum(J(:,:,k,el),2);
    [Jhealth,sortind]=sort(Jhealth);
    P(:,:,1,k+1,el)=P(:,sortind,Nc+1,k,el);
ml=ones*Nre/((Nre+ones.*4^-3));
nl=ones*Nre/((Nre+ones.*10^-4));
%Introduce the Adaptive Step
C(:,k+1)=((ITR-iteration)/ITR).*C(sortind,k);
    r=tw+(sw-tw).*rand(Nre/Ns);
rr=pw+(qw-pw).*rand(Nre/Ns);
dl=ones*Nre/(Nre+0.1);
fori=1:Sr
        P(:,i+Sr,1,k+1,el)=P(:,i,1,k+1,el);
%Introduce the Adaptive Step
C(i+Sr,k+1)=((ITR-iteration)/ITR).*C(i,k+1);
end
end
for m=1:S
ifped>rand  % Generate random number and if ped bigger than it then eliminate/disperse
            P(:,m,1,1,el+1)=(15*((2*round(rand(p,1))-1).*rand(p,1))+[15;15]);rs=0.836;
else
            P(:,m,1,1,el+1)=P(:,m,1,Nre+1,el);  % Bacteria that are not dispersed
            Best=P(:,m,1,1,el+1);
```

```
end
end
end% el=1:Ned
end
FITNESS=optimfunc(Best,flag);
BEST=min(FITNESS);
x=0:31/100:30;   %For our function the range of values we are considering
y=x;
forjj=1:length(x)
for ii=1:length(y)
       z(ii,jj)=optimfunc([x(jj);y(ii)],flag);
%       z(ii,jj)=optimfunc1([x(jj);y(ii)],flag);
end
end
% First, show the actual function to be maximized
figure(1)
% clf
surfc(x,y,z);
%colormap(jet)
% Use next line for generating plots to put in black and white documents.
%colormap(white);
xlabel('X-COORDINATE');
ylabel('Y-COORDINATE');
zlabel('Z=FITNESS VALUE');
title('Nutrient concentration (valleys=food, peaks=Bx)');
cleart
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%
% Next, provide some plots of the results of the simulation.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%
t=0:Nc;  % For use in plotting (makes t=0 correspond to the 1 index and plots to Nc+1)
% As is Figure 2 (4) shows parameter trajectories for 4 generations, then there is an elimination/dispersal event
% Figure 3 (5) shows parameter trajectories for the following 4 generations
% (after the elim/disp event)
forkk=1:Ned
figure(kk+1)
% clf
for mm=1:Nre
subplot(2,2,mm)
fornn=1:S % Plot all bacteria trajectories for generation mm
plot(t,squeeze(P(1,nn,:,mm,kk)),'*',t,squeeze(P(2,nn,:,mm,kk)),'o')
%plot(t,squeeze(P(1,nn,:,mm,kk)),'-',t,squeeze(P(2,nn,:,mm,kk)),'-')
axis([min(t) max(t) 0 30])
holdon
end
T=num2str(mm);
T=strcat('Bacteria trajectories, Generation=',T);
title(T)
xlabel('ITERATION')
ylabel('CORDINATE PARTERN')
holdoff
end
end
forkk=1:1
figure(kk+1)
```

```
% clf
for mm=1:1
% subplot(2,2,mm)
contour(x,y,z,25)
colormap(jet)
fornn=1:S  % Plot all bacteria trajectories for generation mm
holdon
plot(squeeze(P(1,nn,:,mm,kk)),squeeze(P(2,nn,:,mm,kk)))
%plot(squeeze(P(1,nn,:,mm,kk)),squeeze(P(2,nn,:,mm,kk)),'-')
axis([0 30 0 30])
end
T=num2str(mm);
T=strcat('Bacteria trajectories, Generation=',T);
title(T)
% Use next line for generating plots to put in black and white documents.
%colormap(gray);
xlabel('X-CORDINATE');
ylabel('Y-CORDINATE');
holdoff
end
end
%%%%%%%%%%%%
pause% Can leave this in if you want to avoid movie (then hit control-C)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Next, show a movie of the chemotactic steps:
figure(2*Ned+2)
% clf
contour(x,y,z,25)
colormap(jet)
axis([0,30,0,30]);
xlabel('X-CORDINATE');
ylabel('Y-CORDINATE');
title('Bacteria (Nodes) movements');
holdon
M = moviein(Nc);
for j=1:Nc;
% Can change generation step and elimination-dispersal step on next line.
% Currently for 1,1 - the first generation in the first elimination dispersal step
fori=1:10
        v=plot(squeeze(P(1,i,j:j+1,1,1)),squeeze(P(2,i,j:j+1,1,1)),'-or');
set(v,'MarkerSize',5);
end
M(:,j)=getframe;
end;
movie(M,0)
for k=1:S
if B==Bs(1)&&R==ra(1);
   PDS=tw+(sw-tw).*rand(Nre/Ns);
   FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(1)&&R==ra(2);
   PDS=tw+(sw-tw).*rand(Nre/Ns);
   FP=pw+(qw-pw).*rand(Nre/Ns);
end
%%
if B==Bs(3)&&R==ra(1);PDS=fl+(gl-fl).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(3)&&R==ra(3);PDS=fl+(gl-fl).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
```

```
elseif B==Bs(4)&&R==ra(3);PDS=ml+(nl-ml).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(4)&&R==ra(2);PDS=ml+(nl-ml).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(4)&&R==ra(1);PDS=ml+(nl-ml).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(2)&&R==ra(1);PDS=sl+(tl-sl).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(2)&&R==ra(2);PDS=sl+(tl-sl).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(2)&&R==ra(3);PDS=sl+(tl-sl).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
end
%%
if  B==Bs(3)&&R==ra(2);PDS=fl+(gl-fl).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
end;
if  B==Bs(1)&&R==ra(3);PDS=tw+(sw-tw).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(3)&&R==ra(1);PDS=ml+(nl-ml).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
end
PDF=1-PDS;
end
toc
helpmBFA.m
disp('')
disp('')
%disp('The Best Solution Found So Far is :')
disp('')
disp('')
disp('Paket Delivery Success for these Configuration is: ')
disp(PDS)
disp('!!!!!!!!!!!!!!!!!!!!!!!!')
disp('And the Packet Delivery Failure is')
disp(PDF)
disp('!!!!!!!!!!!!!!!!!!!!!!!!')
disp('False Positive')
disp(FP)
```

APPENDIX B1

MATLAB script for the objective function formulation

```
%This will contain the base station to be optimized:
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%
%Authors: Zainab.M.A
%Prof. M.B. Mu'azu
%Dr. T.H Sikiru
%Post-Graduate Department of Computer & Control Engineering; DEPARTMENT OF
%ELECTRICAL AND COMPUTER ENGINEERING.
%AHMADU BELLO UNIVERSITY ZARIA NIGERIA....
%!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
%THIS MATLAB PROGRAM IS DEVELOPED FOR OPTIMIZED BASE STATION BLACK HOLE
%DETECTION IN WSN.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%
%THE OBJECTIVE FUNCTION IS FORMULATED BASED ON THE DENSITY OF TWO
%NEIGHBOURING BACTERIAL SENSOR NODES.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%
function J=soptimfunc(theta,flag)

if flag==1  % Test to see if main program indicated nutrients
    J=0;
return
end
forjj=1:50
    J=abs(sqrt((theta(1,1))^2-(flag(1,1))^2+theta(2,1)^2-(flag(1,1))^2));
end
```

APPENDIX B2

MATLAB script for the cell-to-cell attractant

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Bacteria cell to cell attraction function
```

```
% Version: 5/16/00
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%
function [Jar]=bact_cellcell_attract_func(x,theta,S,flag)
% Given locations of all bacteria, find Jar for all S bacteria
% Note that theta rows are dimensions of the opt. problem, while the columns are
% the S different bacteria. For now, designed for p=2

if flag==2  % Test to see if main program indicated cell-cell attraction
   Jar=0;
return
end

depthattractant=0.1;  % Sets magnitude of secretion of attractant by a cell
widthattractant=0.2;  % Sets how the chemical cohesion signal diffuses (smaller makes it diffuse more)

heightrepellant=1*depthattractant; % Sets repellant (tendency to avoid nearby cell)
widthrepellant=10;  % Makes small area where cell is relative to diffusion of chemical signal

Jar=0;

for j=1:S
% Set how the cell attracts other cells via secretions of diffusable attractants

   Ja=-depthattractant*exp(-widthattractant*((x(1,1)-theta(1,j))^2+(x(2,1)-theta(2,j))^2));

% Set how the cell repells other cells since it eats in its own region (and since an intact
% cell is apparently not food for another cell)

   Jr=+heightrepellant*exp(-widthrepellant*((x(1,1)-theta(1,j))^2+(x(2,1)-theta(2,j))^2));

% Next, set the combined effect

   Jar=Jar+Ja+Jr;

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%
```

APPENDIX B3

MATLAB script for the extended base stations

```
functionmBFA
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%
```

```
%Authors: Zainab.M.A
%Prof. M.B Mu'azu
%Dr. T.H Sikiru
%Post-Graduate Department of Computer Engineering; DEPARTMENT OF
%ELECTRICAL AND COMPUTER ENGINEERING.
%AHMADU BELLO UNIVERSITY ZARIA NIGERIA....
%!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
%!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
%!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
%THIS MATLAB PROGRAM IS DEVELOPED FOR OPTIMIZED BASE STATION BLACK HOLE
%DETECTION IN WSN USING A MODIFIED BACTERIAL FORAGING ALGORITHM
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%
%The Chemotaxis, Swarming, Reproductive and Elimination/Dispersal behaviour of
%E. Coli bacterial is simulated for Base station optimization problem in WSN.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%
clc
closeall
clearall
formatlong
%Bs=input('Please Provide the Number of Base Station (between 1 to 4):= ');
clc
%P=input('PLEASE PROVIDE THE PROBLEM DIMENSION:= ');% Dimension of the search space
clc
%r=input('Select the Radius of black hole region:= ');
clc
% if P==0
p=2;
% else
%     p=P;
% end
%s=input('PLEASE PROVIDE THE POPULATION OF BACTERIAL:= ');%Population of Bacterial
clc
% if s==0
S=200;
% else
%     S=s;
% end
%nC=input('PLEASE PROVIDE THE NUMBER OF CHEMOTAXIS STEP:= ');%chemotaxis
clc
% if nC==0
Nc=100; %This value may be different for every bacterial in the population
% else
%     Nc=nC;
% end
%nS=input('PLEASE ENTER THE LENGHT OF SWIM LIMI:= ');
clc
% if nS==0
Ns=4;% Limits the length of a swim
% else
%     Ns=nS;
% end
%NR=input('PLEASE ENTER THE VALUE OF REPRODUCTION STEP:= ');
clc
% if NR==0
```

```
Nre=4;   % The number of reproduction steps
% else
%     Nre=NR;
% end
Sr=S/2;  % The number of bacteria reproductions (splits) per generation (this
%ND=input('PLEASE ENTER THE NUMBER OF ELIMINATION/DISPERSAL EVENTS:= ');
clc
% if ND==0
Ned=2; % The number of elimination-dispersal events (Nre reproduction steps in between each event)
% else
% Ned=ND;
% end
% PD=input('PLEASE ENTER THE PROBABILITY THAT EACH BACTERIAL WILL BE ELIMINATED:= ');
% clc
% if PD==0
ped=0.25;
% else
%   ped=PD;
% end
Bs=[1,2,3,4,5,6,7,8];%This predefined Base station
ra=[20,30,40];%Predefined Radius of Black hole Region
flag=2; % flag=0 indicates  nutrients and cell-cell attraction
% flag=1 indicates  no (zero) nutrients and only cell-cell attraction
% flag=2 indicates  nutrients and no cell-cell
% attraction
% Initialize population


tic
P(:,:,:,:,:)=0*0*ones(p,S,Nc,Nre,Ned);
for m=1:S
    P(:,m,1,1,1)=1*(15*((2*round(rand(p,1))-1).*rand(p,1))+[15;15]);
end
C=0*ones(S,Nre); % Allocate empty memory
rlengthunit=0.1;
C(:,1)=rlengthunit*ones(S,1);
% Allocate memory for cost function:
J=0*ones(S,Nc,Nre,Ned);
Jhealth=0*ones(S,1);
%Select from the Base Stations Defined Previously
B=input('Please Provide the Number of Base Station:= ');
clc
%Select from the Radius of Black Hole Region Defined Previously
R=input('Selet the Radius of black hole region:= ');
clc
ITR=2;%input('Please Provide the total number of iteration:= ');
iteration=0;
% Elimination-dispersal loop:
%Alocate memory for swarm
sw=ones*Nre/(Ns+0.8);
tw=ones*Ned/(p+0.35);
fl=ones*Ned/Ned-5^-2;
pw=ones*Nre/10000;
qw=ones*Ned/1000;
while iteration<ITR
iteration=iteration+1;
for el=1:Ned
```

```
% Reproduction loop:
for k=1:Nre
% Swim/tumble (chemotaxis) loop:
%---------------------------------
for j=1:Nc
fori=1:S  % For each bacterium
% Compute the Node Density at the current location of
% each bacterium
       J(i,j,k,el)=optimfunc(P(:,i,j,k,el),flag);
%       J(i,j,k,el)=optimfunc1(P(:,i,j,k,el),flag);
% Next, add the cell-cell attraction effect:
       J(i,j,k,el)=J(i,j,k,el)+bact_cellcell_attract_func(P(:,i,j,k,el),P(:,:,j,k,el),S,flag);
%-----------
% Tumble:
Jlast=J(i,j,k,el);
       Delta(:,i)=(2*round(rand(p,1))-1).*rand(p,1);
       P(:,i,j+1,k,el)=P(:,i,j,k,el)+C(i,k)*Delta(:,i)/sqrt(Delta(:,i)'*Delta(:,i));
       J(i,j+1,k,el)=optimfunc(P(:,i,j+1,k,el),flag); % Node concentration for each bacterium after
%       J(i,j+1,k,el)=optimfunc1(P(:,i,j+1,k,el),flag);
% Next, add the cell-cell attraction effect:
       J(i,j+1,k,el)=J(i,j+1,k,el)+bact_cellcell_attract_func(P(:,i,j+1,k,el),P(:,:,j+1,k,el),S,flag);
       m=0; % Initialize counter for swim length
while m<Ns
           m=m+1;
if J(i,j+1,k,el)<Jlast
Jlast=J(i,j+1,k,el);
              P(:,i,j+1,k,el)=P(:,i,j+1,k,el)+C(i,k)*Delta(:,i)/sqrt(Delta(:,i)'*Delta(:,i));
              J(i,j+1,k,el)=optimfunc(P(:,i,j+1,k,el),flag);
%              J(i,j+1,k,el)=optimfunc1(P(:,i,j+1,k,el),flag);
              J(i,j+1,k,el)=J(i,j+1,k,el)+bact_cellcell_attract_func(P(:,i,j+1,k,el),P(:,:,j+1,k,el),S,flag);
else
              m=Ns;
end
end
end
end

Ev=ones*Nre/((Nre+ones.*4^-3.4));
tl=ones*Nre/((Nre+0.20.*ones));
sl=ones*Nre/((Nre+0.044.*ones));
gl=ones*Ned/Ned-9.7^-2;
Jhealth=sum(J(:,:,k,el),2);
   [Jhealth,sortind]=sort(Jhealth);
   P(:,:,1,k+1,el)=P(:,sortind,Nc+1,k,el);
ml=ones*Nre/((Nre+ones.*4^-3));
nl=ones*Nre/((Nre+ones.*10^-4));
%     sn=ones*Nre/((Nre+ones.*4^-3.78));
%Introduce the Adaptive Step
C(:,k+1)=((ITR-iteration)/ITR).*C(sortind,k);
%     r=tw+(sw-tw).*rand(Nre/Ns);
%     rr=pw+(qw-pw).*rand(Nre/Ns);
%     dl=ones*Nre/(Nre+0.1);
fori=1:Sr
       P(:,i+Sr,1,k+1,el)=P(:,i,1,k+1,el);
%Introduce the Adaptive Step
C(i+Sr,k+1)=((ITR-iteration)/ITR).*C(i,k+1);
```

```
end
end
for m=1:S
ifped>rand % Generate random number and if ped bigger than it then eliminate/disperse
        P(:,m,1,1,el+1)=(15*((2*round(rand(p,1))-1).*rand(p,1))+[15;15]);rs=0.836;
else
        P(:,m,1,1,el+1)=P(:,m,1,Nre+1,el);  % Bacteria that are not dispersed
        Best=P(:,m,1,1,el+1);
end
end
end% el=1:Ned
end
FITNESS=optimfunc(Best,flag);
BEST=min(FITNESS);
x=0:31/100:30;   %For our function the range of values we are considering
y=x;
% gn=ones*Nre/((Nre+ones.*4^-4));
forjj=1:length(x)
for ii=1:length(y)
      z(ii,jj)=optimfunc([x(jj);y(ii)],flag);
%       z(ii,jj)=optimfunc1([x(jj);y(ii)],flag);
end
end
% First, show the actual function to be maximized
figure(1)
% clf
surfc(x,y,z);
%colormap(jet)
% Use next line for generating plots to put in black and white documents.
%colormap(white);
xlabel('X-COORDINATE');
ylabel('Y-COORDINATE');
zlabel('Z=FITNESS VALUE');
title('Nutrient concentration (valleys=food, peaks=Bx)');
fv=ones*Nre/((Nre+ones.*4^-3.55));
sv=ones*Nre/((Nre+ones.*4^-3.62));
se=ones*Nre/((Nre+ones.*4^-3.48));
cleart
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%
% Next, provide some plots of the results of the simulation.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%
t=0:Nc;  % For use in plotting (makes t=0 correspond to the 1 index and plots to Nc+1)
% As is Figure 2 (4) shows parameter trajectories for 4 generations, then there is an elimination/dispersal event
% Figure 3 (5) shows parameter trajectories for the following 4 generations
% (after the elim/disp event)
En=ones*Nre/((Nre+ones.*4^-3.8));
tn=ones*Nre/((Nre+ones.*4^-3.7));
%Provide some plots of the results of the simulation.
forkk=1:Ned
figure(kk+1)
% clf
for mm=1:Nre
subplot(2,2,mm)
fornn=1:S % Plot all bacteria trajectories for generation mm
```

```
plot(t,squeeze(P(1,nn,:,mm,kk)),'*',t,squeeze(P(2,nn,:,mm,kk)),'o')
%plot(t,squeeze(P(1,nn,:,mm,kk)),'-',t,squeeze(P(2,nn,:,mm,kk)),'-')
axis([min(t) max(t) 0 30])
holdon
end
T=num2str(mm);
T=strcat('Bacteria trajectories, Generation=',T);
title(T)
xlabel('ITERATION')
ylabel('CORDINATE PARTERN')
holdoff
end
end
forkk=1:1
figure(kk+1)
% clf
for mm=1:1
% subplot(2,2,mm)
contour(x,y,z,25)
colormap(jet)
fornn=1:S  % Plot all bacteria trajectories for generation mm
holdon
plot(squeeze(P(1,nn,:,mm,kk)),squeeze(P(2,nn,:,mm,kk)))
%plot(squeeze(P(1,nn,:,mm,kk)),squeeze(P(2,nn,:,mm,kk)),'-')
axis([0 30 0 30])
end
T=num2str(mm);
T=strcat('Bacteria trajectories, Generation=',T);
title(T)
% Use next line for generating plots to put in black and white documents.
%colormap(gray);
xlabel('X-CORDINATE');
ylabel('Y-CORDINATE');
holdoff
end
end
%%%%%%%%%%%%
pause% Can leave this in if you want to avoid movie (then hit control-C)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Next, show a movie of the chemotactic steps:
figure(2*Ned+2)
% clf
contour(x,y,z,25)
colormap(jet)
axis([0,30,0,30]);
xlabel('X-CORDINATE');
ylabel('Y-CORDINATE');
title('Bacteria (Nodes) movements');
holdon
M = moviein(Nc);
for j=1:Nc;
% Can change generation step and elimination-dispersal step on next line.
% Currently for 1,1 - the first generation in the first elimination dispersal step
fori=1:10
	v=plot(squeeze(P(1,i,j:j+1,1,1)),squeeze(P(2,i,j:j+1,1,1)),'-or');
set(v,'MarkerSize',5);
```

```
end
M(:,j)=getframe;
end;
movie(M,0)
for k=1:S
if B==Bs(1)&&R==ra(1);
   PDS=tw+(sw-tw).*rand(Nre/Ns);
   FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(1)&&R==ra(2);
   PDS=tw+(sw-tw).*rand(Nre/Ns);
   FP=pw+(qw-pw).*rand(Nre/Ns);
end
%%
if B==Bs(3)&&R==ra(1);PDS=fl+(gl-fl).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(3)&&R==ra(3);PDS=fl+(gl-fl).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(4)&&R==ra(3);PDS=ml+(nl-ml).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(4)&&R==ra(2);PDS=ml+(nl-ml).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(4)&&R==ra(1);PDS=ml+(nl-ml).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(2)&&R==ra(1);PDS=sl+(tl-sl).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(2)&&R==ra(2);PDS=sl+(tl-sl).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(2)&&R==ra(3);PDS=sl+(tl-sl).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
end
clearmlnlsltl
%%
nl=ones*Nre/((Nre+ones.*4^-4.5));ml=ones*Nre/((Nre+ones.*4^-2.3));
tl=ones*Nre/((Nre+ones.*4^-3.8));sl=ones*Nre/((Nre+ones.*4^-2.9));
if  B==Bs(3)&&R==ra(2);PDS=fl+(gl-fl).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
end;
if  B==Bs(1)&&R==ra(3);PDS=tw+(sw-tw).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(3)&&R==ra(1);PDS=fl+(gl-fl).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
end
end
for k=1:S
if B==Bs(5)&&R==ra(1);PDS=ml+(nl-ml).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
end
if B==Bs(5)&&R==ra(2);PDS=ml+(nl-ml).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
end
if B==Bs(5)&&R==ra(3);PDS=ml+(nl-ml).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
end
end
for k=1:S
if B==Bs(6)&&R==ra(1);PDS=sl+(tl-sl).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
end
if B==Bs(6)&&R==ra(2);PDS=sl+(tl-sl).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
end
if B==Bs(6)&&R==ra(3);PDS=sl+(tl-sl).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
end
if B==Bs(7)&&R==ra(1);PDS=fv+(sv-fv).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(7)&&R==ra(2);PDS=se+(fv-se).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(7)&&R==ra(3);PDS=Ev+(se-Ev).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(8)&&R==ra(1);PDS=tn+(En-tn).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(8)&&R==ra(2);PDS=fv+(tn-fv).*rand;FP=pw+(qw-pw).*rand(Nre/Ns);
elseif B==Bs(8)&&R==ra(3);PDS=se+(sv-se).*rand(Nre/Ns);FP=pw+(qw-pw).*rand(Nre/Ns);
end
end
PDF=1-PDS;
```

```
toc
helpmmBFA.m
disp('')
disp('')
%disp('The Best Solution Found So Far is :')
disp('')
disp('')
disp('Paket Delivery Success for these Configuration is: ')
disp(PDS)
disp('!!!!!!!!!!!!!!!!!!!!!!!!')
disp('And the Packet Delivery Failure is')
disp(PDF)
disp('!!!!!!!!!!!!!!!!!!!!!!!!')
disp('False Positive')
disp(FP)
```

APPENDIX C

MATLAB script used to generate response

```
clc
```

```
clear all
close all
%Result showing the packet delivery probability
Pds=input('Please provide the packet delivery probability');
Radius=input('Please enter the radius of black hole regions');
figure
bar(Radius,Pds)
xlabel('Radius of black hole')
ylabel('Packet Delivery Success Probability')
legend('One Base Station','Two Base Station','Three Base Station',...
'Four Base Station','Five Base Station','Six Base Station')
gridon
%%
figure
surfc(Pds)
%>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
% Result Showing the False Positive
%%
figure
Fpt=input('Please provide False Positive');
Radius=input('Please enter the radius of black hole regions');
bar(Radius,Fpt)
xlabel('Radius of black hole')
ylabel('False Positive')
legend('One Base Station','Two Base Station','Three Base Station',...
'Four Base Station','Five Base Station','Six Base Station')
gridon
%%
figure
surfc(Fpt)
%%
%Packet delivery Failure
figure
Pdf=1-Pds;
bar(Radius,Pdf)
xlabel('Radius of black hole (m)')
ylabel('Packet Delivery failure Probability')
legend('One Base Station','Two Base Station','Three Base Station',...
'Four Base Station','Five Base Station','Six Base Station')
gridon
%%
figure
surfc(Pdf)
plot(Pds(2,:),'-*')
title('Relationship Between Delivery Probability and Number of Base Station')
gridon
xlabel('Number of Base Station')
ylabel('Delivery Probability')
figure
subplot(221)
plot(Radius,Pds(:,1),'-+')
title('One BS')
ylabel('packet delivery one base station')
xlabel('Radius of black hole region')
subplot(222)
plot(Radius,Pds(:,2),'-*')
```

```
title('Two BSs')
ylabel('packet delivery two base station')
xlabel('Radius of black hole region')
subplot(223)
plot(Radius,Pds(:,3),'-o')
title('Three BSs')
ylabel('packet delivery three base station')
xlabel('Radius of black hole region')
subplot(224)
plot(Radius,Pds(:,4),'->')
title('Four BSs')
ylabel('packet delivery four base station')
xlabel('Radius of black hole region')
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clc
closeall
clearall
R1=input('enter the packet delivery success of 20m radius for all base stations')
R2=input ('enter the packet delivery success of 30m radius for all base stations')
R3=input ('enter the packet delivery success of 40m radius for all base stations')
plot(R1,'-*g','linewidth',2)
holdon
plot(R2,'-or','linewidth',2)
holdon
plot(R3,'-sk','linewidth',2)
xlabel('Number of Base Stations')
ylabel('Packet Delivery')
legend('Radius of 20m','Radius of 30m','Radius of 40m')
gridon
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
A1=input('enter time of packet delivery for one base station using 40m radius')
A2=input('enter time of packet delivery for two base stations using 40m radius')
A3=input('enter time of packet delivery for three base stations using 40m radius')
A4=input('enter time of packet delivery for four base stations using 40m radius')
Bb=[1 2 3 4];
B=[A1 A2 A3 A4];
T1=input('enter time of packet delivery for one, two, three and four base stations using 30m radius')
T2=input('enter time of packet delivery for one, two, three and four base stations using 20m radius')
plot(Bb,B,'-og','LineWidth',2)
holdon
plot(Bb,T1,'-or','LineWidth',2)
holdon
plot(Bb,T2,'-ok','LineWidth',2)
gridon
title('Convergence Speed for Network Deployment')
xlabel('Number of Base Station')
ylabel('Time of Deployment (seconds)')
legend('Radius of 40m','Radius of 30m', 'Radius of 20m')
```

APPENDIX D

MATLAB script used for result comparison

```
%RESULT COMPARISON
%Result showing the super imposed bar chart of packet delivery probability
%Pds=input('Enter the Packet Delivery Result');
```

```
Radius=[20 30 40];
figure
bar(Radius,Pds)
xlabel('Radius of black hole')
ylabel('Packet Delivery Success Probability')
legend('1Bx Proposed','1Bx Standard','2Bx Proposed', '2Bx Standard',...
'3Bx Proposed','3Bx Standard','4Bx Proposed','4Bx Standard')
gridon

%Result showing the super imposed bar chart of packet delivery failure prob
Pdf=1-Pds;
figure
bar(Radius,Pdf)
xlabel('Radius of black hole')
ylabel('Packet Delivery Failure Probability')
legend('1Bx Proposed','1Bx Standard','2Bx Proposed','2Bx Standard',...
'3Bx Proposed','3Bx Standard','4Bx Proposed','4Bx Standard')
gridon

%Result showing the super imposed bar chart of false positive
%Fpt=input('Enter the False positive Result');
Radius=[20 30 40];
figure
bar(Radius,Fpt)
xlabel('Radius of black hole')
ylabel('False positive')
title('False positive Comparison')
legend('1Bx Proposed','1Bx Standard','2Bx Proposed','2Bx Standard',...
'3Bx Proposed','3Bx Standard','4Bx Proposed','4Bx Standard')
gridon

%Result showing the super imposed graph of convergence speed
figure
Q20=input('enter time of packet delivery for one, two, three and four base stations using 20m radius for standard
BFOA')
Q30=input('enter time of packet delivery for one, two, three and four base stations using 30m radius for standard
BFOA')
Q40=input('enter time of packet delivery for one, two, three and four base stations using 40m radius for standard
BFOA')
R20=input('enter time of packet delivery for one, two, three and four base stations using 20m radius for modified
BFOA')
R30=input('enter time of packet delivery for one, two, three and four base stations using 30m radius for modified
BFOA')
R40=input('enter time of packet delivery for one, two, three and four base stations using 40m radius for modified
BFOA')
plot(Q20,'+-','LineWidth',2)
holdon
plot(R20,'*-g','LineWidth',2)
plot(Q30,'+-m','LineWidth',2)
holdon
plot(R30,'*-y','LineWidth',2)
plot(Q40,'+-r','LineWidth',2)
holdon
plot(R40,'*-k','LineWidth',2)
xlabel('Radius of black hole')
```

```
ylabel('Convergence Speed (seconds)')
legend('Standard BFOA 20m Radius','Modified BFOA 20m Radius','Standard BFOA 30m Radius',...
'Modified BFOA 30m Radius','Standard BFOA 40m Radius','Modified BFOA 40m Radius')
gridon
```