Republic of Iraq

Ministry of Higher Education and Scientific Research

University of Babylon

College of Information Technology

Department of Networks

Modeling the Coverage Area of Wireless Sensor Network Based on Regression

A Thesis

Submitted to the Council of the College of Information Technology, University of Babylon in Partial Fulfillment of the Requirements for the Degree of Master in Information Technology/networks

By

Zainab Qassim Mohammed Ali Hadawi

Supervised by

Prof. Dr. Saad Talib Hasson Aljebori

2022 A.C. **1444 A.H.**

بِسْمِ اللَّهِ الرَّحْمَٰنِ الرَّحِيمِ

۞ اللَّهُ نُورُ السَّمَاوَاتِ وَالْأَرْضِ مَثَلُ نُورِهِ كَمِشْكَاةٍ
فِيهَا مِصْبَاحٌ الْمِصْبَاحُ فِي زُجَاجَةٍ الزُّجَاجَةُ كَأَنَّهَا
كَوْكَبٌ دُرِّيٌّ يُوقَدُ مِنْ شَجَرَةٍ مُبَارَكَةٍ زَيْتُونَةٍ لَا
شَرْقِيَّةٍ وَلَا غَرْبِيَّةٍ يَكَادُ زَيْتُهَا يُضِيءُ وَلَوْ لَمْ تَمْسَسْهُ نَارٌ
نُورٌ عَلَىٰ نُورٍ يَهْدِي اللَّهُ لِنُورِهِ مَنْ يَشَاءُ وَيَضْرِبُ اللَّهُ الْأَمْثَالَ
لِلنَّاسِ وَاللَّهُ بِكُلِّ شَيْءٍ عَلِيمٌ ۞

صدق الله العظيم

سورة النور الآية 35

Supervisor Certification

I certify that this thesis was prepared under my supervision at the Department of network, Collage of Information Technology, University of Babylon, by **Zainab Qassim Mohammed Ali** as a partial fulfillment of the requirements for the degree of **Master in Information Technology**.

Signature:

Name: **Prof. Dr. Saad Talib Hasson**

Title: **Professor.**

Date: / / 2022

The Head of the Department Certification

In view of the available recommendation, we forward this thesis for debate by the examining committee.

Signature:

Name: **Prof. Dr. Saad Talib Hasson**

Title**:** **Professor.**

Date: / / 2022

Declaration

I hereby declare that this thesis, submitted to the University of Babylon in partial fulfillment of the requirement for the degree of Master in Information Technology \ network, has not been submitted as an exercise for a similar degree at any other University. I also certify that this work described here is entirely my own except for experts and summaries whose source is appropriately cited in the references.

Signature:

Name: Zainab Qassim

Date: / /2022

Dedications

This work is dedicated to…

My father,

I will always be your daughter, who is proud of you,
You are my best idol in life and I will not disappoint
you.

My mother,

I ask Allah to protect you from all evil you are the best
example of a kind, giving, friendly mother ever
And I will keep you in my heart.

My dear husband,

My companion, my support; you are the best thing that
has ever happened to me. I ask Allah to take special
care of you.

My beautiful daughter

My brothers and close friends who stand by me when
things look very difficult.

Acknowledgements

In the name of God, the Most Gracious, the Most Merciful

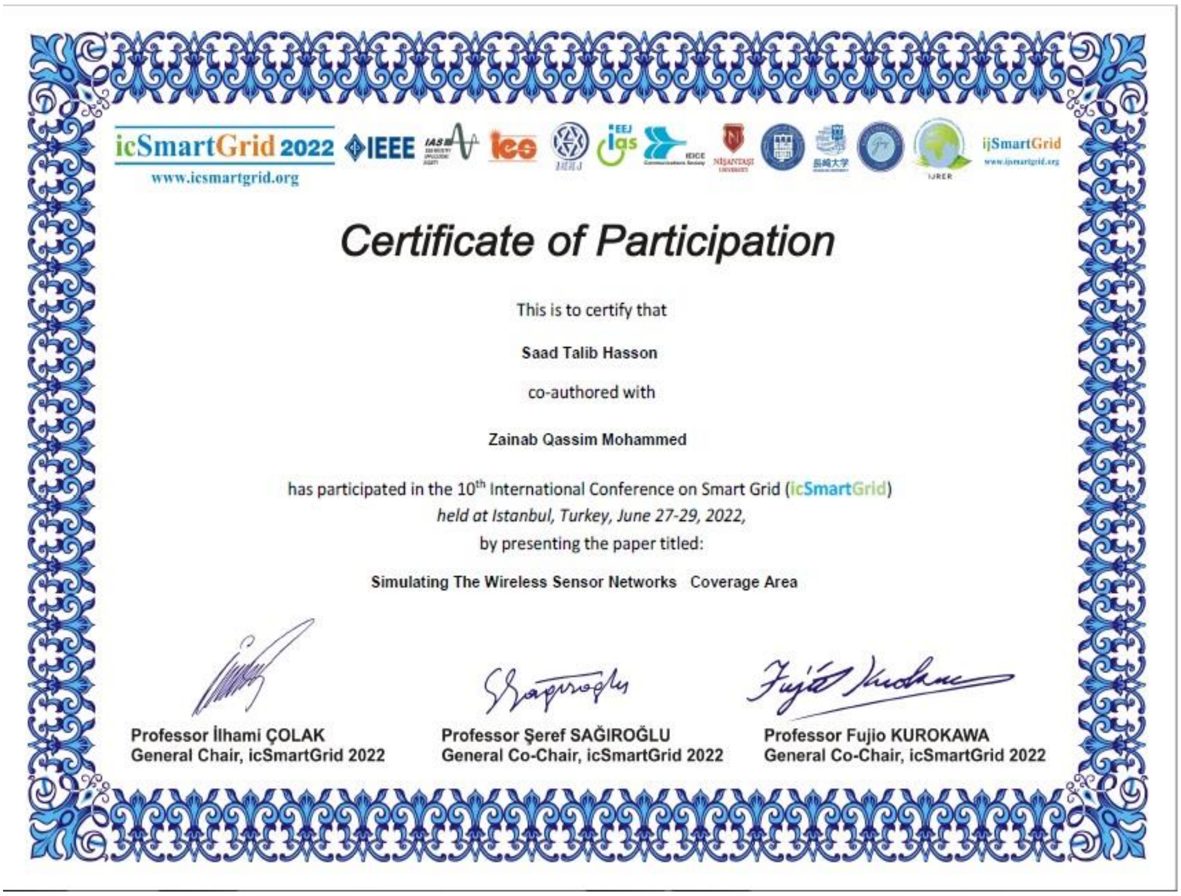
Foremost, I am highly grateful to Allah (God) for His unlimited blessings that continue to flow into my life, and for assisting me to face the difficulty that I met throughout achieving my aims.

I would like to express my sincere thanks and appreciation to my supervisor (**Dr. Saad Talib Hasson**) who supported me from the beginning until the completion of the current thesis. I greatly appreciate his optimistic behaviour, valuable advice, and trust in me that have always encouraged me to complete this work. I am totally sure that this work would have never become true without his encouragement and guidance.

Words cannot express my appreciation to my wonderful parents, the best father and mother. I thank them for their real guidance, advice, invitations, and endless support.

I would like to express my gratitude to my beloved husband for believing in me, and for his constant support, encouragement, and cooperation at all times.

Zainab Qassim Mohammed Ali



icSmartGrid 2022
www.icsmartgrid.org

Certificate of Participation

This is to certify that

Saad Talib Hasson

co-authored with

Zainab Qassim Mohammed

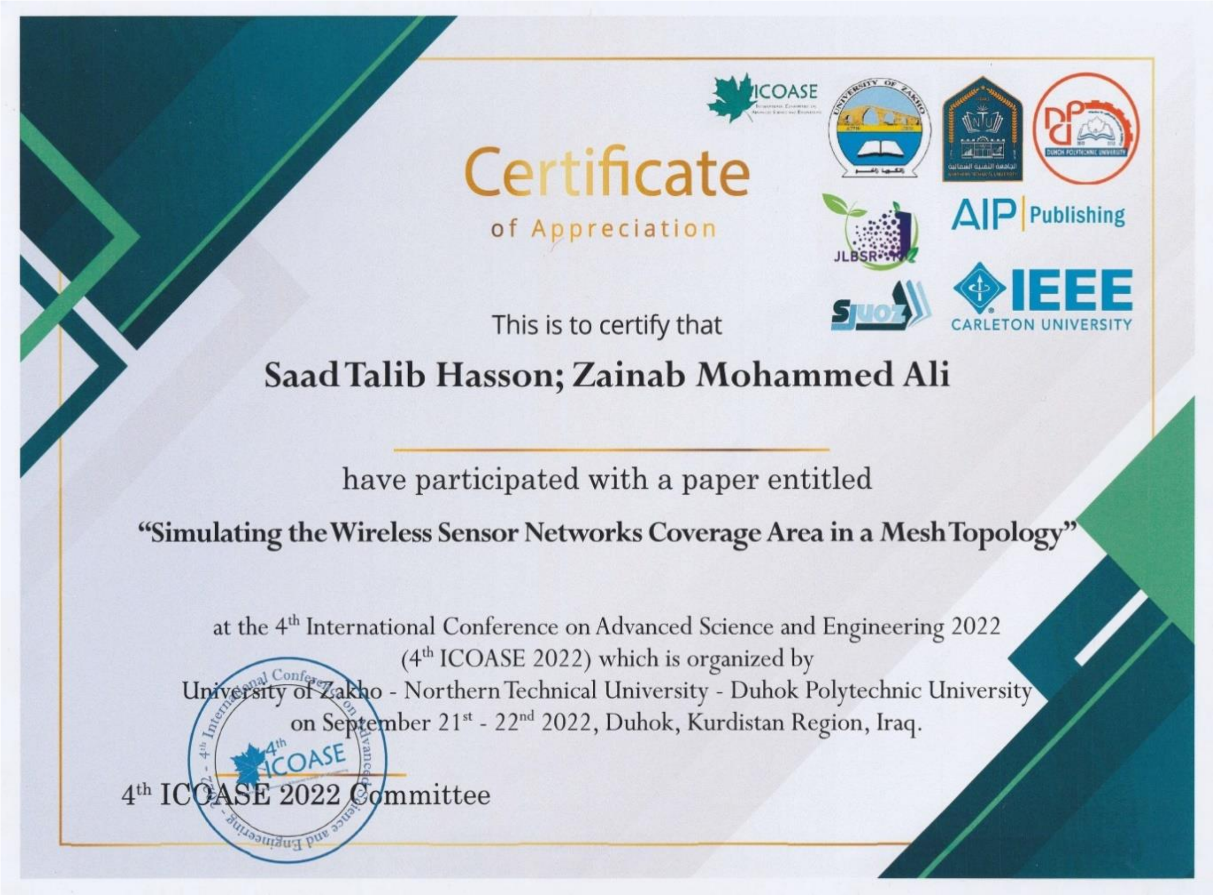
has participated in the 10th International Conference on Smart Grid (icSmartGrid)
held at Istanbul, Turkey, June 27-29, 2022,
by presenting the paper titled:

Simulating The Wireless Sensor Networks Coverage Area

Professor İlhami ÇOLAK
General Chair, icSmartGrid 2022

Professor Şeref SAĞIROĞLU
General Co-Chair, icSmartGrid 2022

Professor Fujio KUROKAWA
General Co-Chair, icSmartGrid 2022


ICOASE
AIP | Publishing
JLBSR
SJUOZ
IEEE
CARLETON UNIVERSITY

Certificate

of Appreciation

This is to certify that

Saad Talib Hasson; Zainab Mohammed Ali

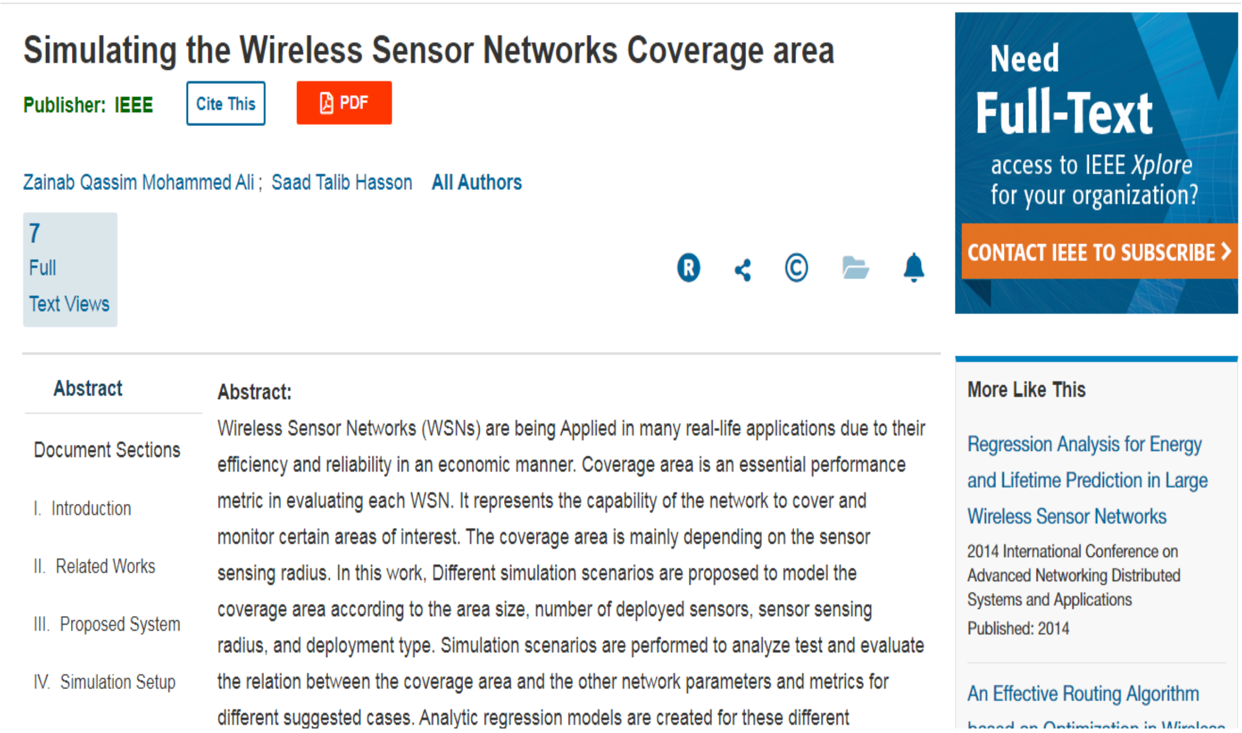
have participated with a paper entitled

"Simulating the Wireless Sensor Networks Coverage Area in a Mesh Topology"

at the 4th International Conference on Advanced Science and Engineering 2022
(4th ICOASE 2022) which is organized by
University of Zakho - Northern Technical University - Duhok Polytechnic University
on September 21st - 22nd 2022, Duhok, Kurdistan Region, Iraq.

4th ICOASE 2022 Committee

4th International Conference on Advanced Science and Engineering 2022
4th ICOASE



Simulating the Wireless Sensor Networks Coverage area

Publisher: IEEE

Cite This

PDF

Zainab Qassim Mohammed Ali ; Saad Talib Hasson All Authors

7
Full
Text Views

Abstract

Document Sections

Abstract:

Wireless Sensor Networks (WSNs) are being Applied in many real-life applications due to their efficiency and reliability in an economic manner. Coverage area is an essential performance metric in evaluating each WSN. It represents the capability of the network to cover and monitor certain areas of interest. The coverage area is mainly depending on the sensor sensing radius. In this work, Different simulation scenarios are proposed to model the coverage area according to the area size, number of deployed sensors, sensor sensing radius, and deployment type. Simulation scenarios are performed to analyze test and evaluate the relation between the coverage area and the other network parameters and metrics for different suggested cases. Analytic regression models are created for these different

More Like This

Regression Analysis for Energy and Lifetime Prediction in Large Wireless Sensor Networks

2014 International Conference on Advanced Networking Distributed Systems and Applications

Published: 2014

An Effective Routing Algorithm based on Optimization in Wireless

Abstract

Wireless Sensor Networks (WSNs) have many crucial applications. Sensors are mainly utilized to track or monitor any modifications in a surrounding area. They can be used to monitor, gather information, then process and transfer the sensed data to a certain sink node.

Coverage area is an essential performance metric in evaluating each WSN. It represents the capability of the network to cover and monitor certain areas of interest. The coverage area is mainly depending on the sensor sensing radius.

In this work, Different simulation scenarios are proposed to model the coverage area according to the area size, number of deployed sensors, sensor sensing radius, and deployment type. Simulation scenarios are performed to analyze test and evaluate the relation between the coverage area and the other network parameters and metrics for different suggested cases. Analytic regression models are created for these different suggested cases. These models are suitable to be used accurately without any need to repeat, execute or re-run the simulation programs. The suggested analytical equations show accurate results during their training and testing examples. The coverage area is depending on the sensing radius, the number of deployed sensors, the deployment approach (random or predefined), and the deployed area. The effect of these parameters on the coverage area is simulated and tested in different scenarios. Different regression curves and their equations are created and tested. These equations can be accurately used to estimate the coverage values.

Declaration Associated with this Thesis

(First Paper)

- **Name of Journal/Conference:** The 4th International Conference on Advanced Science and Engineering 2022
- **Paper Title:** Simulating The Wireless Sensor Networks Coverage Area
- **Publication:** IEEE Xplore
- **Authors:**
    - Saad Talib Hasson
    - Zainab Qassim Mohammed Ali

Information Networks Department, Information Technology College, Babylon University.

(Second Paper)

- **Name of Journal/Conference:** The 10th International Conference on Smart Grid
- **Paper Title:** Simulating the Wireless Sensor Networks Coverage Area in a Mesh Topology
- **Publication:** IEEE Xplore
- **Authors:**
    - Saad Talib Hasson
    - Zainab Qassim Mohammed Ali

Information Networks Department, Information Technology College, Babylon University.

Table of Contents

CHAPTER ONE: INTRODUCTION

CHAPTER TWO: COVERAGE IN WSN

CHAPTER THREE: PROPOSED SYSTEM

CHAPTER FOUR: SIMULATION RESULTS AND ANALYSIS

CHAPTER FIVE: CONCLUSIONS AND FUTURE WORKS

List of Tables

List of Figures

List of Algorithms

List of Abbreviations

C	The coverage
d	The distance between two points
N	Number of nodes
PSM	Probability sensing model
R_c	The communication range
R_s	The sensing range
ROI	Region of interest
WSN	Wireless sensor network

CHAPTER ONE

Introduction

Chapter one

Introduction

1.1 Introduction

Wireless sensor network (WSN) represents a monitoring tool. It was utilized in significant applications such as: tracking, disaster management, biomedical health checking, surveillance, environmental observation, targeting systems, military monitoring, and healthcare (Singh et al., 2018). The sensor nodes work together to collect data concerning the area of interest (Aziz et al., 2009)

A sensor node involves a unit of sensing, a radio transceiver, a processing unit and the power management section. Sensor nodes generate some measurable responses and send these responses to a communal sink (also called a base station) over a wireless channel. A wireless sensor network's nodes usually have limited energy, as their batteries are limited and cannot be recharged(Jindal DAV College, 2018).

Sensors are characterized by a very constrained power supply. Sensors nodes are usually randomly deployed in a designated area. If there are enough sensors, each sensor node can take turn performing the task of the sensing. As a result, scheduling sensors becomes crucial to extending the WSNs' network lifetime (Seah et al., 2009). The sensing (coverage) of the sensor and the network connectivity issues are as important as the scheduling issue. The coverage of the sensor refers to the ability of at least one sensor node to monitor each point within a deployed field (Elhoseny et al., 2017).

Network coverage represents the significant issue in WSN. It indicates how effectively a particular region of interest is being monitored (detected) by sensors. Usually, the sensing range of each sensor node is limited (Priyadarshi & Gupta, 2020). Any event is supposed to be detectable if it lies within the coverage area of at least one sensor node. A node will cover a circular area with radius equal to the sensor coverage range (Amutha et al., 2020).

Several researchers have studied the coverage for WSN. The majority of them have considered the coverage for wireless sensor networks that are randomly deployed (Sharma et al., 2016).

Two nodes will have an ability to communicate with each other if the distance between them is less or equal to the transmission range of the other. In other words, if there is a link between them, they can be considered as neighbor nodes (Guo & Jafarkhani, 2016).

Each sensor node has the ability to collect data, process it, and connect with its neighboring sensor nodes (Kocakulak & Butun, 2017). An effective sensor deployment strategy should take connection and coverage into account. The deployment of sensors determines how successfully a specific area will be monitored as well as how much it will cost to build the network (Tripathi et al., 2018).

This thesis assumed that the sensing radius for each node is of a disk shape and all sensor nodes (are homogenous) having the same sensing range with certain communication range.

If for example the variable b is assumed to represent the base station location in a certain area, the variable (s_k) be a sensor node (k) in a WSN, then the distance d(b, s_k) be the Euclidean distance between b and s_k (Hossain et al., 2008).

It's important to appropriately deploy sensors in the target area for the coverage to be effective. However, nodes might not be placed in the desired locations in harsh environments like disaster areas, underwater, and on battlefields. Instead, sensors are usually distributed across the area in random (Boualem et al., 2018).

1.2 Problem statement

One of the main problems to be considered in proposing and creating a wireless sensor network is finding a way to improve the performance of the network. For effective utilization of WSNs, a number of fundamental issues, including localization, security, tracking, placement, and aggregation, must be addressed.

Sensing problem is an example of a more general problem of configuring sensor parameters. It means that each point in the deployed area must be covered by at least one sensor. Depending on the application of a deployed WSN and the sensor type being used. The most important of them are the sensing range, communication range, sensors location in addition to the deployment problem.

The sensing model mainly depends on the target distance, orientation, and visibility. The sensing model must facilitate the analysis and helps understand the coverage problem. The network based on different deployment approaches can be utilized to create a developed sensing model. Any sensing model must ensure certain level of coverage with less sensing working nodes and ensure the connectivity of the network.

1.3 Thesis Aims

The main aim of this thesis is to improve the WSN performance by modeling and simulating the coverage area criterion. This aim can be achieved by the following objectives:

1. Study the fundamentals of coverage area and analyzing the random sensors deployment approaches.

2. Estimate the optimal number of deployed sensors to cover certain area.
3. Develop a model to measure the network coverage levels.
4. Evaluate the effects of the proposed models on the network performance.
5. Creating regression models to estimate the coverage area.

1.4 Related Works

There are different related works related to the coverage area in WSNs. Most recent of these are summarized in the following:

Author & year	**Problem**	**Tools**	**Results**
Mini & Pal, 2021	Earlier researchers have ignored the impact of BEs in the presence of sensor failure while evaluating the coverage performance of WSNs.	Simulation	Created a closed form expression for the k-coverage performance. They derived an analytical model based on a sensor failure. Their proposed model results with a Root Mean Square Error of less than 0.03 and a decreasing the k-coverage probability with increasing in k.

Kim & Yoo, 2020	This paper formulates the sensor target coverage problem into an object function and solves it with an NIA.	Bat algorithm	Simulation results show that the proposed method gives better results with respect to other related work in terms of the network lifetime.
Tarnaris et al., 2020	The optimization of the performance of wireless sensor networks in terms of area coverage	Computation al intelligence algorithms	Their performance was evaluated via comparative simulation tests. This appraisal was made using statistical testing. The test results, that proved the efficacy of the algorithms proposed.
Chen & Chen, 2018	Pre-defined deployment under probabilistic sensing model.	Direct approach	Their study showed that more than 86% of the deployed sensors can be saved.
Qin & Chen, 2018	Lifetime requirements and coverage demands	An area coverage algorithm based on differential evolution	Their simulation results showed that the coverage algorithm based on differential evolution has provided a coverage area of about 90%.

Fen et al., 2016	How to completely cover the network without knowing the locations of the nodes	Coverage control model for randomly deployed sensors.	Their results showed that the quality of the coverage is higher than the threshold coverage quality.
Akbarzadeh et al., 2013	Sensor placement for the line-of-sight sensors based on the coverage area	Covariance Matrix Adaptation Evolution Strategy and simulated annealing as optimization techniques.	Their findings demonstrated a maximum in the coverage of a global sensor network others in terms of the network lifetime.
Duoqiang et al., 2012	Reduce the number of working nodes, save networks' residual energy, and prolong the life of networks.	Simulation	The agreement minimizes the number of active nodes and successfully prolongs the life cycle of the network to provide the preferred coverage.
Wang & Tseng, 2008	The distributed dispatch issue and the k-coverage placement issue	Simulation	Their simulation results outcomes were used to prove their network effectiveness.

1.5 Thesis Layout

Additional for this chapter, thesis contains four chapters layout as following:

Chapter two describes the coverage in WSN theories.

Chapter three displays how to simulates the coverage of the selected area in wireless sensor networks using NetLogo.

Chapter four holds results of the proposed system and its evaluation.

Chapter five displays the results conclusions and the future works.

CHAPTER TWO

Coverage in Wireless Sensor Network

Chapter Two

Coverage in WSN

2.1 Overview

Wireless sensor networks (WSNs) are created by deploying self-organized, separately sensors. WSN is involving of a large number of sensor nodes distributed at random in specific locations (Chang & Pakzad, 2014). The size and power of sensor nodes are modest, and they have limited processing power and computational capabilities (Aslam et al., 2009). The major purpose of the sensors is to acquire data on different factors like temperature, pressure, humidity, and other physical and environmental parameters (Toth & Jóźków, 2016). A base station or sink must receive the collected data. The main challenges in WSNs are the network scalability, fault tolerances, secure, reliable data transmission, robustness and need for efficient energy (Savitha & Shantala, 2014).

2.2 Wireless Sensor Networks (WSNs)

Wireless sensor networks (WSNs) usually made up of spatially distributed deployed separate, self-contained system units. WSN involves of a big number of randomly deployed sensor nodes in certain areas. Nodes of sensor are small in size with restricted power battery, limited computational and processing resources (Xie et al., 2018). The sensors are mostly used to keep track of physical and environmental variables, collect data such as temperature, humidity, pressure and so on. The gathered data must be transferred to a base station or sink (Abbasian Dehkordi et al., 2020). The main challenges in WSNs are the network scalability, fault tolerances, secure, reliable data transmission, robustness and need for efficient energy (Yildiz, 2017).

WSNs must be able to work in circumstances where there is no direct human monitoring. Sensor nodes are frequently placed at random in most WSN systems (Yildiz, 2017). Figure (2.1) A wireless sensor network's general architecture.

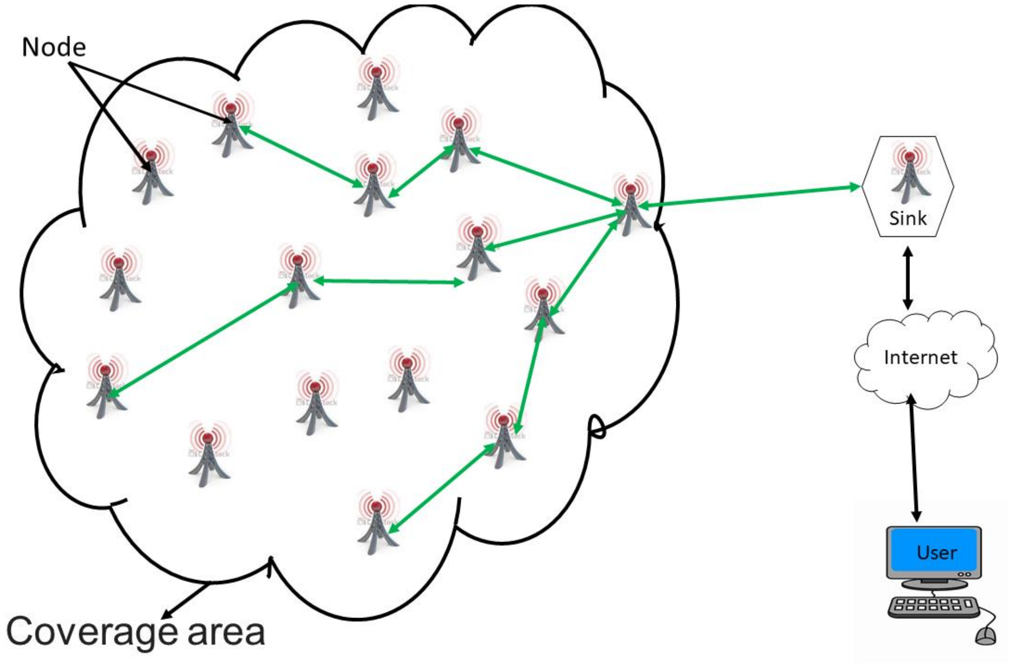


Figure (2.1): A wireless sensor network's general architecture (WSN).(Djedouboum et al., 2018)

2.3 Application of WSN

There is a wide range of fields the sensor networks are utilized, such as military applications, medical, environmental monitoring (Ramson & Moni, 2017). In general, Due to the fact sensor networks support the interaction between humans and material science to it, it would be all over the place in the near future. To get a lot of new applications to explore in the real world sensor nodes applied in the uncontrolled environment (Kassim & Harun, 2016). Deployment tightly sensing node with the ability to remote sensing, wireless communications, and the calculation in the uncontrolled environment, which helps in measuring the ambient conditions, and get the important characteristics surrounding this node, transforming sensed and collected data into electronic signals can be processed, is the idea behind this several applications (Parwez et al., 2017).

Sensor networks play an important role in the following applications (Parwez et al., 2017).

- Environmental applications
- Health care applications
- Agricultural applications
- Structural monitoring
- Intelligent home monitoring
- Military applications
- Industrial applications
- Vehicle detection
- Congestion control
- RFID indoor tracking system

2.4 Sensors

Sensor nodes are cheap, moveable and they have restricted energy resources and processing power. It probably works on gathering data from its surroundings; processing data locally then transfer sensed data one more time to worker (Zhang et al. 2018). The main five components of sensor nodes are: “computing unit”, “sensing unit”, “communication unit”, “power supply unit”, and “memory unit”. Figure (2.2) shows the architecture of wireless sensor (Feng,2009).

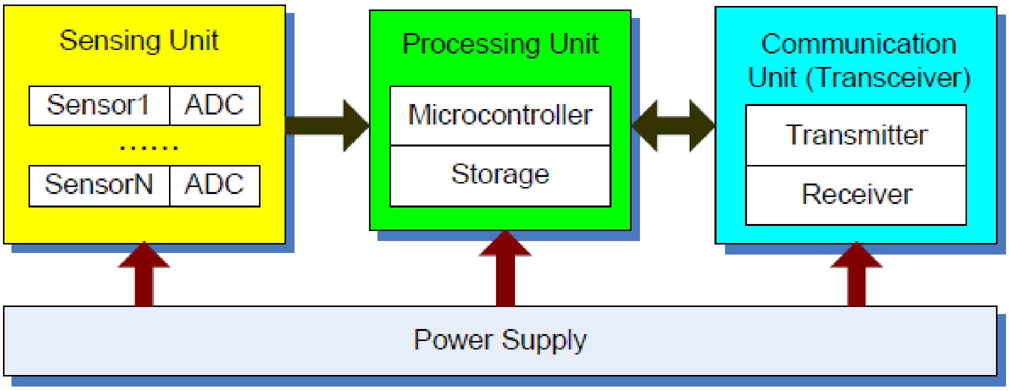


Figure (2.2): wireless sensor architected (Feng, 2009)

The computing unit contains a microprocessor. Management the communication protocols, handling gathered information of aboard sensors, besides accomplishment organization of power that run by the microprocessor (Balamurali & Kathiravan, 2015). A single communication unit found in each sensor node has the ability to transmit and receive messages. The functionality of both transmitter and receiver merge in this unit. Four operational states found in the communication unit. Sensing unit is frequently a sensor panel that contains single or extra sensors. Sensors necessity owns very little power consumption. Based on different needs for storage there are two types of memory units in a sensor node (Abo-zahhad et al., 2015). Some on-chip memory found in the microprocessor itself applied to save system software. Usually flash memory exists wherever workers save their personal applications and data. The power gets from the power unit and affords for other four units. Although all operations involving sensing, computation, and communication need energy. Data communication requires more energy than sensing and computation do. Thus, decreasing data transfer among sensor nodes probably advance energy effectiveness besides spread time of sensor networks (Bofu, 2010).

2.5 Sensing Models

There are two main types of sensing models:

2.5.1 Binary sensing model

It is very simple. Event detection is believed to have occurred if the event occurs within a node's sensing range; otherwise, it is not.

This model fails to take into account how the work of sensing is affected by the condition of the environment and the intensity of the signal that is being emitted (Hossain, 2014). The binary sensing model is shown in figure (2.3).

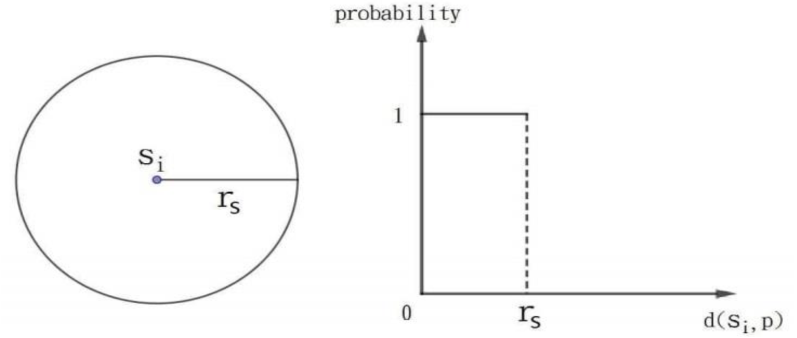


Figure (2.3) .The Binary sensing model (Hossain, 2014)

2.5.2 Probability Sensing model

The way in which the sensor detects its surrounding is known as sensing model. It provides a relationship between the probability of detecting a point and the space between the point and sensor. Sensing obey certain probability. The type of the Probabilistic Sensing Model (PSM) is continuous. It is more able to match the real data found in the real environment of the network and it can satisfy the network's quality (Hossain et al., 2012).

PSM transforms the real sensing signals into probability indication values to evaluate the level of confidence in the data that the sensor has gathered. Assume, k sensors were deployed during the first stage of sensor deployment in a sensor field grid with dimensions of m by n. R is the detection range for each sensor. Assume that sensors $s_i\,, 1 \leq i \leq k$ are positioned at a point (x_i,y_i) . For any point P(x,y) , we indicate the Euclidean distance between P and s_i as d(s_i,P) (Zorlu & Sahingoz, 2016).

The equation illustrates that the coverage c (P) of the grid point P by the sensor is expressed by the probability detection model as:

$$c_{s_i} = \begin{cases} 1, & d(si, P) \leq r \\ e^{-\lambda\alpha^{\beta}}, & r < d(s_i, P) \leq R \\ 0, & d(s_i\,, P) > R \end{cases}$$

Where $r < R$ represents the degree of certainty in sensor detection, $\alpha = d(s_i, P) - r$, and β, $\boldsymbol{\lambda}$ are parameters that represent the likelihood that a target will be detected while being farther away from the sensor than r. Grid point units are used to measure the distances. With various values for the parameters, β produces various translations that are expressed by various detection probabilities, which can be thought of as the features of different kinds of physical sensors. This model accurately illustrates how the range of the sensing devices, such as infrared and ultrasound sensors, behaves (Shan et al., 2016). Alternative sensor models may also be employed because the suggested target localization method is independent of the sensor model (Hossain, 2014). The probability sensing model is shown in figure (2.4).

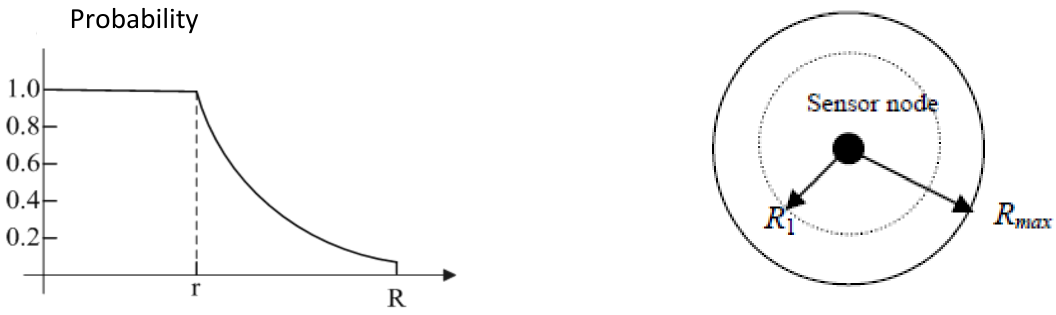


Figure (2.4) The probability sensing model (Hossain, 2014)

2.6 Simulation

Simulation is a process over time or artificial state of a real-world system's operation. When the actual system is costly and more difficult to build and reconfigure repeatedly, it becomes more important. As a result, the simulation can be used as a method to evaluate the system's performance under a variety of different constraints and arrangements that might exist in real life. The cost, danger, and failure condition associated with the actual system's construction operation can all be reduced by simulation (Goodall et al., 2019).

It can also be thought of as the process of scheming mathematical or logical model that matches to the real system, followed by using the computer to perform out several trials in order to predict and explain the performance of the real system. To simulate any system, we need a simulator. Various network simulators have varies property hand-me-down and practical. Several examples of network simulators are NetLogo, OPNET, NS2, and NS3 (Fawzia, 2014).

2.7 NetLogo

NetLogo is a multi-agent besides modeling environment programming language for simulating natural and communal event. It is mostly fine suitable to forming compound systems developing extra period. Hundreds or thousands of autonomous agents wholly working simultaneously get commands from the modelers. This makes exploring associates among micro-level performances of entities plus macro-level designs that appear by their communications possible (Wilensky & Rand, 2015)**.**

Users can open simulations through NetLogo and observe their performance under several conditions. NetLogo can be considered as authoring environment that it is simple enough to enable learners and academics to formula their personal models, and it was not professional program writer. NetLogo designed for both education and research (Railsback et al., 2017).

NetLogo can run on almost important computing stands and considered as individual application programmed in Java. It is a developed product that it is stable as well as fast afterward years of progress. It is freeware anyone can download it, free to construct models without constraint. It has a big group of examples of models, tutorials and broad certification. As a linguistic, NetLogo is a participant of the Lisp family as maintenances representatives and concurrency. Moveable agents named turtles travel above a net of patches, which are moreover programmable agents. All of the agents can interact with each other and implement multiple jobs simultaneously (Seth & Uri, 2004). Figure (2.5) shows the samples of NetLogo simulator.

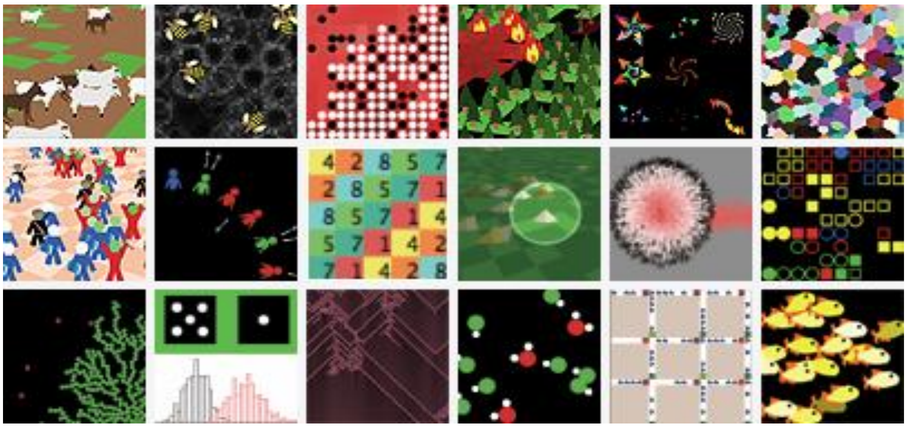


Figure (2.5): NetLogo samples [NetLogo user manual,2015]

NetLogo contains three main tabs a (interface, information, and code).

Interface tab used to display execution of your model. It also has tools used for reviewing and modifies what's going on inside the model. The Interface tab is empty when NetLogo is opened at first, except for observation, wherever turtles and patches look, and Command Center, lets you produce NetLogo commands. The toolbar in Interface tab holds buttons lease you edit, delete, also generate items on Interface tab and menu leases you to choose diverse interface items (like buttons plus sliders) figure (2.6) show the interface items.

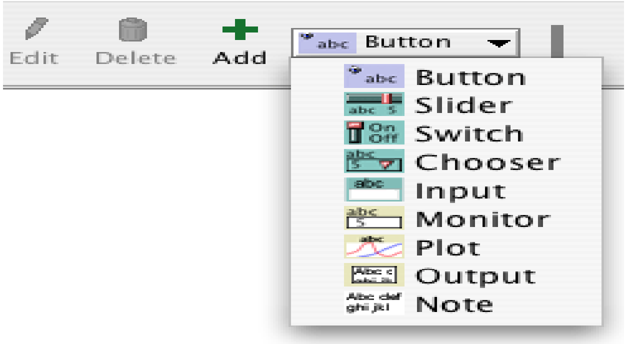


Figure (2.6): Interface item [NetLogo user manual, 2015].

Info tab offers an overview about models. It describes any structure its existence demonstrated, in what way model was created, plus in what way use it. It can moreover propose things for discovery besides methods for spread model, otherwise request your care to certain NetLogo structures model usages.

Code tab refer to the NetLogo programming language in aspect. [NetLogo user manual, 2015].

2.8 Dijekstra algorithm

The following steps are implemented to create the shortest path in a suggested wireless sensor network topology in this thesis.

The algorithm of Dijkstra is utilized to estimate the shortest path between a source and a destination node in a suggested graph network by estimating the distance to each node from another to the starting node. Each node in a graph is visited with all its neighbors to estimate the shortest sub-path from a node to its neighbors (Broumi et al., 2016). The detailed steps taken by Dijkstra's algorithm to determine the shortest path between a single source vertex and every other vertex in a given graph are shown in the figure (2.7).

Algorithm

1) Build a tree of shortest path called “*sptSet”*.

2) Create an input graph to assign a value for all vertices distances

3) For each vertices not included in *sptSet.*

a) Choose a not in *sptSet* vertex o with its minimum distance.

b) Add o to the *sptSet*

c) The distance value to all the neighboring vertices of o will be updated.

2.9 Correlation

A statistical metric known as correlation, which is given as a number, describes the direction and size of a relationship between two or more variables. If two variables are correlated, it does not necessarily imply that changing one of the variables led to a change in the values of the other variable (Veličković, 2015).

A common goal of scientific study or research is to determine how closely one variable is related to another.

Whether there is a correlation between the two variables; if so, this may serve as guidance for future research into determining whether one action causes the other. Understanding correlation enables programs and policies that attempt to bring about a desired result to be more precisely targeted.

A correlation coefficient is a metric for determining the strength of an association and it is represented by the letter r. It represents a linearity association measure. Correlation coefficient known as “Pearson's correlation”. If the relation is curvature (not linear), difficult measures must be applied.

The correlation coefficient is measured on a scale ranging from -1 to +1. The value of (±1) represents a variable's complete correlation with another .When one variable rises while the other falls, the correlation's value is negative; however, when both variables rise at once, it is positive. The number 0 represents a complete absence of correlation (Makowski et al., 2020).

2.10 Regression

The term "correlation" describes the degree of a relationship between two variables, and because it is perfectly symmetrical, the correlation between two variables A and B and B and A is the same. However, if there is a relationship between the two variables, it indicates that when one changes on average by a certain amount, the other variable also changes by a certain amount. Regression of y on x is the name given to this relationship when y serves as the dependent variable and x serves as the independent variable (Walpole et al., 2017).

A basic equation that can be used to express the relationship is the regression equation. The term "regression" basically indicates that the average value of a variable X is a "function" of a variable y, which alternate in accordance with the value of y.

On a scatter diagram, a regression line can be created using the regression equation. Regression equation represents a straight line to reflect the effect of change y on x in its simple case. The existence of a negative or positive correlation determines the slope of the line. The line slopes upward from left to right when both sets of observations increase or decrease simultaneously (positive); conversely, when one set decreases as the other set increases, the line slopes downward from left to right. Since the line must be straight, it will likely cross through few of the dots. Given that the relationship is well defined by a straight line, two characteristics of the line can be defined in order to properly position it on the diagram. The slope and the height above the baseline are the essential characteristics. There are various kinds of regression analysis, and in addition to recognizing the nature of variables and their distribution, assumptions must also be taken into account (Arkes, 2019).

- Polynomial Regression
- Linear Regression
- Logistic Regression

In this thesis the polynomial regression was used because simple linear regression algorithm only works when the relationship between the data is linear but suppose if we have non-linear data then Linear regression will not capable to draw a best-fit line and it fails in such conditions. Hence, we introduce polynomial regression to overcome this problem, which helps identify the curvilinear relationship between independent and dependent variables. Polynomial regression is a form of linear regression where only due to the Non-linear relationship between dependent and independent variables we add some polynomial terms to linear regression to convert it into Polynomial regression (Walpole et al., 2017).

Where X is suggested as independent variable and Y as dependent variable.

CHAPTER THREE

Proposed System

Chapter Three

Proposed system

3.1 Overview

In this chapter, the detailed descriptions of how to calculate the coverage of the wireless sensor network will be displayed. Certain research methodology and algorithms were used in calculating the coverage. Detailed descriptions of each procedure that used to estimate the coverage area will be displayed.

3.2 Research Methodology

NetLogo (a multi-agent modeling language) was suggested as a network simulator to be used in building and implementing the suggested wireless sensor networks.

The following steps illustrate the process of designing and implementing the suggested wireless sensor network in this thesis.

1. Propose and create a wireless sensor network using the NetLogo elements in its interface tab and select the suitable method for entering the required data and the process of objects display.
2. Select the environment area shape and size.
3. Create and call a number of suitable agents (turtles) to represent the sensor nodes in the network and assign their numbers (Node who).
4. The sensor nodes will be distributed randomly in the suggested area.
5. One of these nodes will be selected to be the sink node (base station).
6. The distance between each node and its neighbors will be calculated and they will be connected by links.

7. Select n nodes from the network randomly to represent the source (or the sensing nodes to start transmitting data) and estimate the coverage.
8. The shortest path from each node to the sink will be indicated.
9. The coverage area is estimated depending on the sensing range.
10. Start sending messages from n random nodes to the sink node throw the shortest path to check the coverage.

3.3 Modeling approaches

Modeling is the process of representing the wireless sensor networks systems to promote understanding of the real system behaviors. The following simulation program were designed to achieve the process of data transmission in WSNs. Figure (3.1) presents the general flowchart of the suggested simulation program model.

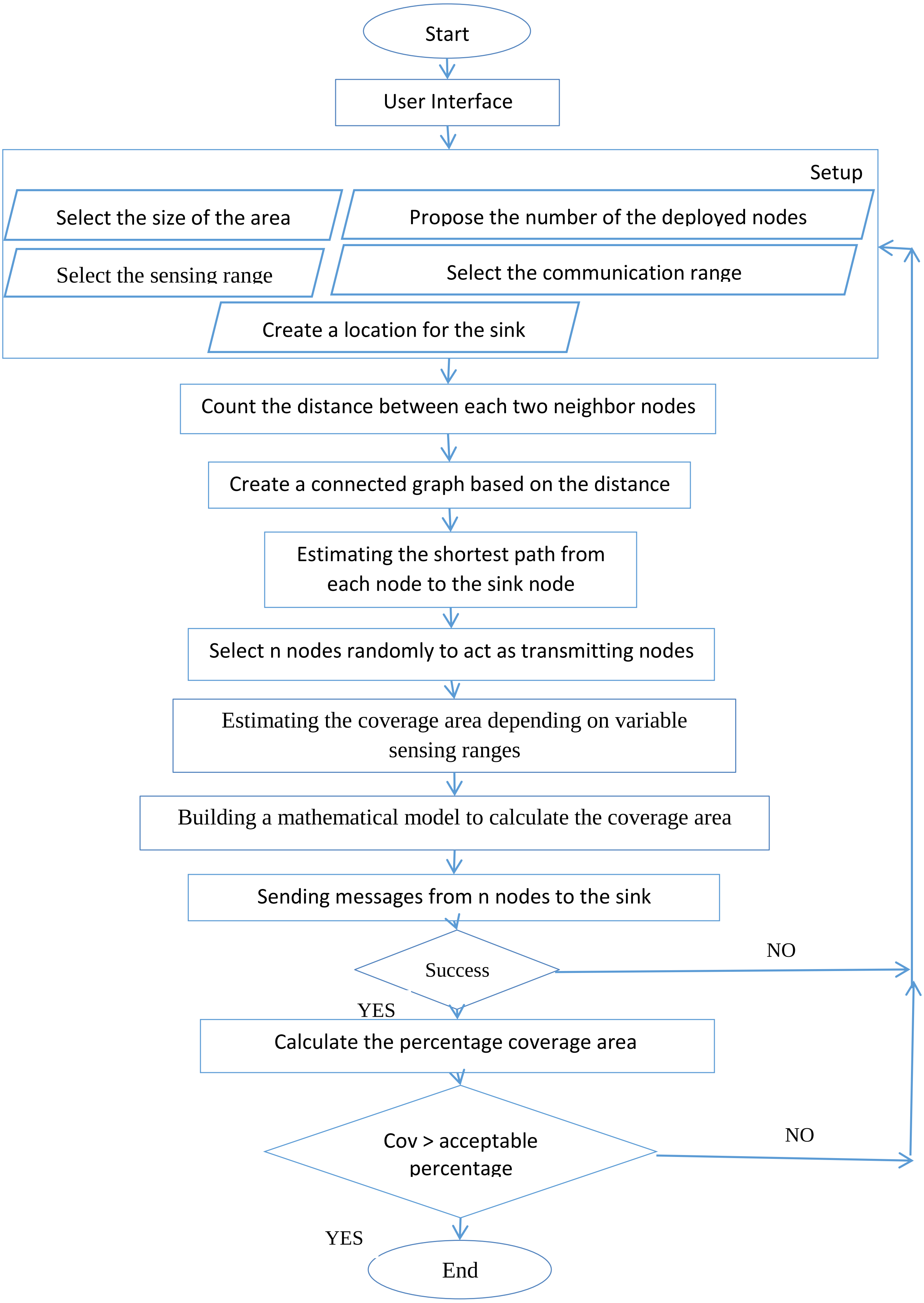

Start
User Interface
Setup
Select the size of the area
Propose the number of the deployed nodes
Select the sensing range
Select the communication range
Create a location for the sink
Count the distance between each two neighbor nodes
Create a connected graph based on the distance
Estimating the shortest path from each node to the sink node
Select n nodes randomly to act as transmitting nodes
Estimating the coverage area depending on variable sensing ranges
Building a mathematical model to calculate the coverage area
Sending messages from n nodes to the sink
Success
NO
YES
Calculate the percentage coverage area
Cov > acceptable percentage
NO
YES
End

Figure (3.1): flowchart of the simulation program.

3.3.1 Setup

In this thesis, a simulation process is started with a setup stage. Setup stage is to initializes the requirements to build and implement a proposed WSN environment. Successive steps are followed to perform the setup stage. These steps are as follows:

A. Select the designated area

In this thesis, the suggested Region of Interest (RoI) to be covered by deployed sensors is a two dimensional square area. It represents a suitable model that can be applied in several situations. Almost any area shape can be separated into set of squares areas. A "homogeneous" network is considered by utilizing sensors having similar characteristics. In this thesis four different sizes areas are studied.

B. Select the sensor's coverage and communication ranges

In this step the sensing range for each sensor also must be determined with testing the effect of using variable numbers for the sensing range at the coverage of the selected area. The communication range also determined to ensure the process of delivering the sent message to the sink.

C. Select the deployment approach

Random and predefined approaches are the two deployment approaches in creating a WSN. Random deployment is used in a remote or difficult to reach areas as well as when deploying very large number of sensors in a big or long area. Different random deployment approaches were developed and implemented in various simulators based on random probability distributions. In this thesis a NetLogo function is utilized to generate the random deployment for any number of sensors in each area size.

D. Select the number of sensors

The numbers of the deployed sensors are selected by the network creator or by the user in a simulation environment. This number is selected based on the area size and the application requirements. In this thesis, a NetLogo simulation environment is created and a slider is designed to select the required number of sensors in each simulation run called (Nodes-Number slider).

A mathematical model is proposed to estimate the approximate number of sensor nodes required to cover the selected area. The sensors number in this model is proposed based on the area, the coverage (sensing) and communication ranges, and level of coverage.

The model:

Z= area size (x * y)

R_s= coverage range

R_c= communication range

N= number of sensors

C= coverage level

One sensor coverage area = $\pi * R_s^2$

N sensors coverage area = $N * \pi * R_s^2$

$C = \frac{N * \pi * R_s^2}{x * y} * 100\%$ ……………………..… (3.1)

For example if C is required to be 75% then the required number of sensors to achieve this coverage level is:

$0.75 = \frac{N * \pi * Rs2}{x * y}$

$N= \frac{0.75 * x * y}{\pi * R_s^2}$ …………………………..……. (3.2)

Figure (3.2) shows a snapshot from the program that illustrates the values used in the previous example.

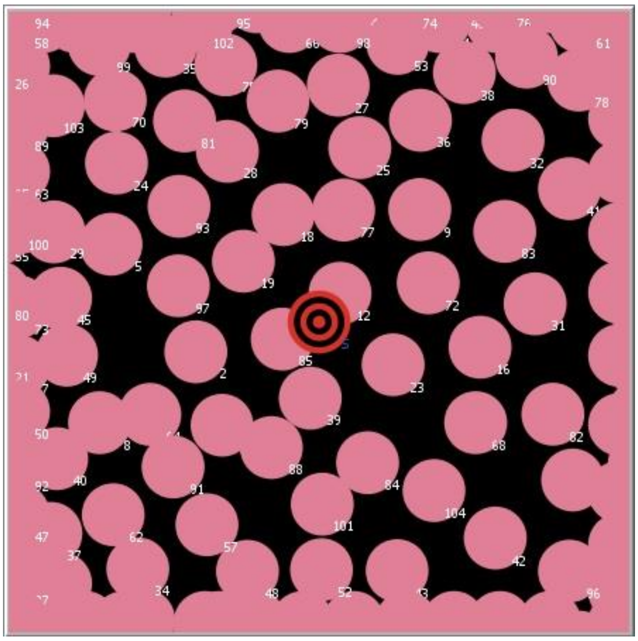


Figure (3.2) covering 75% of the area

Algorithm (3.1) presents the steps to estimate the number of the required sensors to achieve the required coverage level.

Algorithm (3.1) number of nodes estimation:

Input: coverage percentage, area size (Z), Sensing range (R_s)

Output: number of nodes.

Process:

```
1. Z ← area size
2. c ← coverage percentage
3. $R_s$ ← sensing distance
4. N is estimated using equation (3.1)
5. End
```

Algorithm (3.1) number of nodes estimation

E. Select the location of the base station

Certain node will be selected by the user to be the sink node (its color, shape and size will be changed). This node will be the base station which collects the information from all the other nodes. The location of the sink node can be anywhere in the selected area but in this thesis the location is at the center of the area. Figure (3.3) shows a snapshot of the setup process.

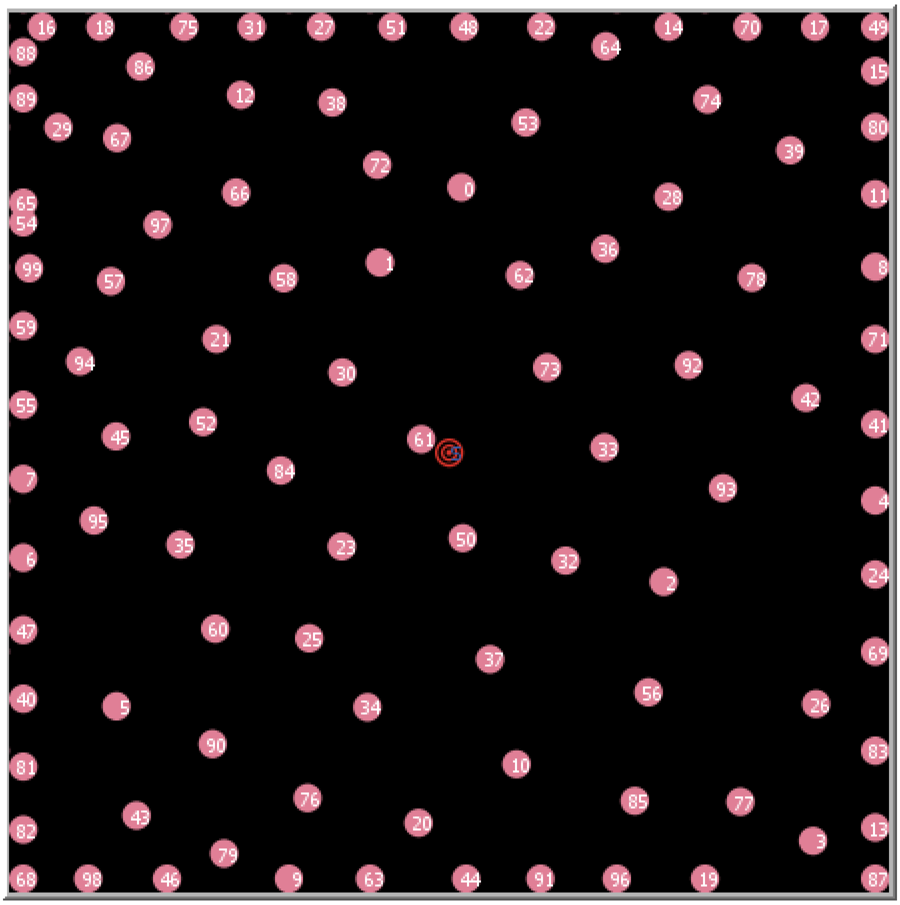


Figure (3.3): a program snapshot for a setup procedure.

3.3.2 Create a connected graph

After randomly deploying the sensor nodes, the coordinates (x, y) for each sensor location are collected. The distance between each two sensors can be calculated based on their ground (2D) locations using the Euclidean distance equation. The Euclidean distance gives the distance between two points or the straight line distance. Assuming that (x1, y1) and (x2, y2) are the coordinated of two points in a two-dimensional plane.

The distance (d) separated any two neighbor nodes can be estimated by the Euclidean distance formula.

A link between any two sensors nodes can be established if the distance between them is equal or less than the value of the selected communication range. Each node will be selected and the distance between it and its neighbors will be measured. The value of the distance between each two nodes will be compared with the value of the communication range to create links. After that a connected graph will be created. Algorithm (3.2) shows the connected graph.

Algorithm (3.2) connected graph:
Input: number of nodes, communication range (R_c), nodes position (x, y)
Output: distance among nodes, connected graph.
Process:

```
For i = 1 to number of nodes
    For j = i+1 to number of nodes
        $d_{ij} = \sqrt{[(xj - xi)^2 + (yj - yi)^2]}$
        If $d_{ij}$ <= R   then create link
            Connect node i to node j
        Else
    Next
Next
End
```

Algorithm (3.2) the connected graph

3.3.3 Estimate the coverage area

Any point within the deployed area is said to be covered if it lies within the coverage radius of at least one sensor. Furthermore, the area is said to be covered by the WSN if every point inside the area lies within the coverage radius of one sensor or more. In this step the coverage area will be estimated depending on the value of the sensing range and it displayed as the ratio of the covered area to the whole selected area.

Variable values for the sensing range are used. The coverage area for each node is calculated as a circle area (πr^2) with a diameter r (Hossain et al., 2012). Algorithm (3.3) presents the main steps to achieve the process of estimating the coverage area.

Algorithm (3.3) coverage estimation:
Input: number of nodes (n), sensing range (R_s), area of interest (Z).
Output: covered area percentage.
Process:

```
1. Set $R_s$ = variable
2. Set d = 0
3.      For i = 1 to n
4.          $d_i = \pi * r^2$
5.          $d = d + d_i$
6.      Next
7.      Coverage =d/Z
8. End
```

Algorithm (3.3) the coverage estimation

3.3.4 Calculate coverage percentage

For each monitored or tracked area, a full coverage (100%) is the main goal of each proposed WSN. Practically, it is difficult to achieve the full coverage percentage exactly. Different coverage levels ranges from (0 – 100) are possible to be achieved based on different parameters such as the area size and shape, number of deployed sensors and their coverage ranges. Any value greater or equal to 75% is proposed as an accepted level of coverage in this thesis.

3.3.5 Estimate the shortest path

The shortest path from any node can be found using the information that measures the distance between nodes. This path is used to send the messages from any selected node to the sink and it provides an efficient approach to transmit the message from the source to the destination with minimum energy and time. Dijkstra algorithm is utilized to create the shortest path from each node to the sink (see 2.8).

Algorithm (3.4) presents the developed steps to create the shortest path in NetLogo environment.

Algorithm (3.4) Pseudo code of shortest path:

Input: node-list.

Output: shortest path from n nodes to sink.

Process:

```
1. Check the paths from these random nodes to sink and choose the shortest by
Calculate the shortest link distance and it is considered less cost.
2. Change some properties of nodes
Begin
Set color ←   blue
Set size ←   2
End
3. Save (ID) and location of the sensor node
Begin
Set id1 ←   who
Set x1 ←   x1cor
Set y1 ←   y1cor
End
END.
```

Algorithm (3.4) the pseudo code of shortest path

One of the main problems in WSNs transmission is the common path or node busy case. This case happens when one or more nodes located at two or more shortest paths. These nodes are known as common nodes. The common nodes cannot receive and send the messages from / to two or more nodes in the same time. Lost messages will probably happen (the common node can pass only one message at a certain time).

This thesis developed an approach to indicate the common nodes by utilizing the sets intersections. The developed program estimates the shortest path and indicates all its nodes as a vector or a set. Four cases are suggested and tested in this thesis to observe and analyze the paths busy cases due to common nodes. These cases are when only one node send a message which represents a simple and usual case, two, three, and four nodes send messages at the same time which creating common nodes. To analyze and improve this case problem, the following mathematical model is proposed and implemented:

Model assumptions:

n: number of the selected sending nodes at the same time

S_i: The sending node id.

S_K: The sink node

i: The shortest path index (name).

n1: represent the minimum number of the shortest path.

N_i: set of path intermediate nodes id.

P_i: set of nodes on path i.

C: set of common nodes.

$P_i = (S_i, N1_i, N2_i, \ldots\ldots\ldots\ldots S_K)$ … … … … … … … … … (3.3)

$$[\mathrm{C}] = \bigcap_{i=1}^{n} P_i \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \quad (3.4)$$

$$= P1 \cap P2 \cap P3 \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \quad (3.5)$$

$IF\ [C] \neq \emptyset$. Then proposed alternative path (s) must be created.

The suggested solution for the problem of common path is to let one source node path as it is and select alternative path for the other source nodes to avoid any intersection. Figure (3.4) shows a snapshot from the program where the shortest paths of two nodes intersect in their way to the sink.

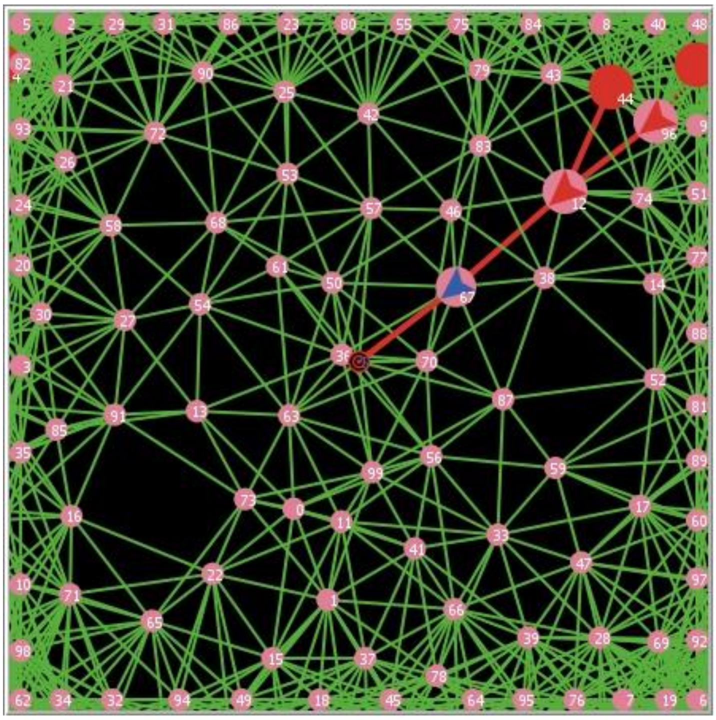


Figure (3.4) two paths intersection

Figure (3.5) shows the solution of this problem and choosing an alternative path for each intersected nodes.

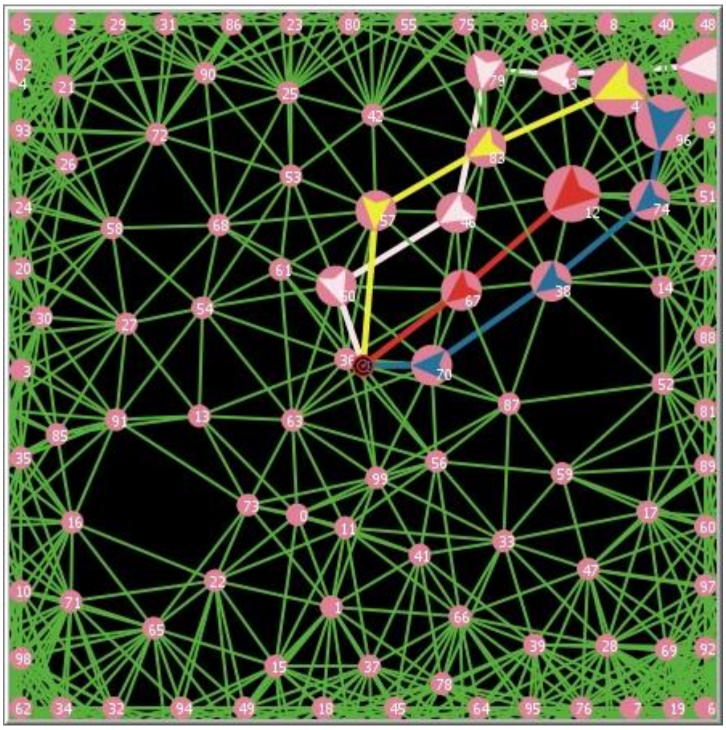


Figure (3.5) alternative path

3.3.6 Transmission process

Wireless Sensor Networks are a multi-hop wireless networks that are formed by self-organizing the sensor nodes of large-scale deployment. By WSNs, the data information of the perceived object in the monitoring area is collected and transmitted by a certain specific manner. Since WSNs has more advantages, they are widely used to collect information from the physical world and produce large-scale sensor data sets.

A number of nodes will be selected randomly to act as sending nodes (where n=1, 2, 3 and 4) at the same time. These nodes are selected in their implementation either randomly or predefined by the user to test the ability of sending information to the sink node and to ensure that the coverage information will reach the end user.

3.3.7 Building regression equations

Depending on the results of coverage by using multiple numbers for sensing range an equation is derived to calculate the coverage using any sensing range.

Simulation scenarios are performed to analyze, test and evaluate the relation between the coverage area and the other network parameters and metrics for different suggested cases. Analytic regression models are created for these different suggested cases.

These models are suitable to be used accurately without any need to repeat, execute or re-run the simulation programs. The suggested analytical equations can be tested to ensure that they will show accurate results during their training and testing examples.

The coverage area is depending on the sensing radius, the number of deployed sensors, the deployment approach (random or predefined), and the deployed area. The effect of these parameters on the coverage area is simulated and tested in different scenarios. Different regression curves and their equations are created and tested. These equations can be accurately used to estimate the coverage values. For more information about the correlation and the regression see (2.9) and (2.10).

CHAPTER FOUR

Simulation results and analysis

Chapter four

Simulation results and analysis

4.1 Overview

In this chapter, NetLogo simulator was used to apply the simulation process in order to evaluate the coverage of the nodes to the selected area using different parameters for (sensing range, communication range, number of nodes and the area that must be covered). The effect of these parameters was also studied.

4.2 Simulation setup

Different simulation scenarios are implemented to model, analyze and test the effects of different WSN parameters on the coverage area. Table (4.1) presents the suggested simulation setup.

Table (4.1): WSN environment

Parameter	Value
The simulator	NetLogo 6.1.1
Number of Nodes	Variable(80,100,130,150)
Deployment	Random approach
Sensing Range	Variable(0.8,0.9,…..,2)
Area	Variable(300*300, 400*400, 500*500, 600*600)

Different parameters affect the network coverage area such as the deployed area, number of nodes, deployment approach (random or predefined) and sensing ranges. Figure (4.1) represent a simulation program environment sample.

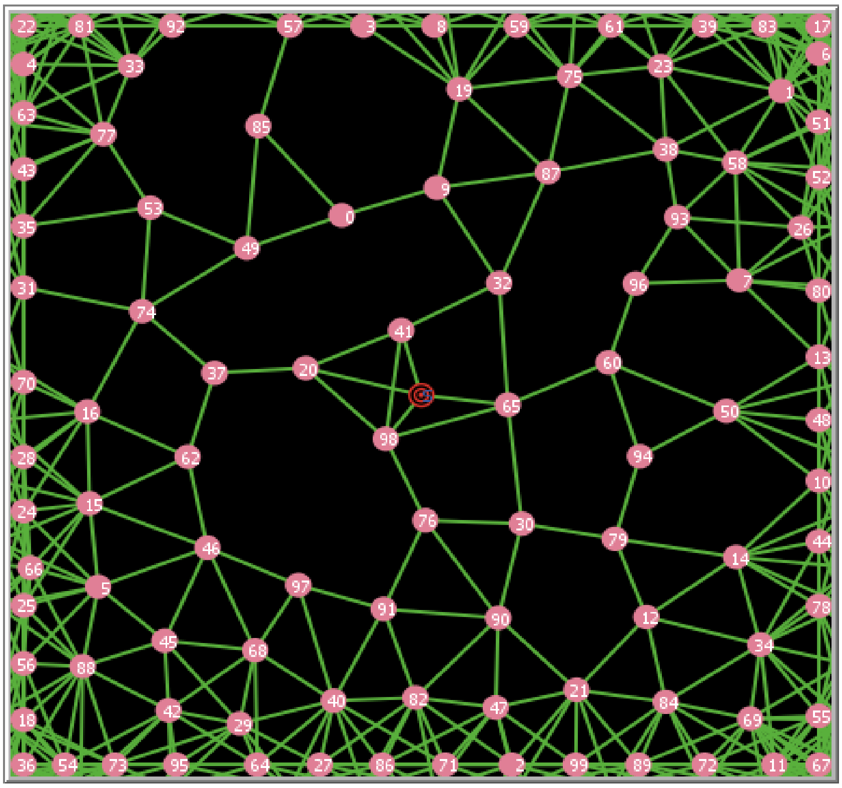


Figure (4.1) Network environment sample

4.3 Simulation results

Four cases are proposed in implementing the simulation experiments to model the coverage area of the WSNs.

4.3.1 Case 1 results

Different scenarios are implemented to simulate the effect of each parameter on the coverage area. In this case the deployed area size is selected as 300*300 m. Number of deployed nodes is varied (80,100,130 and 150) with different sensing ranges values (0.8, 0.9, 1, 1.2, 1.4, 1.6, 1.8 and 2). The following four scenarios represent the simulation scenarios to show the effects of varying the number of nodes with different sensing ranges in the same area.

A. scenario 1: The first simulation scenario comprises of 80 sensor nodes deployed randomly in a 300*300 m area with a sink node in a center point. The sensing range for each sensor node is varied from (0.8, 0.9, 1, 1.2, 1.4, 1.6, 1.8 and 2). Figure (4.2) shows a snapshot from the program for this scenario contain the shortest paths from a random node to the sink.

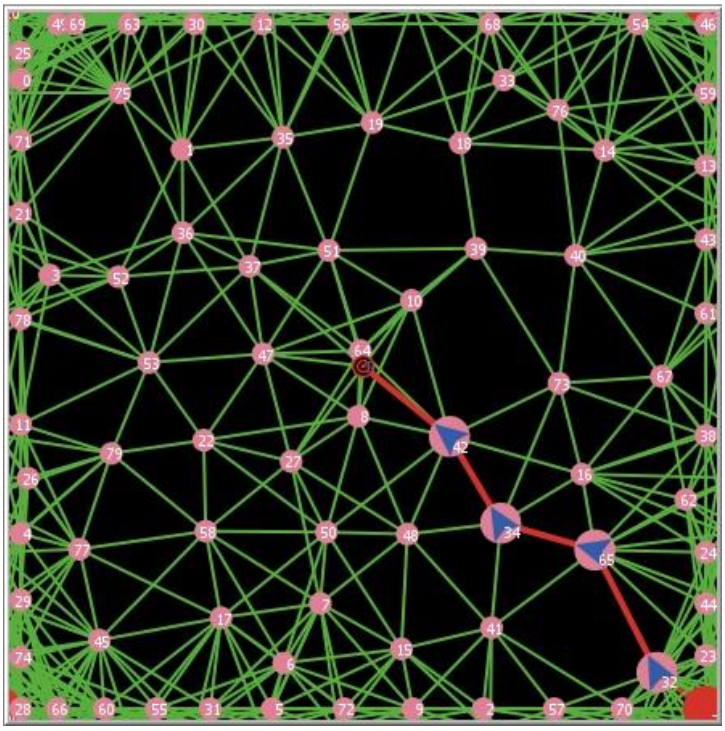


Figure (4.2): Snapshot of the program in case 1, scenario 1

Table (4.2) specifies the resulted coverage area in each case. The coverage area is calculated according to the algorithm in figure (3.6) and equations 3.1. Shortest paths are also estimated from each sensor node to the sink node based on algorithm in figure (3.7). A shortest path sample with its length is presented in table (4.2).

Table (4.2): Simulation results for case 1, scenario 1

Area	No. of Nodes	Sensing range	Coverage	Shortest path sample	Shortest path length
300*300	80	0.8	36%	[(node 21) (node 109) (sink 129)]	14.718
		0.9	46%	[(node 30) (node 96) (sink 129)]	19.335
		1	57%	[(node 47) (node 12) (sink 129)]	24.933
		1.2	83%	[(node 10) (node 67) (node 14) (sink 129)]	23.289
		1.4	100%	[(node 82) (sink 129)]	12.435
		1.6	100%	[(node 85) (sink 129)]	13.978
		1.8	100%	[(node 121) (node 35) (node 17) (sink 129)]	36.766
		2	100%	[(node 77) (node 104) (node 52) (node 17) (sink 129)]	31.624

Table (4.2) represents certain estimated values for the coverage area based on certain sensing ranges. To generalize this case, a regression model is developed based on these measured values to estimate the coverage area for each sensing range value. Figure (4.3) represents the regression equation developments. Microsoft excel is used to fit the measured values with the expected values curves.

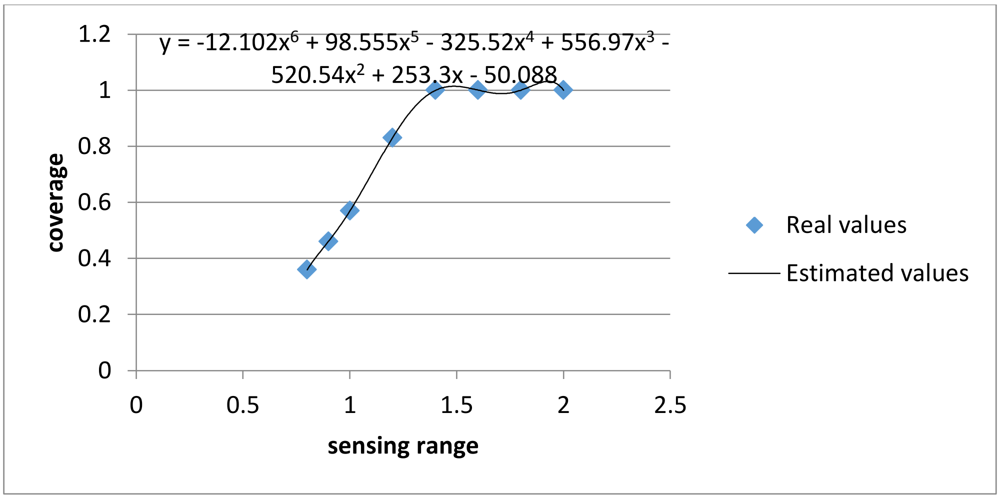


Figure (4.3): sensing range and coverage area for case1, scenario1

This regression equation is tested with some of the tabular values to show its validity and then can be used to estimate the coverage area based on any sensing range value. The variable x in equation (4.1) represents the suggested sensing range and the variable y represents the resulted coverage area.

$y = -12.102x^6 + 98.555x^5 - 325.52x^4 + 556.97x^3 - 520.54x^2 + 253.3x - 50.088$... (4.1)

To test the validity of equation (4.1), different values from table (4.2) are used as inputs to calculate the required coverage values.

Example:

- When the sensing range = 0.8 then

  $y = -12.102(0.8)^6 + 98.555(0.8)^5 - 325.52(0.8)^4 + 556.97(0.8)^3 - 520.54(0.8)^2 + 253.3(0.8) - 50.088 \quad = 0.36$

- When the sensing range = 1 then y = 0. 57

These y results ensure the validity and the accurate of this created equation. So, this equation can be used to estimate a coverage area for any sensing range value without running the simulation program. In the following, two random values for the sensing range are suggested to calculate their related coverage area:

- When the sensing range = 0.95 then y = 0.517
- When the sensing range = 1.5 then y = 1

Another proposed approach in this thesis is how to estimate the required number of sensors to cover certain area based on the available sensors specifications (their coverage range). This proposal represents a practical issue and useful in designing any new efficient WSN. Usually, the coverage radius of each sensor is installed by its production factory. This approach is modeled and simulated to add a proposed foundation in designing and creating reliable WSNs applications.

Table (4.3) shows a suggested example to implement this proposal approach. In this example, the sensor's sensing range is suggested to be = 0.8, and the deployed area = 300*300 m.

Different simulation runs are performed to estimate the required number of sensors to achieve certain coverage level. In this experiment the simulation program is implemented and a coverage level is calculated for each case. Usually the coverage level increased with increasing the number of sensors. This approach can assist the simulation users or the network managers and engineers to select the suitable wanted coverage level.

Table (4.3) the coverage values for case 1, scenario 1

Area	Sensing range	No. of Nodes	Coverage
300*300	0.8	80	36%
		90	41%
		100	46%
		120	55%
		140	64%
		160	73%
		180	82%
		200	91%

Table (4.3) represents the estimated values for the coverage area based on variable number of nodes. Regression model have been developed to estimate the coverage for each used number of nodes based on these measured values. Figure (4.4) represents the developed regression equation.

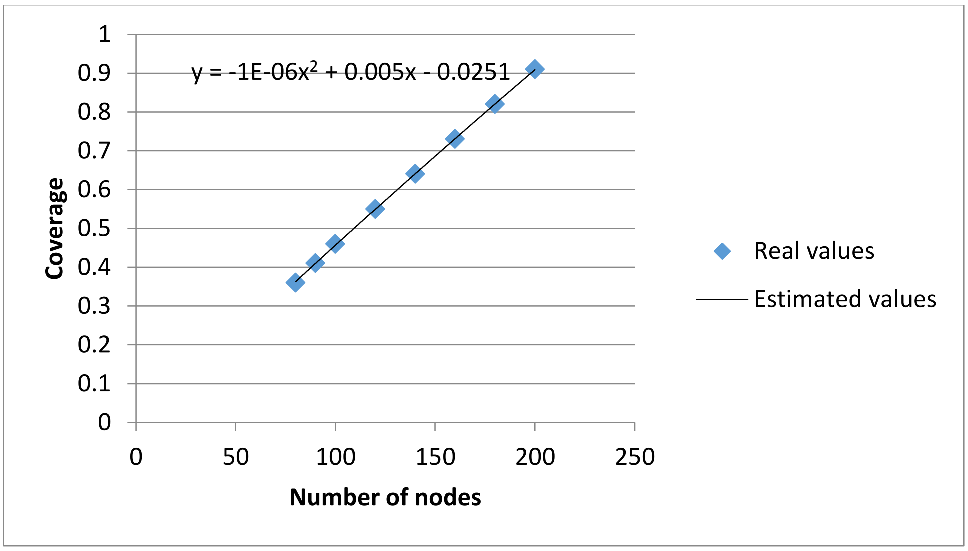


Figure (4.4) number of nodes and coverage area for case1, scenario1

The variable x in equation (4.2) represents the used number of nodes and the variable y represents the resulted coverage area. This equation is tested and can be used to estimate the coverage area based on any number of nodes value.

$y = -1E\text{-}06x^2 + 0.005x - 0.0251$ ……... (4.2)

Different values from table (4.3) are used as inputs to in order to check the validity of equation (4.2). Example:

- When the number of nodes = 90 then y = 0.41
- When the sensing range = 100 then y = 0. 46

These y values confirm that the equation that was generated is valid and accurate. Therefore, without running the simulation program, this equation can be utilized to predict the coverage area for any number of nodes used. Two random values for the number of nodes are suggested below:

- When the number of nodes = 110 then y = 0.51
- When the number of nodes = 150 then y = 0.70

As a network representation, certain important performance metrics are measured and listed in figure (4.5). These metrics are A) timing sequence, B) sensor transmitted data, C) sensor received data and D) sampled received data while E) represents the sent from four random nodes to the sink node.

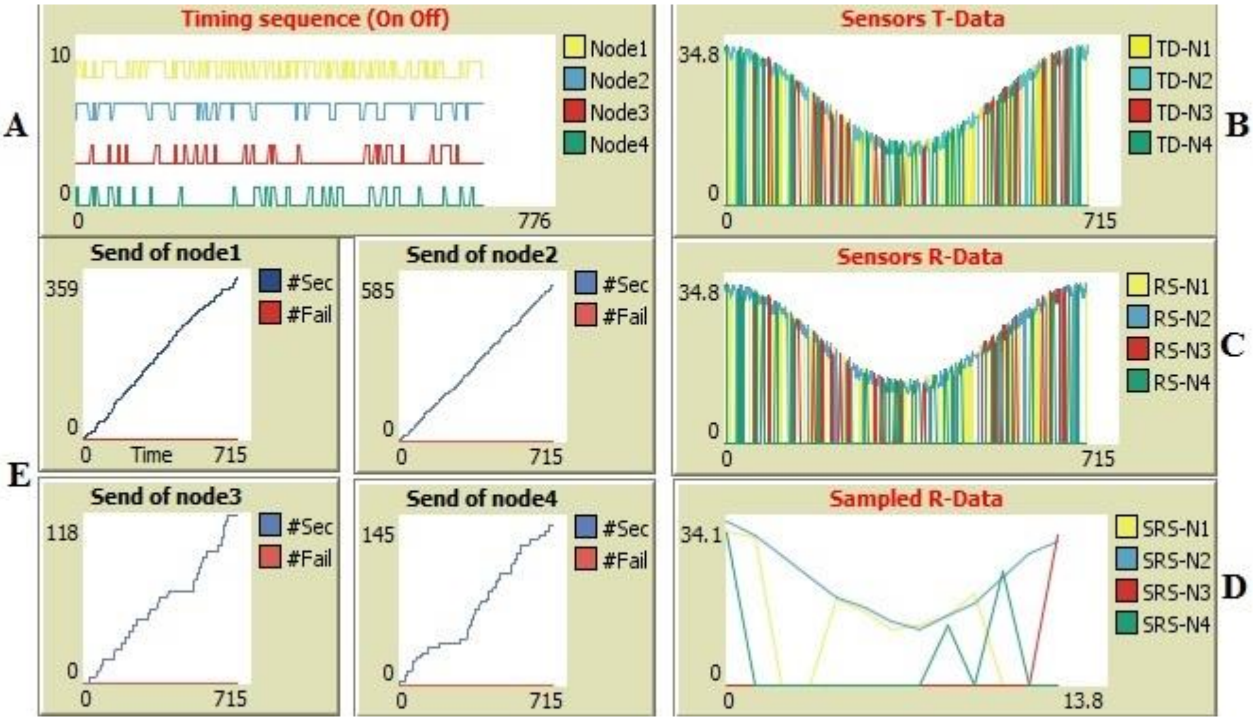


Figure (4.5): performance metrics for case 1, scenario 1.

From figure (4.5), the time sequence represents a sample from the sending nodes states (node 1, node 2, node 3, node 4) with time in part A, the sensor transmission data in part B represents their transmitted data with time, the sensor received data represents their received data in part C, part D of figure (4.5) represents a sample from the received data by node 1, node 2, node 3 and node 4, Part E shows the sending state for each sending node with time.

B. scenario 2: The second simulation scenario consists of 100 sensor nodes deployed randomly in a 300*300 m area with a sink node in a center point. The sensing range is as indicated previously. Figure (4.6) shows a snapshot from the program for this scenario contain the shortest path from n random nodes to the sink at the same time.

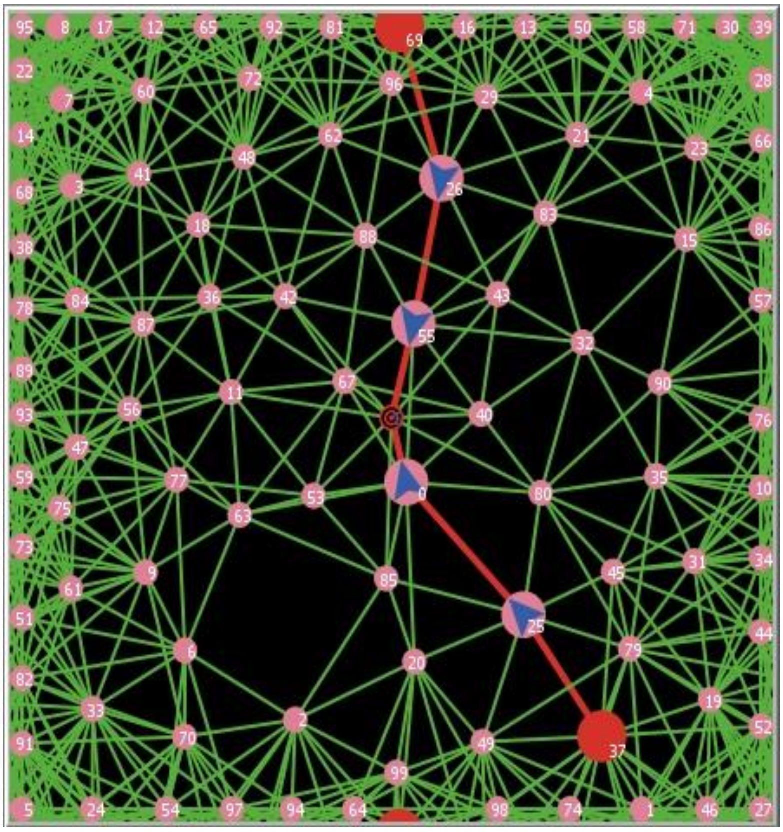


Figure (4.6): Snapshot of the program in case 1, scenario 2

Table (4.4) specifies the estimated coverage while using different sensing ranges. The coverage area is calculated according to algorithm in figure (3.6) and equation 3.1.

Table (4.4): Simulation results for case 1, scenario 2

Area	Number of Nodes	Sensing range	Coverage	Shortest path sample	Shortest path length
300*300	100	0.8	46%	[(node 93) (sink 100)]	6.201
		0.9	58%	[(node 21) (node 95) (sink 100)]	15.331
		1	71%	[(node 61) (sink 100)]	3.590
		1.2	100%	[(node 57) (node 9) (node 55) (sink 100)]	19.986
		1.4	100%	[(node 60) (node 61) (sink 100)]	11.767
		1.6	100%	[(node 64) (node 81) (sink 100)]	11.590
		1.8	100%	[(node 11) (sink 100)]	3.67
		2	100%	[(node 18) (node 9) (node 55) (sink 100)]	18.872

Table (4.4) represents the simulation results to estimate the values for the coverage area for each sensing range. Figure (4.7) presents the collected results from table (4.4) and the fitted curve with a created equation for coverage calculations.

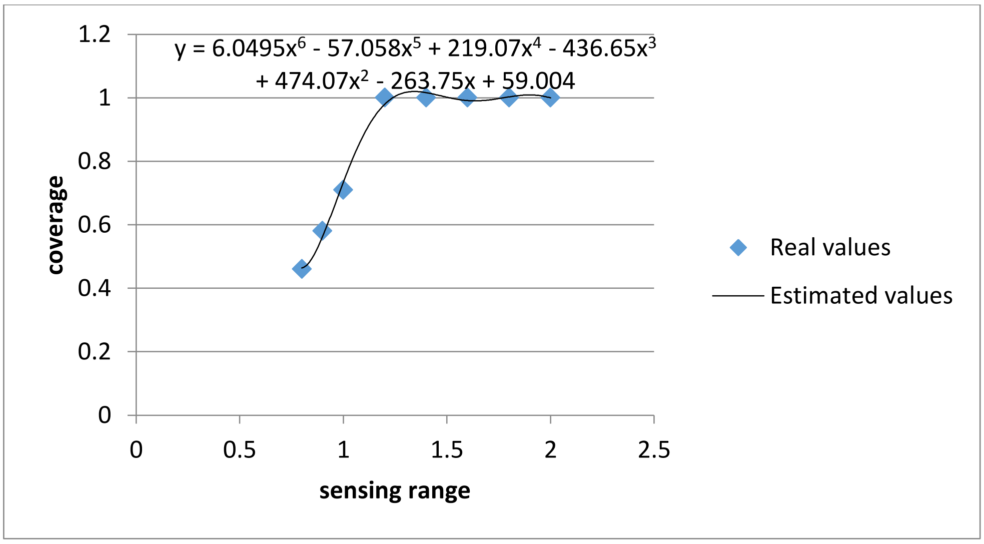


Figure (4.7): sensing range and coverage area for case1, scenario 2

Equation (4.3) is created after fitting the expected values with the calculated values in figure (4.4).

$y = 6.0495x^6 - 57.058x^5 + 219.07x^4 - 436.65x^3 + 474.07x^2 - 263.75x + 59.004$…... (4.3)

The variable x in equation (4.3) represents the suggested sensing range and the variable y represents the resulted coverage area.

To test the validity of equation (4.3), different values from table (4.4) are used as inputs to calculate the required coverage values.

Example:

- When the sensing range = 0.8 then y =0.46
- When the sensing range = 1.4 then y = 1

These y results ensure the validity and the accuracy of this created equation. So, this equation can be used to estimate a coverage area for any other sensing range value without running the simulation program. Following are examples when selecting two random values for the sensing range not considered in table (4.4) to calculate their related coverage area:

- When the sensing range = 0.95 then y = 0.64
- When the sensing range = 1.5 then y = 1

There are certain important performances metrics are measured and listed as a network representation, in figure (4.8). Five metrics are considered and observed by implementing the simulation program during the simulation run time. These metrics are A) timing sequence, B) sensor transmitted data, C) sensor received data and D) sampled received data and E) shows the sent from four nodes chosen randomly to the sink node.

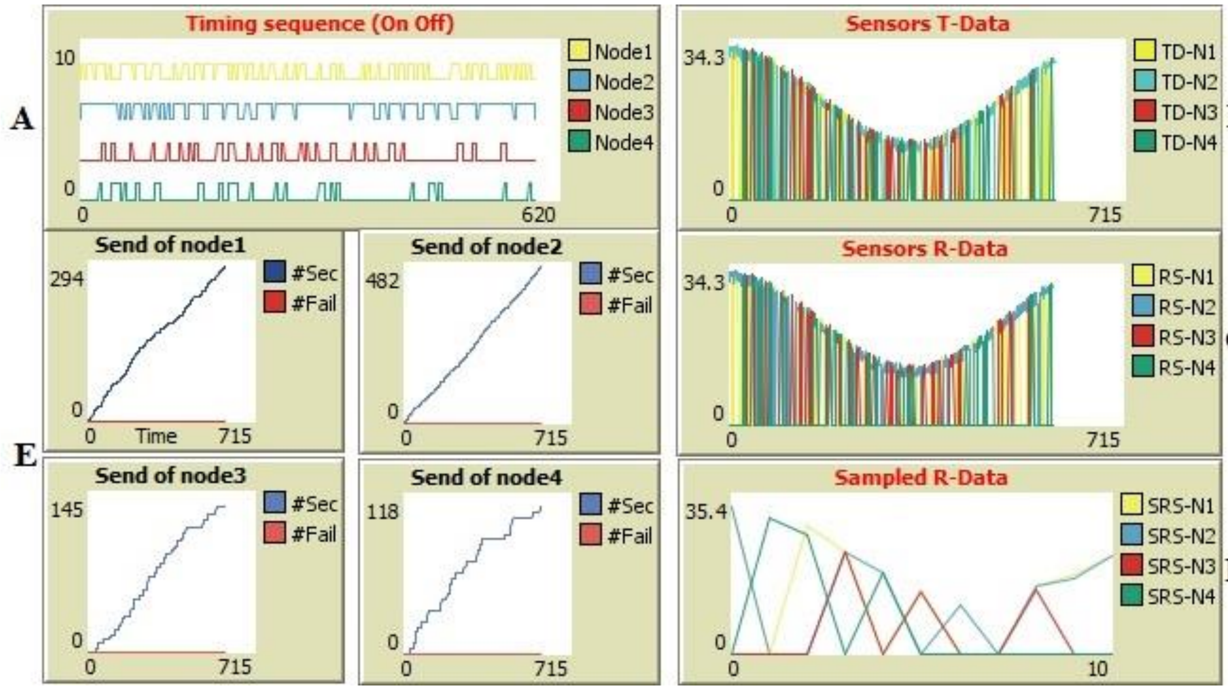
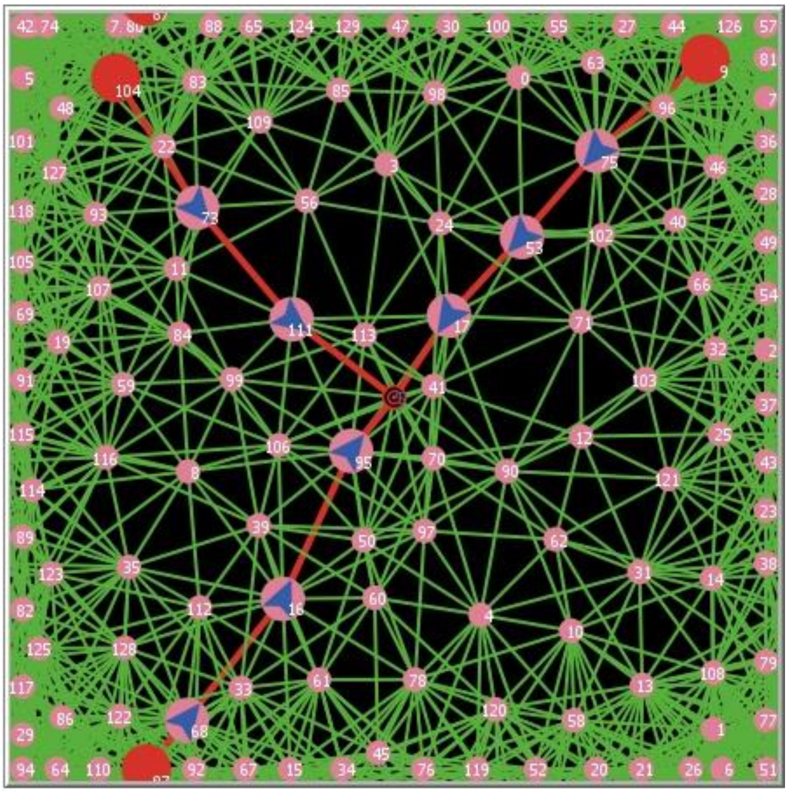


Figure (4.8): performance metrics for case 1, scenario 2.

From figure (4.8), the time sequence represents a sample from the sending nodes states (node 1, node 2, node 3, node 4) with time in part A, the sensor transmission data in part B represents their transmitted data with time, the sensor received data represents their received data at the sink in part C, part D of figure (4.8) represents a sample from the received data by node 1, node 2, node 3 and node 4, Part E shows the sending state for each sending node with time. Comparing figure (4.8) with figure (4.5) shows a similarity in all sub figures behaviours except part D due to the increase in number of the deployed sensors.

C. scenario 3: In the third simulation scenario, 130 nodes where deployed randomly in a same area with a sink node in a center point. The sensing range is as indicated previously. Figure (4.9) shows a snapshot from the program for this scenario contain the shortest path from four random nodes to the sink.



Figure(4.9): Snapshot of the program in case 1, scenario 3

Table (4.5) shows the calculated coverage area by implementing the simulation program for multiple sensing ranges.

Table (4.5): Simulation results for case 1, scenario 3

Area	Number of Nodes	Sensing range	Coverage	Shortest path sample	Shortest path length
300*300	130	0.8	59%	[(node 51) (node 103) (sink 130)]	12.905
		0.9	75%	[(node 107) (node 57) (sink 130)]	15.485
		1	93%	[(node 64) (node 98) (sink 130)]	12.972
		1.2	100%	[(node 41) (node 7) (sink 130)]	15.515
		1.4	100%	[(node 88) (node 79) (sink 130)]	14.045
		1.6	100%	[(node 57) (sink 130)]	9.39
		1.8	100%	[(node 98) (sink 130)]	5.348
		2	100%	[(node 9) (node 73) (node 27) (sink 130)]	16.374

Figure (4.10) presents the shape of the results in table (4.5) with their expected (Estimated) values. Equation (4.4) is created based on the tabular results for the coverage area.

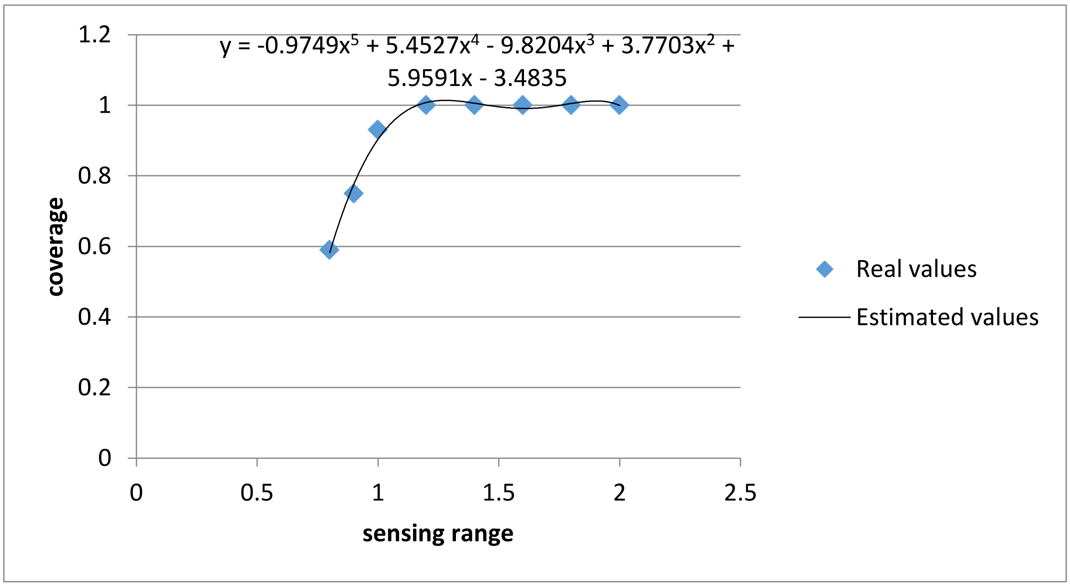


Figure (4.10): sensing range and coverage area for case1, scenario 3

The variable x in equation (4.4) stands for the sensing range and the variable y stands for the coverage area.

$y = -0.9749x^5 + 5.4527x^4 - 9.8204x^3 + 3.7703x^2 + 5.9591x - 3.4835$……..(4.4)

Example:

- When the sensing range = 1.4 then y =1
- When the sensing range = 1.2 then y = 1
- When the sensing range = 0.95 then y = 0.84
- When the sensing range = 1.3 then y = 1

Figure (4.11) shows the simulation results for the five performance metrics.

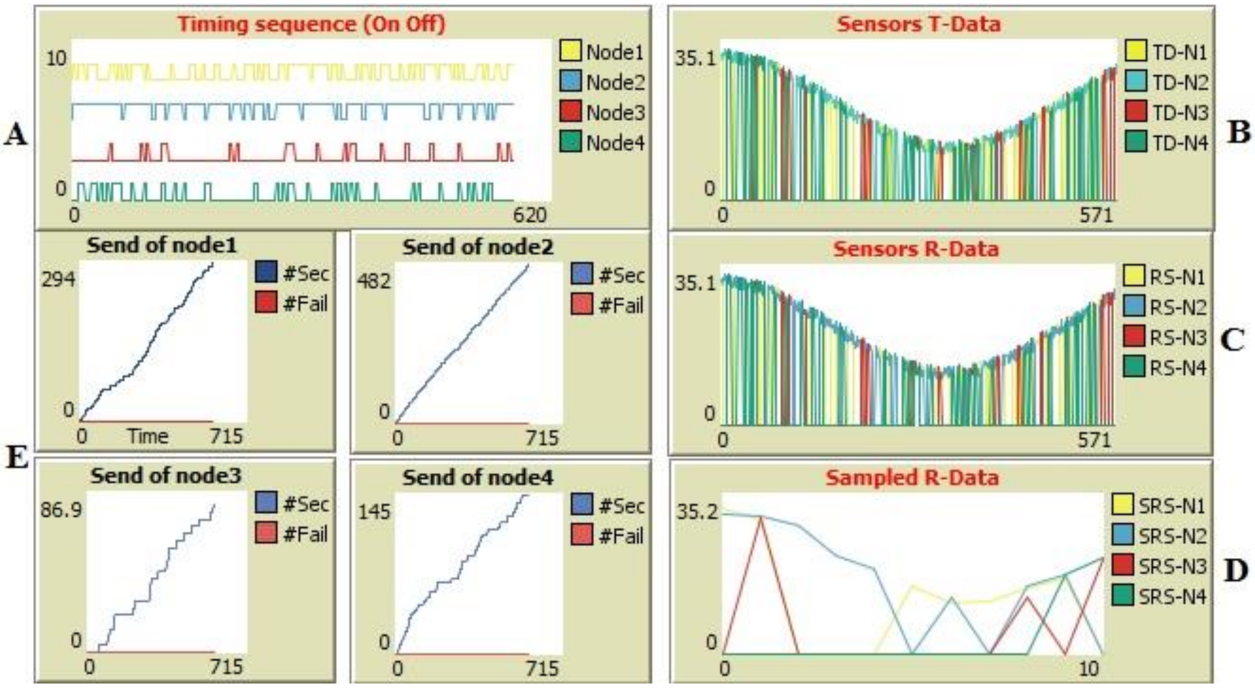


Figure (4.11): performance metrics for case 1, scenario 3.

The behavior of the results in figure (4.11) shows stability and clearly presents the effect of number of nodes.

D. scenario 4: 150 nodes distributed randomly in a 300*300 m area while the sink node is located at a center point. The sensing range is as indicated previously. Figure (4.12) shows a snapshot from the program for this scenario contain the shortest path from four random nodes to the sink.

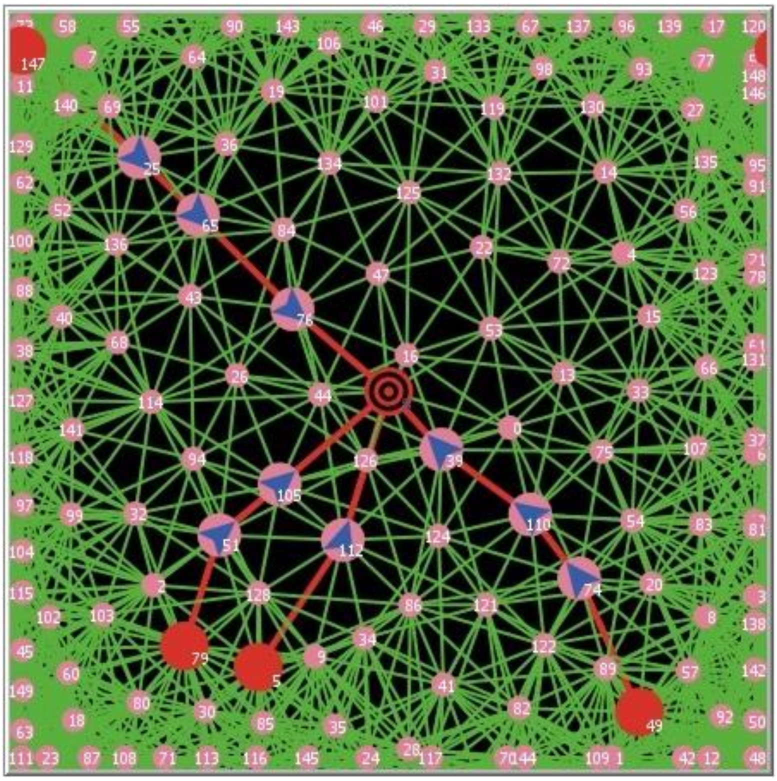


Figure (4.12): Snapshot of the program in case 1, scenario 4

Table (4.6) shows the calculated coverage area by implementing the simulation program for multiple sensing ranges.

Table (4.6): Simulation results for case1, scenario 4

Area	Number of Nodes	Sensing range	Coverage	Shortest path sample	Shortest path length
300*300	150	0.8	68%	[(node 40) (sink 150)]	9.525
		0.9	87%	[(node 66) (node 123) (node 137) (sink 150)]	15.178
		1	100%	[(node 116) (node 52) (node 43) (sink 150)]	14.413
		1.2	100%	[(node 22) (sink 150)]	9.838
		1.4	100%	[(node 10) (node 24) (sink 150)]	15.247
		1.6	100%	[(node 30) (sink 150)]	4.025
		1.8	100%	[(node 61) (node 19) (sink 150)]	15.742
		2	100%	[(node 6) (node 81) (sink 150)]	11.835

Figure (4.13) presents the shape of the results in table (4.6) with their expected (Estimated) values. Equation (4.5) is created based on the tabular results for the coverage area.

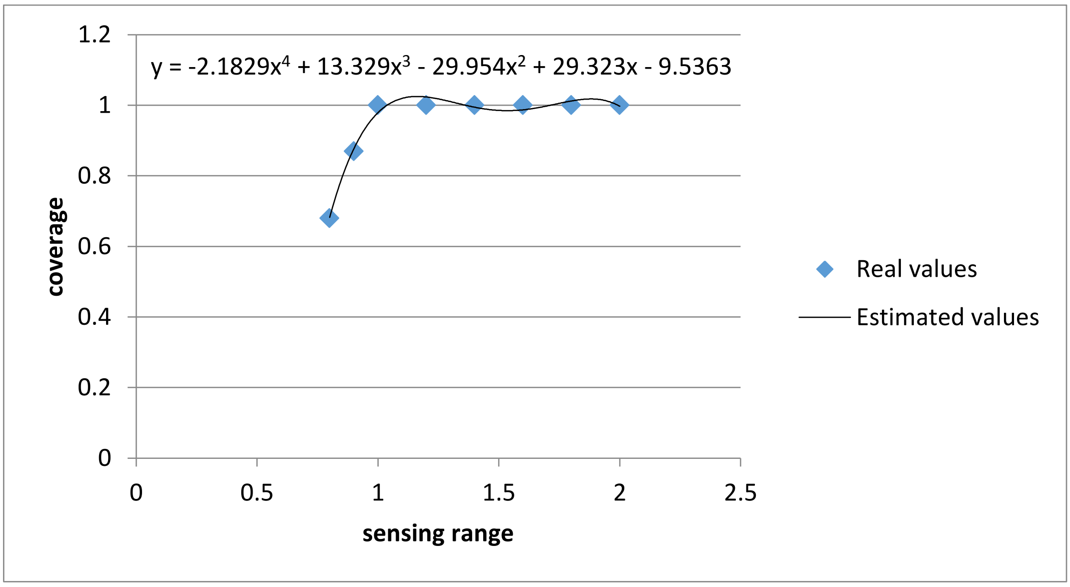


Figure (4.13): sensing range and coverage area for case1, scenario 4

The variable x in equation (4.5) used to represent the sensing range and the variable y used to represent the coverage area.

$$y = -2.1829x^4 + 13.329x^3 - 29.954x^2 + 29.323x - 9.5363 \ldots\ldots\ldots (4.5)$$

Example:

- When the sensing range = 0.9 then y =0.87
- When the sensing range = 1.2 then y = 1
- When the sensing range = 0.95 then y = 0.93
- When the sensing range = 1.5 then y = 1

The same five performance metrics are listed and measured in figure (4.14).

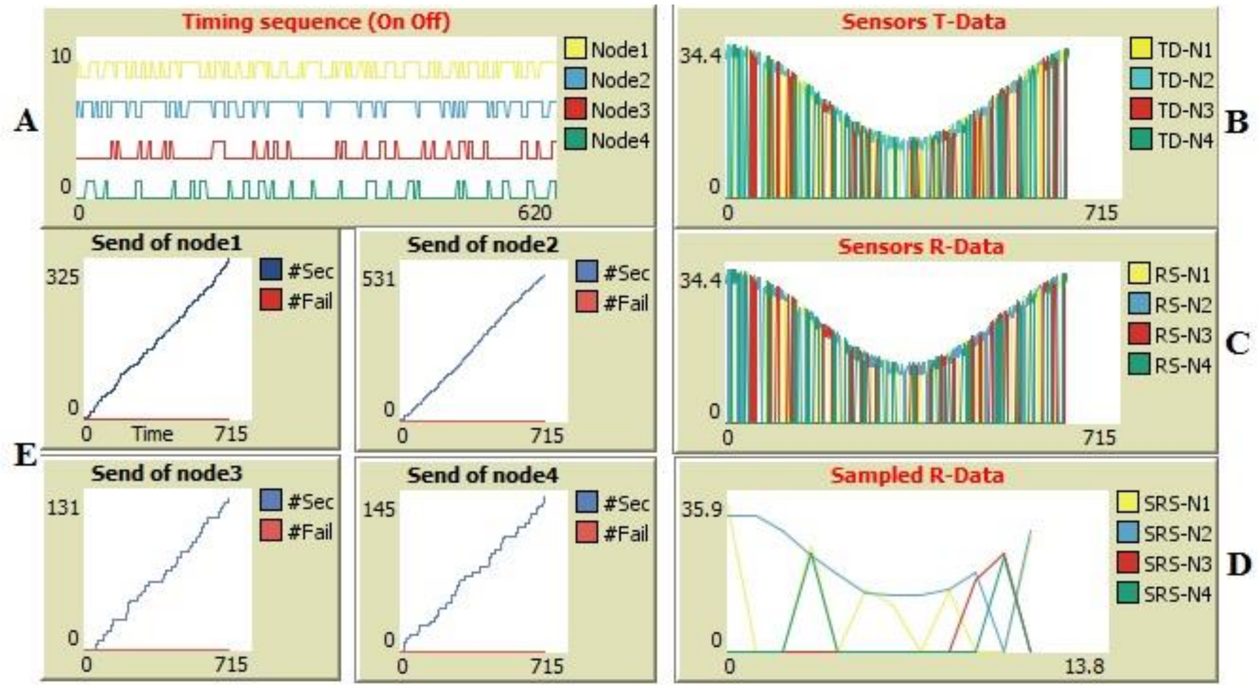


Figure (4.14): performance metrics for case 1, scenario 4.

The behavior of the results in figures (4.5, 4.8, 4.11 and 4.14) shows stability and clearly presents the effect of number of nodes.

4.3.2 Case 2 results

As presented previously different scenarios are implemented to simulate the effect of different parameter on the coverage area. The deployed area is selected as 400*400 m in this case. The number of deployed nodes is variable. The scenarios were used to show the effects of variable number of nodes with different sensing ranges in a same area.

A. scenario 1: The first simulation scenario comprises of 80 nodes deployed randomly in a 400*400 m while the sink node in the center. The sensing range for each sensor node is varied as the scenarios in case 1. Table (4.7) specifies the resulted coverage area for each sensing range. Shortest paths are also calculated from each sensor node to the sink node. A shortest path sample with its length is presented in table (4.7).

Table (4.7): Simulation results for case 2, scenario 1

Area	Number of Nodes	Sensing range	Coverage	Shortest path sample	Shortest path length
400*400	80	0.8	26%	[(node 32) (node 20) (node 29) (sink 80)]	20.614
		0.9	32%	[(node 38) (node 16) (node 26) (sink 80)]	20.698
		1	40%	[(node 72) (node 38) (node 16) (node 26) (sink 80)]	26.551
		1.2	58%	[(node 30) (sink 80)]	3.557
		1.4	79%	[(node 5) (node 4) (sink 80)]	12.044
		1.6	100%	[(node 20) (node 29) (sink 80)]	16.233
		1.8	100%	[(node 7) (node 43) (sink 80)]	17.384
		2	100%	[(node 31) (sink 80)]	8.784

Figure (4.15) represents the regression equation which can be used to calculate the coverage area. Equation (4.6) is created based on the tabular results for the coverage area.

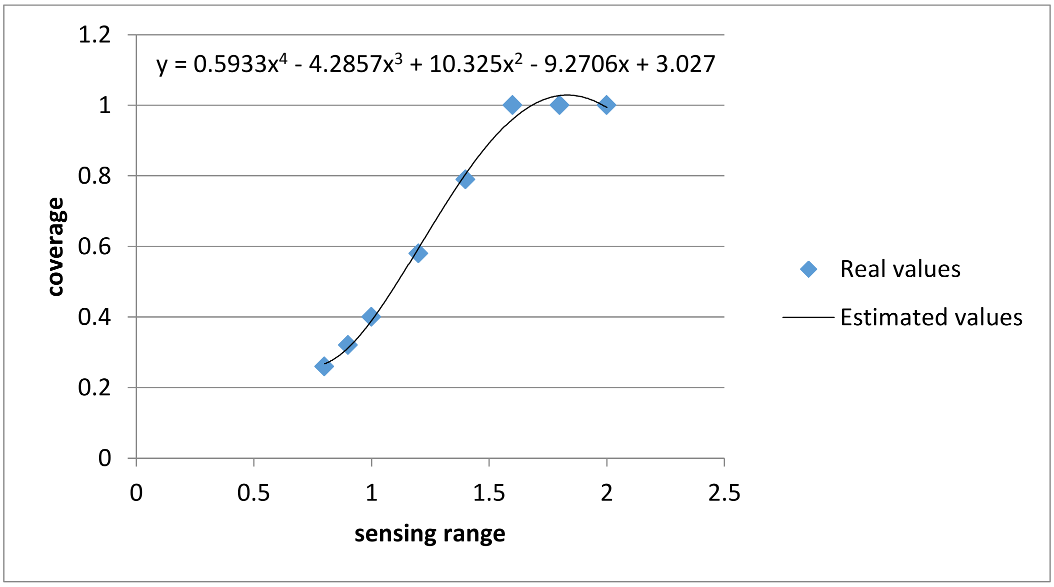


Figure (4.15): sensing range and coverage area for case2, scenario 1

The variable x in equation (4.6) represents the suggested sensing range and the variable y represents the resulted coverage area.

$y = 0.5933x^4 - 4.2857x^3 + 10.325x^2 - 9.2706x + 3.027$ …… (4.6)

Example

- When the sensing range = 0.8 then y =0.26
- When the sensing range = 0.95 then y = 0.34
- When the sensing range = 1.5 then y = 0.89

Figure (4.16) represents the simulation results for multiple important performance metrics measured and listed in as a network representation.

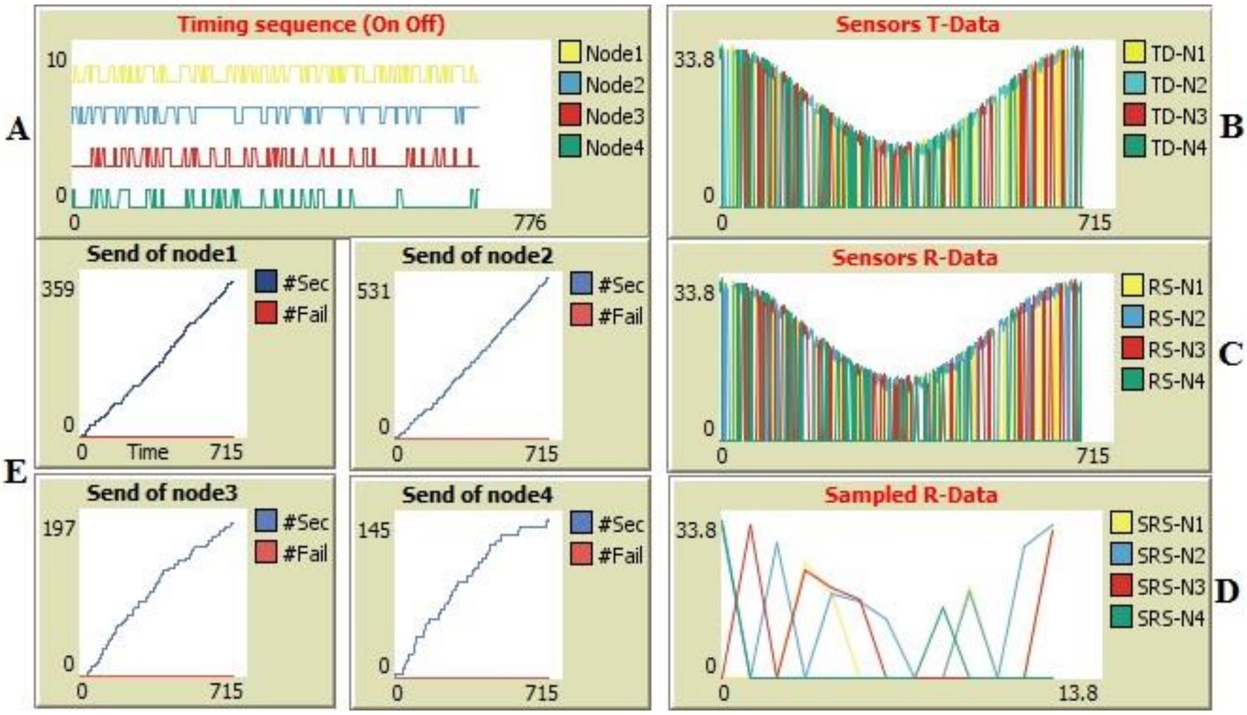


Figure (4.16): performance metrics for case 2, scenario 1.

The behavior of the results in figure (4.16) shows certain stability.

B. scenario 2: In the second simulation scenario 100 nodes are deployed randomly in the same area. The sensing range is as indicated previously. Table (4.8) specifies the coverage depending on using different sensing ranges.

Table (4.8): Simulation results for case 2, scenario 2

Area	Number of Nodes	Sensing range	Coverage	Shortest path sample	Shortest path length
400*400	100	0.8	32%	[(node 9) (node 61) (node 85) (sink 100)]	20.760
		0.9	41%	[(node 51) (node 19) (sink 100)]	13.5
		1	50%	[(node 61) (node 85) (sink 100)]	11.059
		1.2	73%	[(node 8) (node 28) (node 85) (sink 100)]	20.576
		1.4	99%	[(node 0) (node 60) (node 13) (sink 100)]	18.722
		1.6	100%	[(node 76) (sink 100)]	9.433
		1.8	100%	[(node 29) (node 62) (sink 100)]	15.885
		2	100%	[(node 20) (sink 100)]	6.509

Figure (4.17) shows the shape of the results in table (4.8) with their expected (Estimated) values. Equation (4.7) is created based on the tabular results for the coverage area.

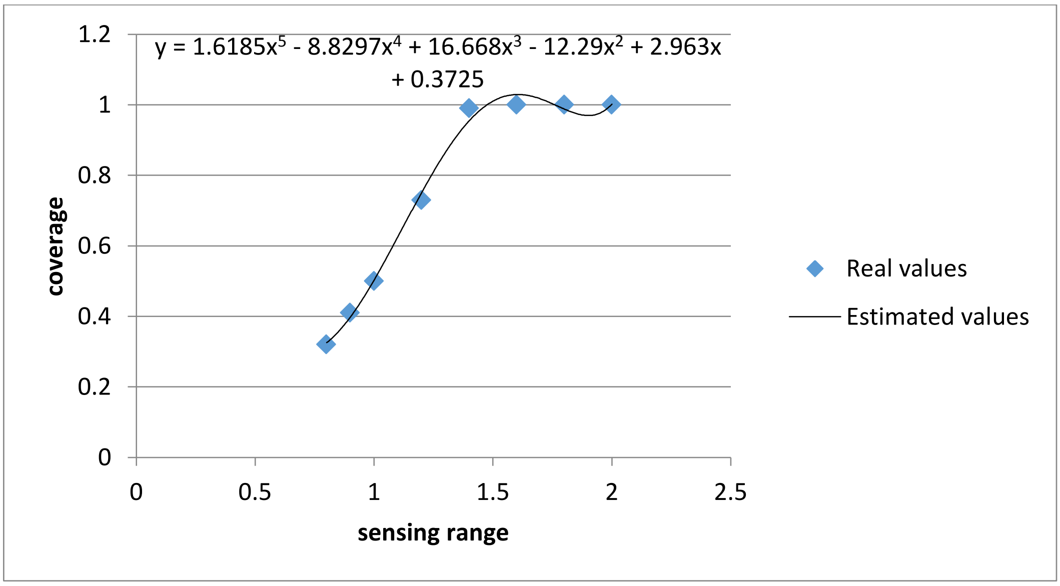


Figure (4.17): sensing range and coverage area for case2, scenario 2

The variable x in equation (4.7) represents the suggested sensing range and the variable y represents the resulted coverage area.

$$y = 1.6185x^5 - 8.8297x^4 + 16.668x^3 - 12.29x^2 + 2.963x + 0.3725 \ldots (4.7)$$

Example

- When the sensing range = 0.8 then y = 0.32
- When the sensing range = 1 then y = 0. 50
- When the sensing range = 0.95 then y = 0.44
- When the sensing range = 1.5 then y = 1

In figure (4.18) the simulation results for the five performance metrics are presented.

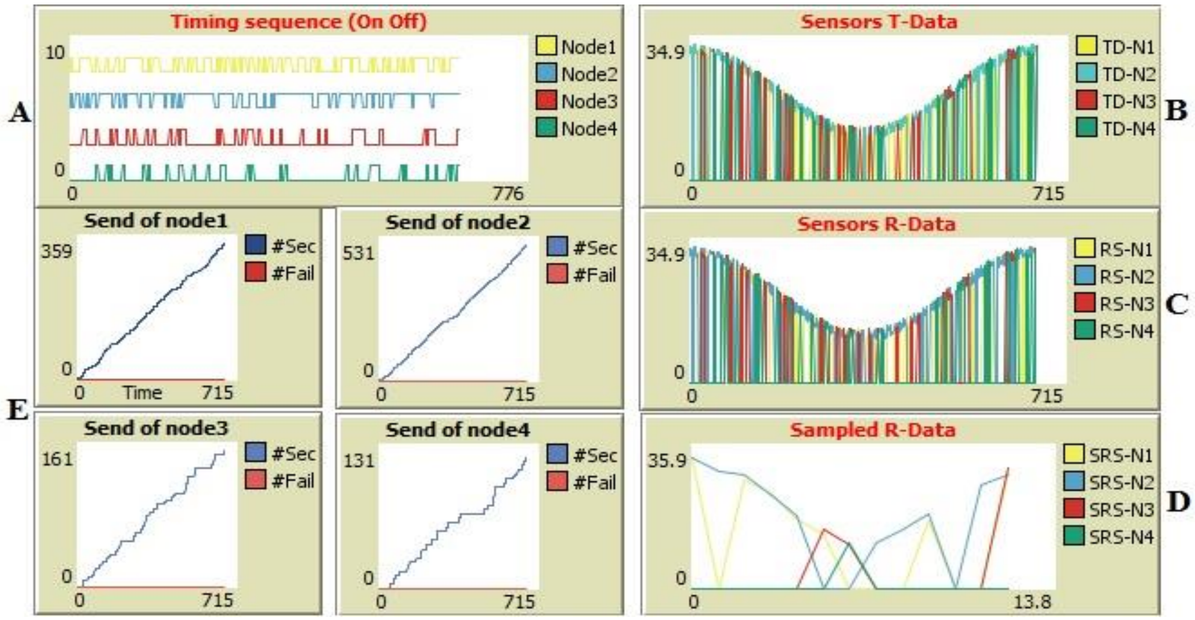


Figure (4.18): performance metrics for case 2, scenario 2.

The behavior of the results in figure (4.18) shows stability and clearly presents the effect of number of nodes if it compared with figure (4.16).

C. scenario 3: In this scenario 130 nodes where in an 400*400 m area these nodes where deployed randomly while the sink node in a center point. The sensing range is as indicated previously. Table (4.9) demonstrates the coverage while multiple sensing ranges are used.

Table (4.9): Simulation results for case 2, scenario 3

Area	Number of Nodes	Sensing range	Coverage	Shortest path sample	Shortest path length
400*400	130	0.8	42%	[(node 54) (node 32) (node 62) (sink 130)]	22.211
		0.9	53%	[(node 10) (node 83) (sink 130)]	16.532
		1	65%	[(node 9) (node 79) (sink 130)]	14.332
		1.2	94%	[(node 83) (sink 130)]	7.647
		1.4	100%	[(node 90) (node 54) (node 32) (node 62) (sink 130)]	27.247
		1.6	100%	[(node 107) (node 83) (sink 130)]	13.173
		1.8	100%	[(node 113) (node 79) (sink 130)]	18.321
		2	100%	[(node 57) (sink 130)]	2.112

Figure (4.19) illustrate the shape of the results in table (4.9) with their expected (Estimated) values. Equation (4.8) is created based on the tabular results for the coverage area.

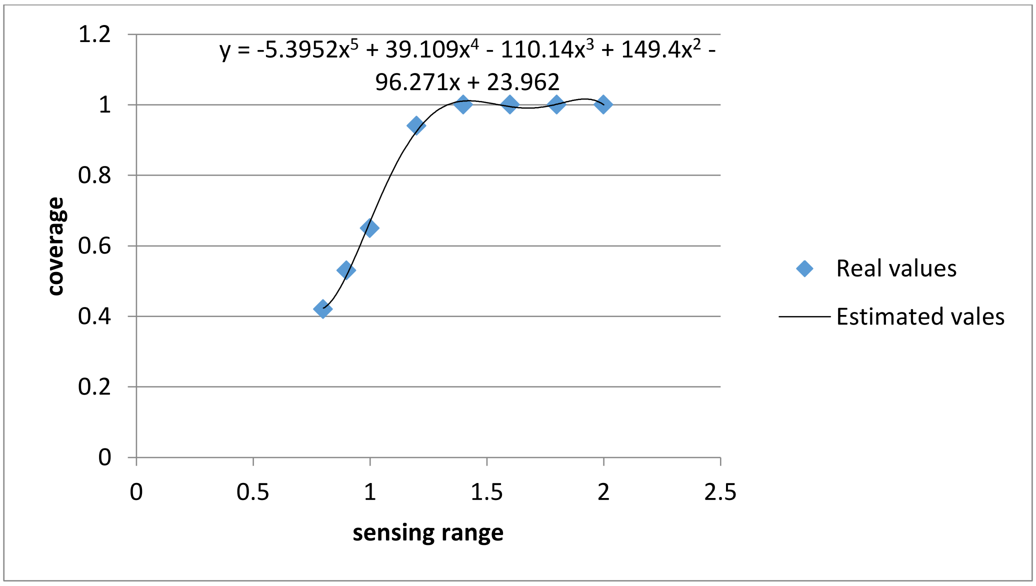


Figure (4.19): sensing range and coverage area for case2, scenario 3

The variable x in equation (4.8) stands for the sensing range and the variable y stands for the coverage area.

$y = -5.3952x^5 + 39.109x^4 - 110.14x^3 + 149.4x^2 - 96.271x + 23.962\ldots$ (4.8)

Example:

- When the sensing range = 0.8 then y =0.42
- When the sensing range = 1.4 then y = 1
- When the sensing range = 0.95 then y = 0.58
- When the sensing range = 1.5 then y = 1

Figure (4.20) presents the simulation results for the five performance metrics.

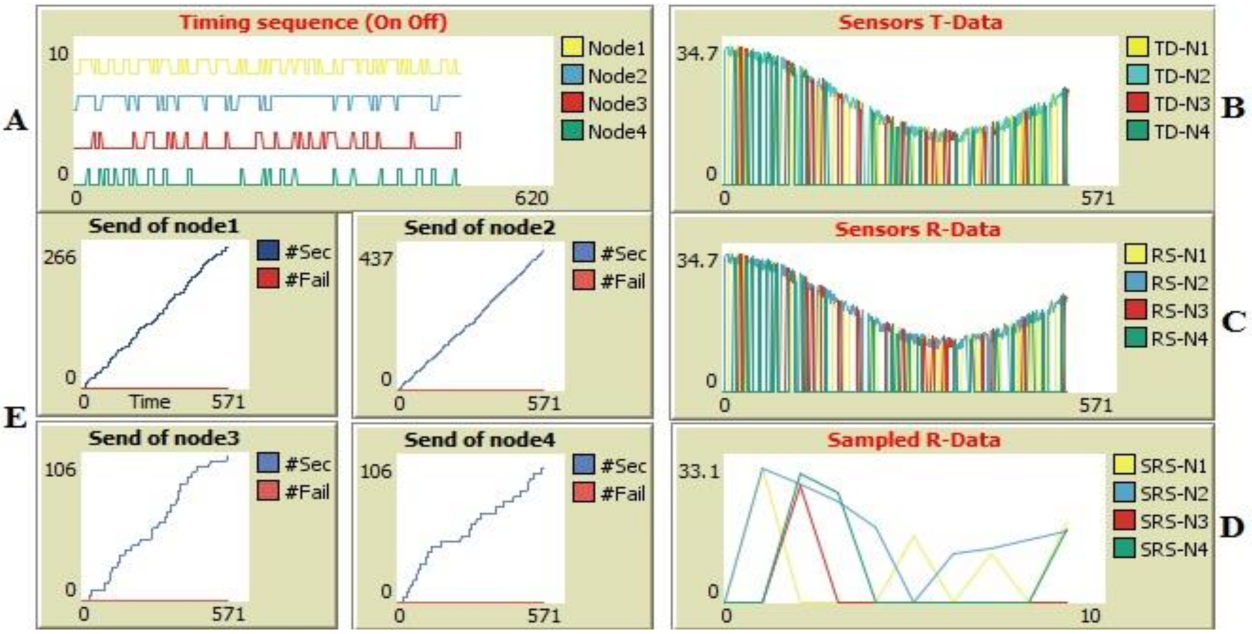


Figure (4.20): performance metrics for case 2, scenario 3.

The behavior of the results in figure (4.20) shows stability and presents the effect of number of nodes clearly.

D. scenario 4: 150 nodes are distributed in a random way in a 400*400 m area and the position of the sink node is at the center point. The sensing range is as indicated previously. Table (4.10) includes the covered area value while using a range of sensing ranges.

Table (4.10): Simulation results for case 2, scenario 4

Area	Number of Nodes	Sensing range	Coverage	Shortest path sample	Shortest path length
400*400	150	0.8	48%	[(node 80) (sink 150)]	9.112
		0.9	61%	[(node 39) (node 58) (sink 150)]	13.061
		1	75%	[(node 53) (node 126) (node 32) (node 80) (sink 150)]	27.837
		1.2	100%	[(node 134) (sink 150)]	4.919
		1.4	100%	[(node 16) (node 42) (sink 150)]	10.765
		1.6	100%	[(node 114) (node 39) (node 58) (sink 150)]	20.587
		1.8	100%	[(node 95) (node 19) (node 66) (sink 150)]	18.680
		2	100%	[(node 54) (sink 150)]	5.093

Figure (4.21) demonstrates the shape of the results in table (4.10) with their expected (Estimated) values. Equation (4.9) is created based on the tabular results for the coverage area.

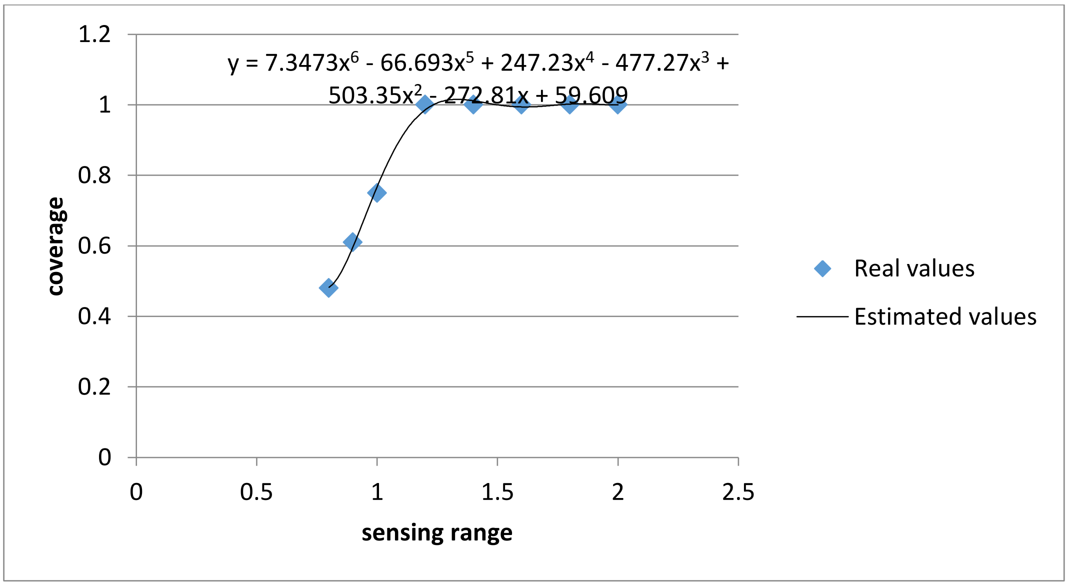


Figure (4.21): sensing range and coverage area for case2, scenario 4

The variable x in equation (4.9) used to represent the sensing range and the variable y used to represent the coverage area.

$y = 7.3473x^6 - 66.693x^5 + 247.23x^4 - 477.27x^3 + 503.35x^2 - 272.81x + 59.609$... (4.9)

Example:

- When the sensing range = 0.8 then y =0.48
- When the sensing range = 1.4 then y = 1
- When the sensing range = 0.95 then y = 0.67
- When the sensing range = 1.3 then y = 1

Figure (4.22) shows the simulation results for the five performance metrics.

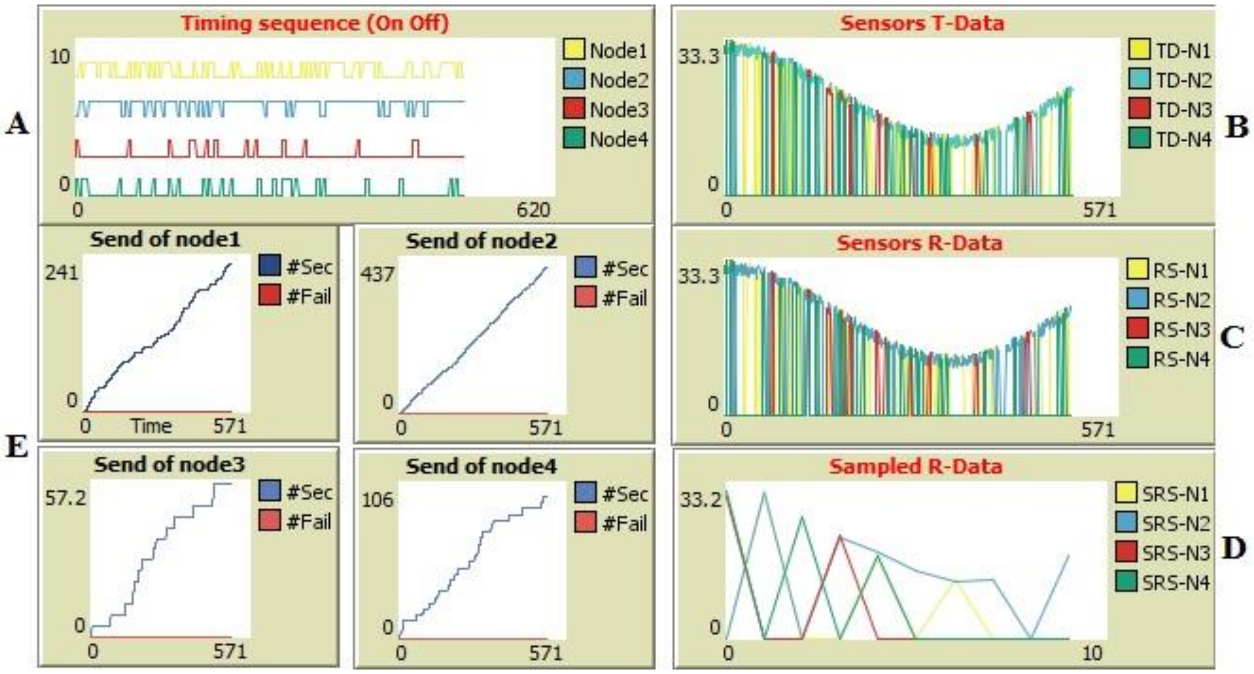


Figure (4.22): performance metrics for case 2, scenario 4.

The behavior of the results in figures (4.16, 4.18, 4.20 and 4.22) shows stability and clearly presents the effect of number of nodes.

4.3.3 Case 3 results

In this case different scenarios are implemented to see the effect of different parameter on the coverage. The area of interest is selected as 500*500 m in this case. A variable number of nodes are distributed. In the scenarios a variable number of nodes with a group of different sensing ranges were used in the same area.

A. scenario 1: At the beginning in the first simulation scenario 80 nodes were randomly deployed in a 500*500 m while the sink node is at the center point. A variable number of sensing range for each sensor node is used as the scenarios in the previous case. Table (4.11) displays the resulted coverage area for each sensing range. The shortest paths from each sensor node to the sink node are also calculated. A sample for the shortest path with its length is displayed in table (4.11).

Table (4.11): Simulation results for case 3, scenario 1

Area	Number of Nodes	Sensing range	Coverage	Shortest path sample	Shortest path length
500*500	80	0.8	18%	[(node 55) (node 62) (node 16) (node 37) (sink 80)]	25.094
		0.9	23%	[(node 30) (node 7) (node 32) (node 8) (node 16) (node 37) (sink 80)]	42.569
		1	29%	[(node 9) (node 16) (node 37) (sink 80)]	19.024
		1.2	42%	[(node 64) (node 57) (sink 80)]	17.837
		1.4	57%	[(node 37) (sink 80)]	7.413
		1.6	74%	[(node 67) (node 57) (sink 80)]	16.555
		1.8	94%	[(node 19) (node 0) (node 44) (node 67) (node 57) (sink 80)]	38.387
		2	100%	[(node 11) (node 57) (sink 80)]	16.389

Figure (4.23) shows the shape of the results in table (4.11) with their expected (Estimated) values. Equation (4.10) is created based on the tabular results for the coverage area.

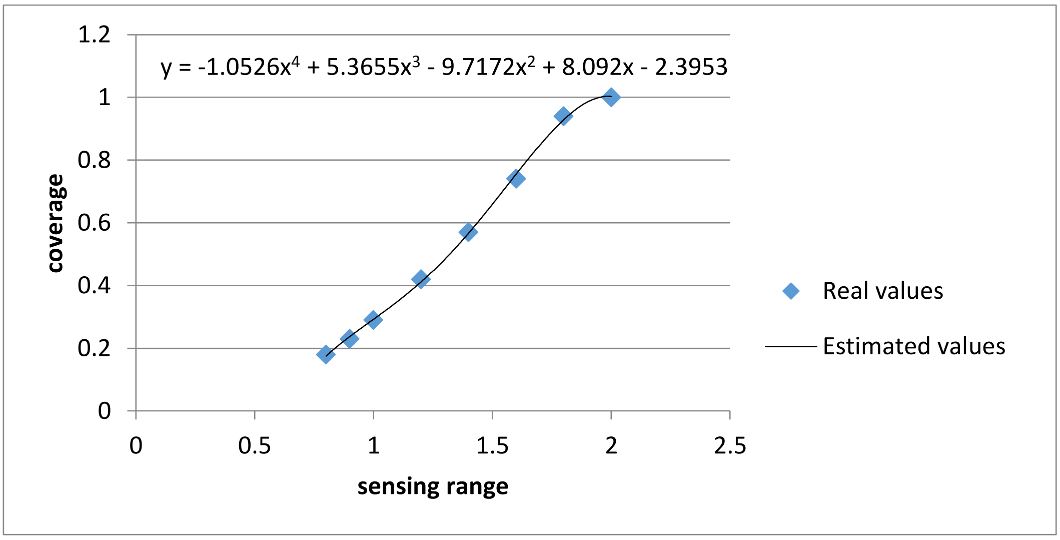


Figure (4.23): sensing range and coverage area for case3, scenario 1

The variable y represents the resulted coverage area and variable x in equation (4.10) represents the suggested sensing range.

$y = -1.0526x^4 + 5.3655x^3 - 9.7172x^2 + 8.092x - 2.3953$ …… (4.10)

Example:

- When the sensing range = 0.9 then y = 0.23
- When the sensing range = 1 then y = 0.29
- When the sensing range = 0.95 then y = 0.26
- When the sensing range = 1.3 then y = 0.48

In figure (4.24) the simulation results for the five performance metrics are presented.

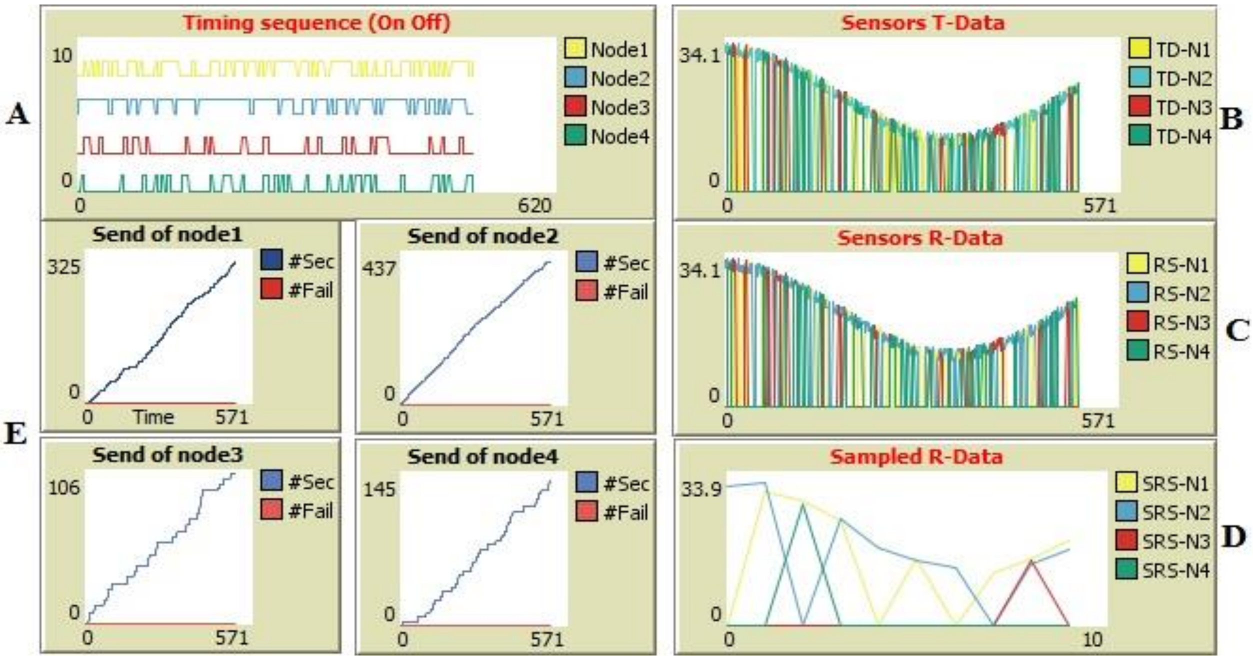


Figure (4.24): performance metrics for case 3, scenario 1.

The behavior of the results in figure (4.24) shows certain stability.

B. scenario 2: In the same area 100 nodes are deployed randomly in the second simulation scenario. The sensing range is as indicated previously. Table (4.12) represents the coverage depending on using a number of different sensing ranges.

Table (4.12): Simulation results for case 3, scenario 2

Area	Number of Nodes	Sensing range	Coverage	Shortest path sample	Shortest path length
500*500	100	0.8	23%	[(node 67) (node 49) (node 20) (sink 100)]	22.528
		0.9	29%	[(node 57) (node 99) (node 59) (node 18) (node 47) (sink 100)]	36.422
		1	36%	[(node 73) (sink 100)]	7.786
		1.2	52%	[(node 24) (node 43) (sink 100)]	14.487
		1.4	71%	[(node 16) (sink 100)]	3.749
		1.6	93%	[(node 80) (node 2) (node 74) (node 73) (sink 100)]	27.767
		1.8	100%	[(node 10) (sink 100)]	1.682
		2	100%	[(node 18) (node 47) (sink 100)]	11.346

Figure (4.25) displays the shape of the results in table (4.12) with their expected (Estimated) values. Equation (4.11) is created based on the tabular results for the coverage area.

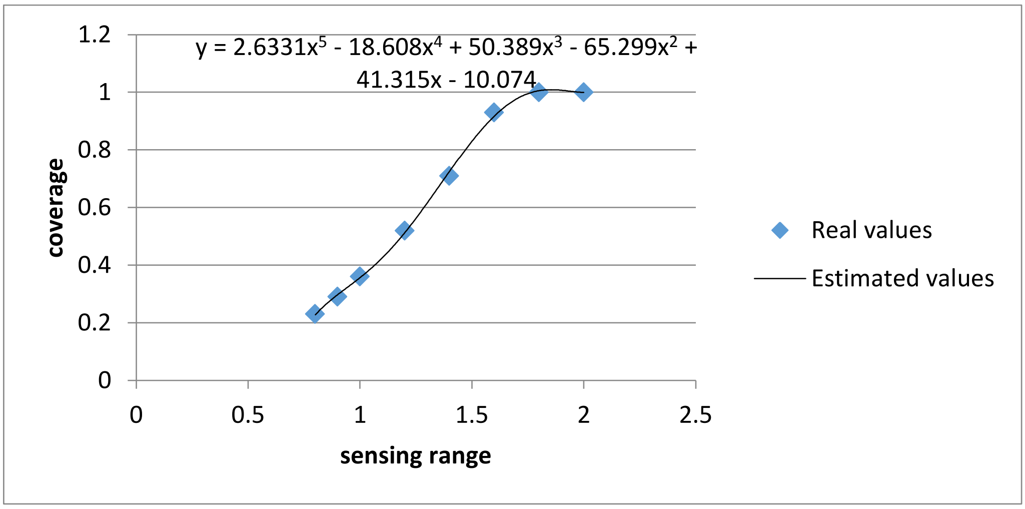


Figure (4.25): sensing range and coverage area for case3, scenario 2

The variable x in equation (4.11) represents the given sensing range and the variable y represents the resulted coverage area.

$y = 2.6331x^5 - 18.608x^4 + 50.389x^3 - 65.299x^2 + 41.315x - 10.074$ ……. (4.11)

Example:

- When the sensing range = 0.9 then y = 0.29
- When the sensing range = 1 then y = 0.36
- When the sensing range = 0.95 then y = 0.32
- When the sensing range = 1.9 then y = 1

Figure (4.26) shows the simulation results for the five performance metrics.

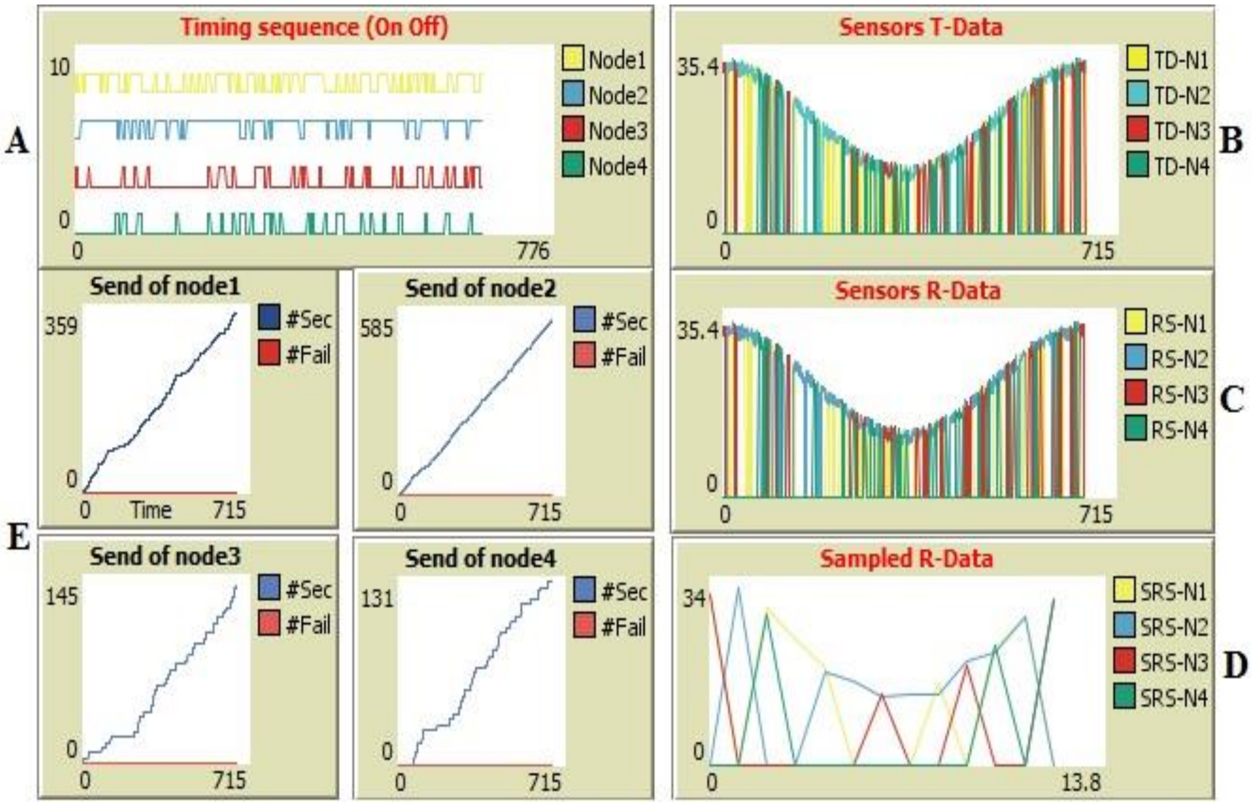


Figure (4.26): performance metrics for case 3, scenario 2.

The behavior of the results in figure (4.26) shows stability and clearly presents the effect of number of nodes.

C. scenario 3: In this scenario the sensing range is as indicated previously and 130 nodes where in 500*500 m area these nodes where randomly deployed while the sink node in the center of the area. The sensing range is as indicated previously. Table (4.13) shows the coverage while variable sensing ranges are used.

Table (4.13): Simulation results for case 3, scenario 3

Area	Number of Nodes	Sensing range	Coverage	Shortest path sample	Shortest path length
500*500	130	0.8	30%	[(node 86) (node 104) (node 4) (sink 130)]	19.415
		0.9	38%	[(node 44) (node 19) (node 103) (node 59) (sink 130)]	29.258
		1	47%	[(node 4) (sink 130)]	5.558
		1.2	68%	[(node 6) (node 0) (sink 130)]	12.524
		1.4	92%	[(node 66) (node 24) (node 6) (node 0) (sink 130)]	31.259
		1.6	100%	[(node 80) (sink 130)]	4.301
		1.8	100%	[(node 19) (node 103) (node 59) (sink 130)]	22.673
		2	100%	[(node 99) (sink 130)]	8.595

Figure (4.27) demonstrates the shape of the results in table (4.13) with their expected (Estimated) values. Equation (4.12) is created based on the tabular results for the coverage area.

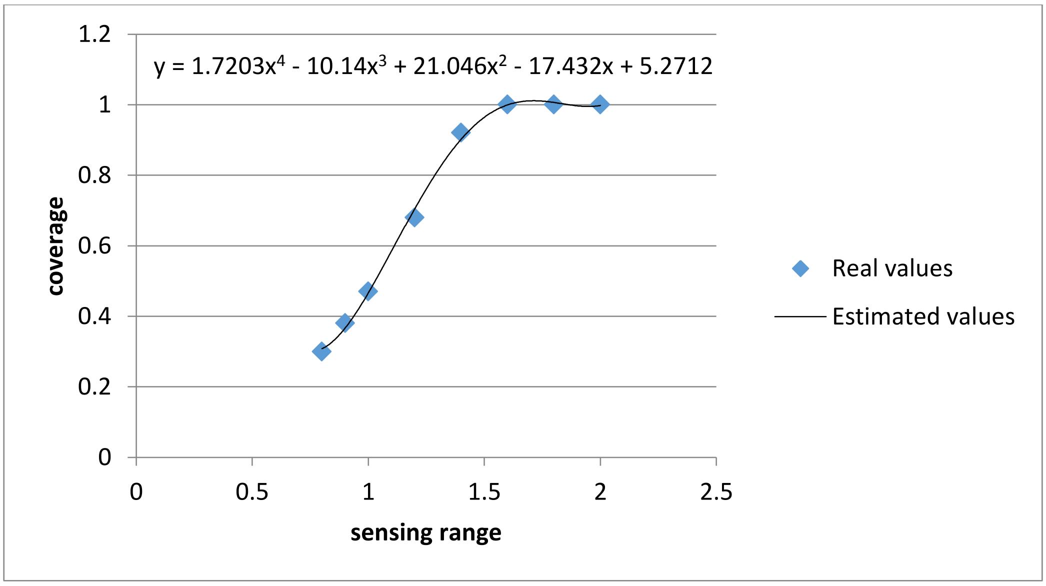


Figure (4.27): sensing range and coverage area for case3, scenario 3

The variable y stands for the coverage area and the variable x in equation (4.12) stands for the sensing range

$$y = 1.7203x^4 - 10.14x^3 + 21.046x^2 - 17.432x + 5.2712 \quad \ldots\ldots \; (4.12)$$

Example:

- When the sensing range = 0.8 then y = 0.30
- When the sensing range = 1 then y = 0.47
- When the sensing range = 0.85 then y = 0.33
- When the sensing range = 1.7 then y = 1

Figure (4.28) shows the simulation results for the five performance metrics.

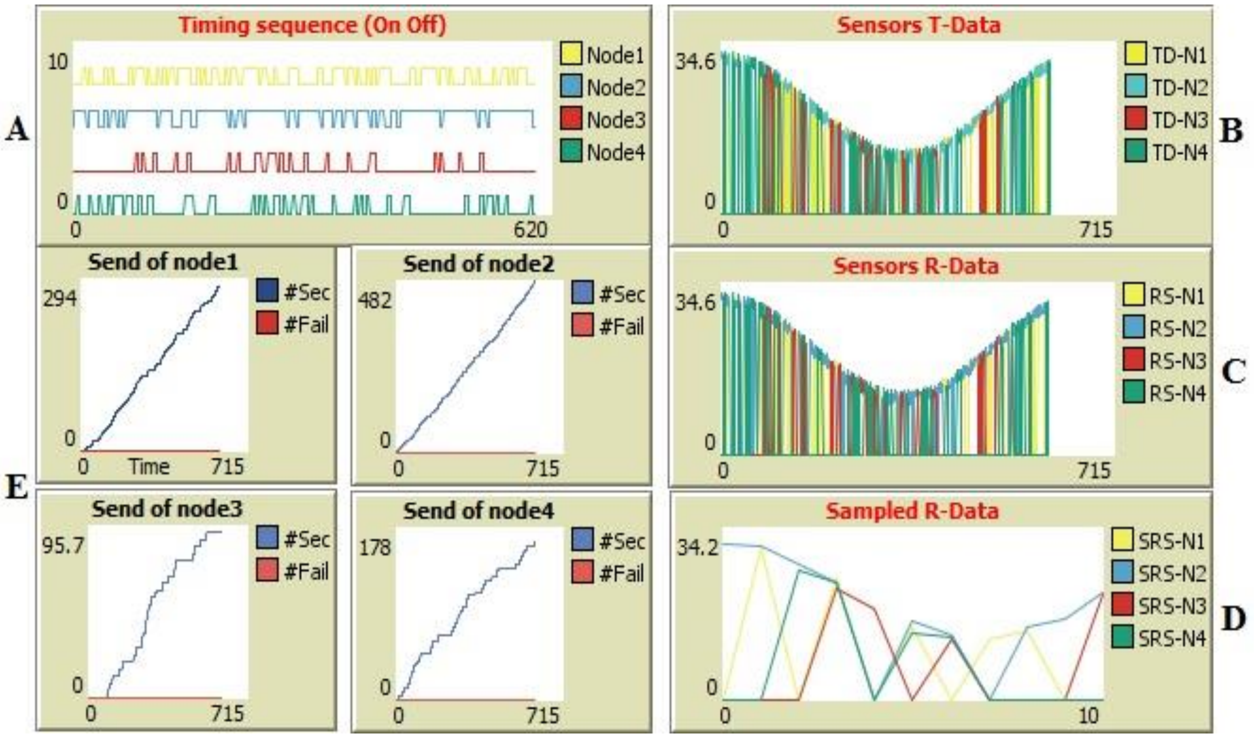


Figure (4.28): performance metrics for case 3, scenario 3.

The behavior of the results in figure (4.28) shows stability and clearly presents the effect of number of nodes.

D. scenario 4: In the last scenario for this case 150 nodes are distributed in a randomly in a 500*500 m area and the sink node positioned at the center point. The sensing range is as indicated previously. Table (4.14) contain the covered area value while using a variable numbers of sensing ranges.

Table (4.14): Simulation results for case 3, scenario 4

Area	Number of Nodes	Sensing range	Coverage	Shortest path sample	Shortest path length
500*500	150	0.8	34%	[(node 31) (node 94) (node 43) (node 77) (node 87) (sink 150)]	34.954
		0.9	44%	[(node 119) (node 45) (node 92) (sink 150)]	22.334
		1	54%	[(node 7) (sink 150)]	6.77
		1.2	78%	[(node 51) (node 7) (sink 150)]	15.097
		1.4	100%	[(node 112) (node 20) (node 9) (sink 150)]	25.984
		1.6	100%	[(node 103) (node 52) (sink 150)]	17.097
		1.8	100%	[(node 32) (node 96) (node 40) (node 56) (sink 150)]	28.669
		2	100%	[(node 45) (node 92) (sink 150)]	14.061

Figure (4.29) displays the shape of the results in table (4.14) with their expected (Estimated) values. Equation (4.13) is created based on the tabular results for the coverage area.

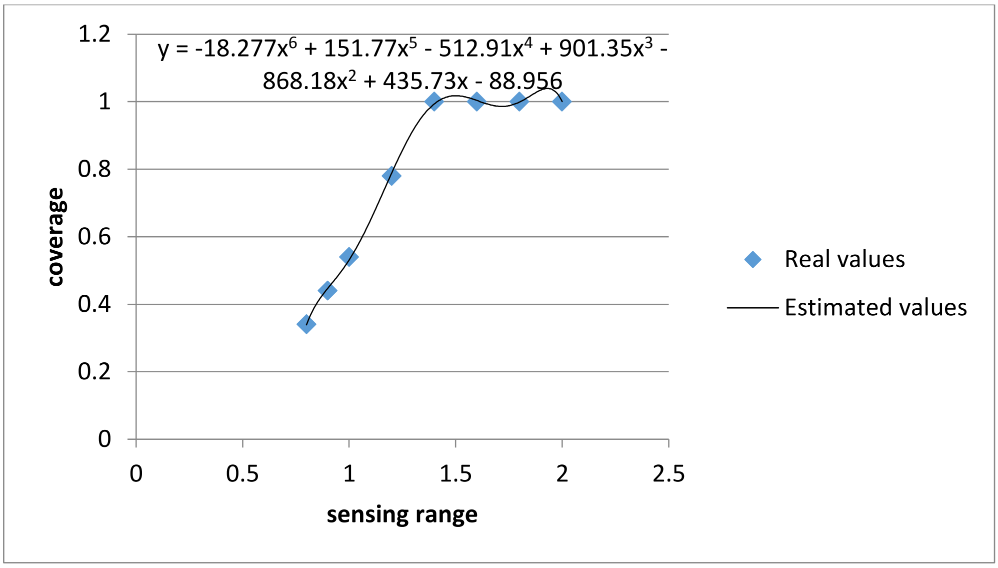


Figure (4.29): sensing range and coverage area for case3, scenario 4

The variable x in equation (4.13) used to represent the sensing range and the variable y used to represent the coverage area.

$y = -18.277x^6 + 151.77x^5 - 512.91x^4 + 901.35x^3 - 868.18x^2 + 435.73x - 88.956..(4.13)$

Example:

- When the sensing range = 0.9 then y =0.44
- When the sensing range = 1.2 then y = 0.78
- When the sensing range = 0.85 then y = 0.40
- When the sensing range = 1.5 then y = 1

Figure (4.30) shows the simulation results for the five performance metrics.

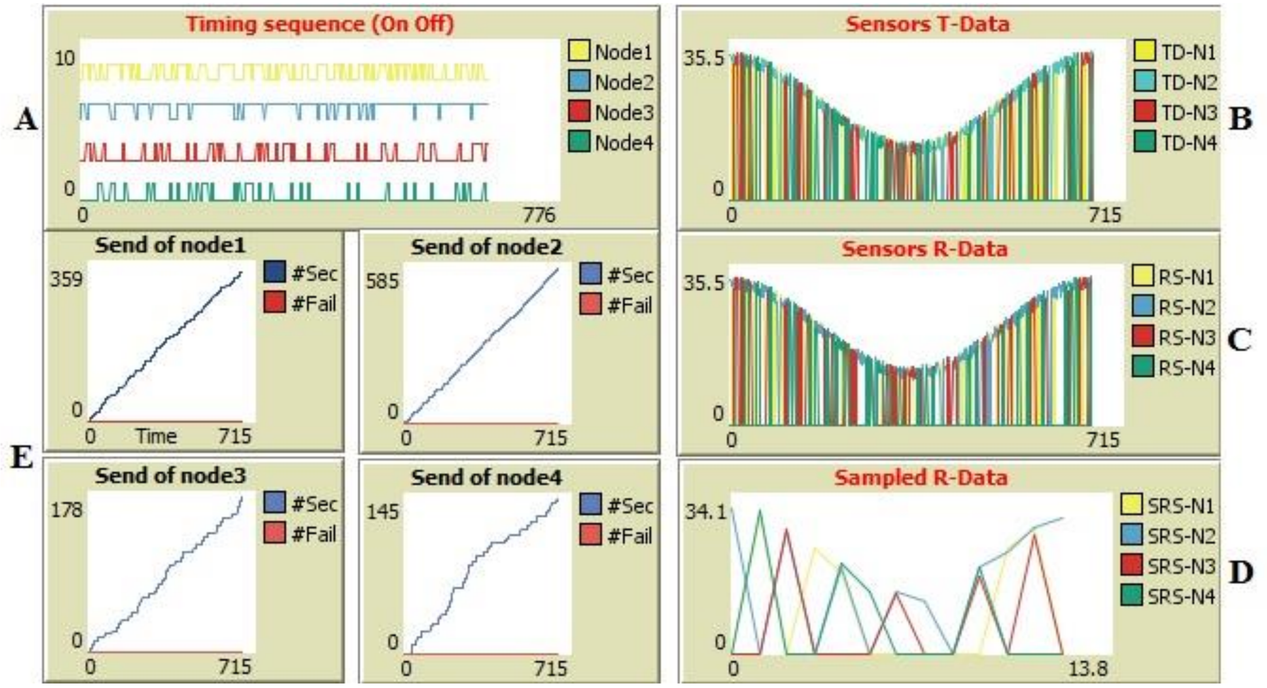


Figure (4.30): performance metrics for case 3, scenario 4.

The behavior of the results in figures (4.24, 4.26, 4.28 and 4.30) shows stability and clearly presents the effect of number of nodes.

4.3.4 Case 4 results

This is the last case in the random distribution and also includes different scenarios which are implemented to notice the effect of multiple parameters on the coverage area. The area to be covered is selected as 600*600 m in this case. Different values for the number of nodes are deployed. In the scenarios a range of sensing ranges were used in the same area.

A. scenario 1: In the first simulation scenario there were 80 randomly deployed nodes in a 600*600 m and the sink node in the center. For each sensor node a variable number of sensing ranges are used as the scenarios in the previous case. The resulted coverage area for each sensing range is displayed in Table (4.15). The shortest paths from the sink node to each sensor node are also calculated. An example for the shortest path with its length is shown in table (4.15).

Table (4.15): Simulation results for case 4, scenario 1

Area	Number of Nodes	Sensing range	Coverage	Shortest path sample	Shortest path length
600*600	80	0.8	15%	[(node 72) (node 70) (node 53) (node 0) (node 17) (sink 80)]	35.501
		0.9	19%	[(node 20) (node 15) (node 5) (node 12) (node 78) (node 36) (node 71) (node 47) (node 74) (sink 80)]	60.488
		1	24%	[(node 54) (sink 80)]	8.170
		1.2	35%	[(node 36) (node 71) (node 47) (node 74) (sink 80)]	24.155
		1.4	47%	[(node 70) (node 53) (node 0) (node 17) (sink 80)]	31.073
		1.6	62%	[(node 53) (node 0) (node 17) (sink 80)]	22.553
		1.8	79%	[(node 74) (sink 80)]	5.952
		2	97%	[(node 40) (sink 80)]	9.357

Figure (4.31) shows the shape of the results in table (4.15) with their expected (Estimated) values. Equation (4.14) is created based on the tabular results for the coverage area.

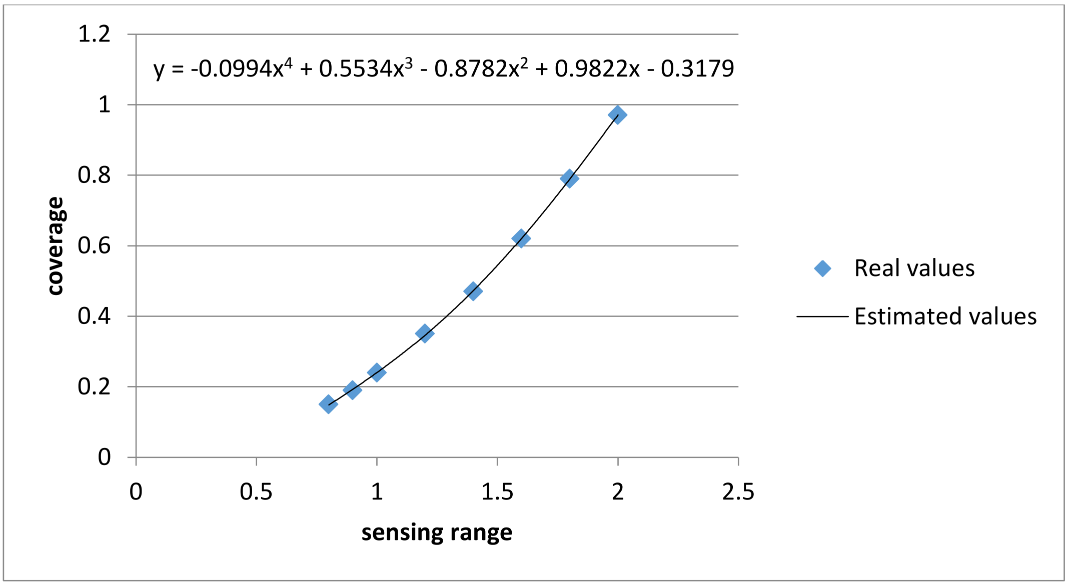


Figure (4.31): sensing range and coverage area for case 4, scenario 1

The variable y represents the resulted coverage area and variable x in equation (4.14) represents the suggested sensing range.

$y = -0.0994x^4 + 0.5534x^3 - 0.8782x^2 + 0.9822x - 0.3179$…… (4.14)

Example:

- When the sensing range = 0.8 then y = 0.15
- When the sensing range = 1 then y = 0.24
- When the sensing range = 0.95 then y = 0.21
- When the sensing range = 1.3 then y = 0.40

Figure (4.32) shows the simulation results for the five performance metrics.

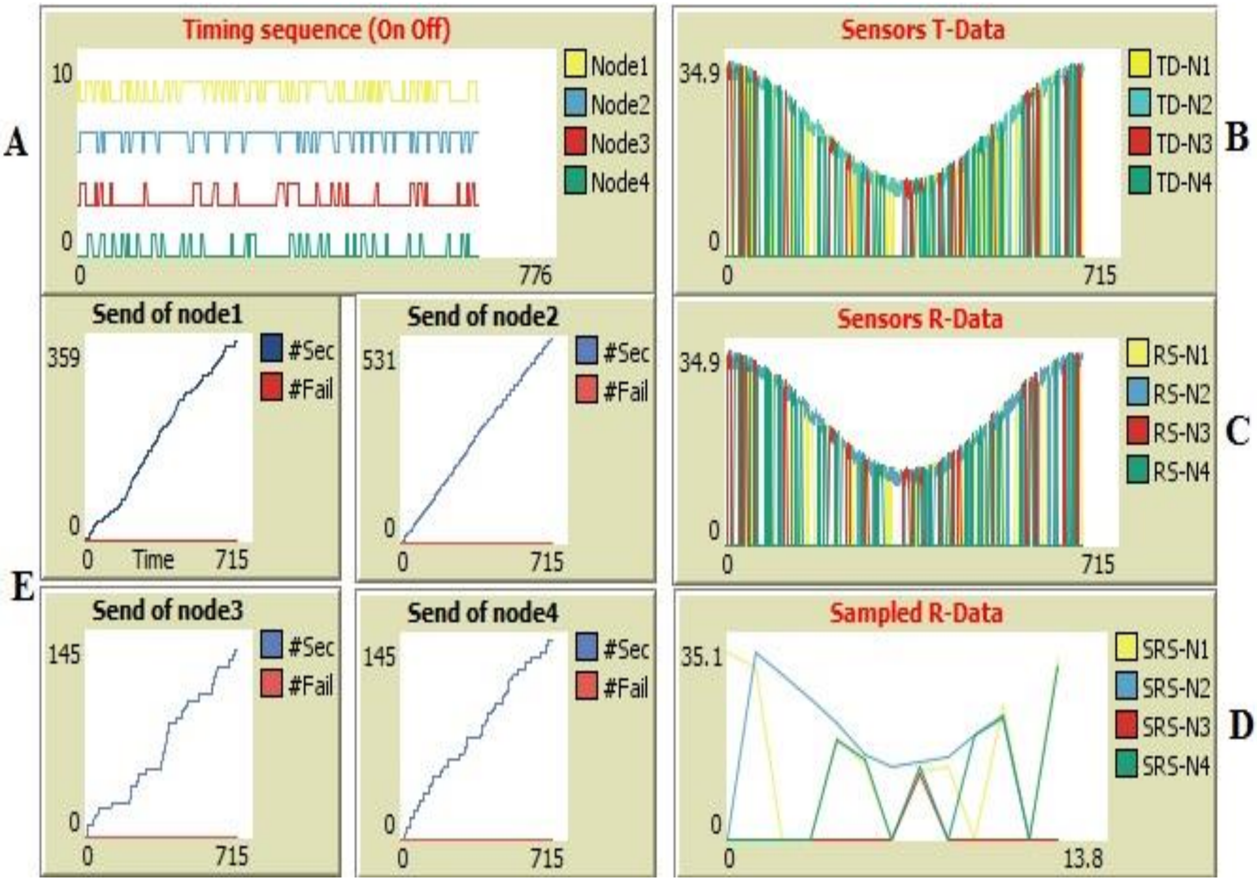


Figure (4.32): performance metrics for case 4, scenario 1.

The behavior of the results in figure (4.32) shows certain stability.

B. scenario 2: In the same area 100 nodes are deployed randomly in the second simulation scenario. The sensing range is as indicated previously. Table (4.16) estimates the coverage depending on using a number of different sensing ranges.

Table (4.16): Simulation results for case 4, scenario 2

Area	Number of Nodes	Sensing range	Coverage	Shortest path sample	Shortest path length
600*600	100	0.8	19%	[(node 92) (node 35) (node 73) (node 2) (node 77) (sink 100)]	40.133
		0.9	24%	[(node 26) (node 46) (node 79) (node 4) (node 84) (node 12) (node 77) (sink 100)]	54.073
		1	30%	[(node 69) (node 70) (node 12) (node 77) (sink 100)]	31.161
		1.2	43%	[(node 5) (node 24) (node 22) (node 34) (sink 100)]	29.668
		1.4	59%	[(node 35) (node 73) (node 2) (node 77) (sink 100)]	33.703
		1.6	77%	[(node 23) (node 44) (node 77) (sink 100)]	21.299
		1.8	98%	[(node 24) (node 22) (node 34) (sink 100)]	23.271
		2	100%	[(node 12) (node 77) (sink 100)]	13.641

Figure (4.33) displays the shape of the results in table (4.16) with their expected (Estimated) values. Equation (4.15) is created based on the tabular results for the coverage area.

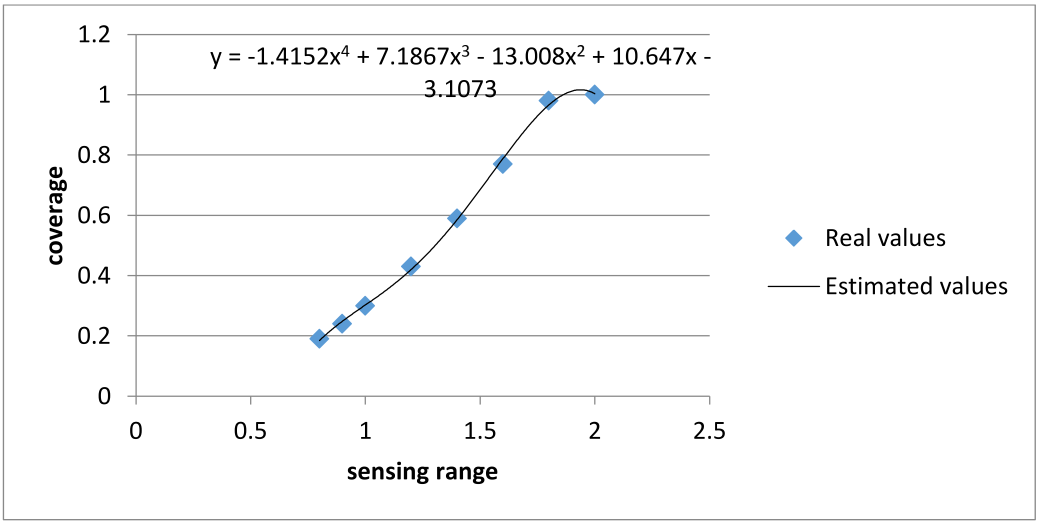


Figure (4.33): sensing range and coverage area for case 4, scenario 2

The variable x in equation (4.15) represents the given sensing range and the variable y represents the resulted coverage area.

$$y = -1.4152x^4 + 7.1867x^3 - 13.008x^2 + 10.647x - 3.1073 \quad ……. (4.15)$$

Example:

- When the sensing range = 0.9 then y = 0.24
- When the sensing range = 1.4 then y = 0.59
- When the sensing range = 0.95 then y = 0.27
- When the sensing range = 1.5 then y = 0.68

Figure (4.34) shows the simulation results for the five performance metrics.

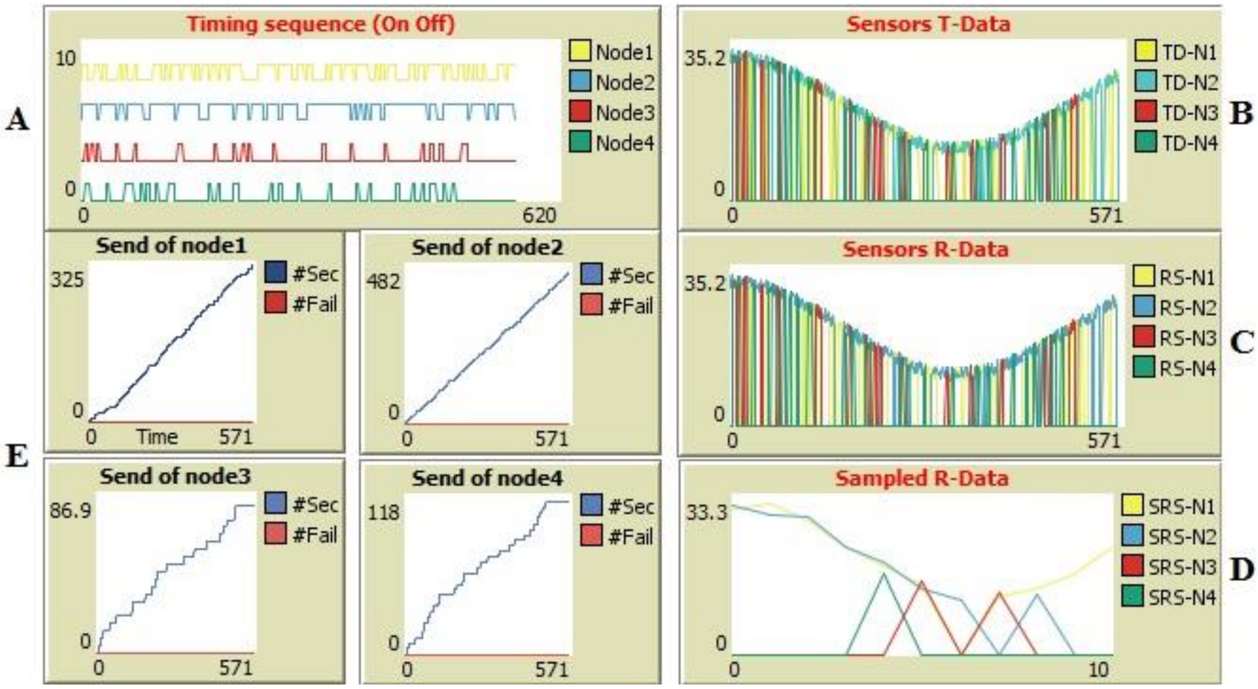


Figure (4.34): performance metrics for case 4, scenario 2.

The behavior of the results in figure (4.34) shows stability and clearly presents the effect of number of nodes.

C. scenario 3: In this scenario the sensing range is as indicated previously and 130 nodes where in a 600*600 m area these nodes where randomly deployed while the sink node in the center of the area. The sensing range is as indicated previously. Table (4.17) shows the values for the coverage with using variable sensing ranges.

Table (4.17): Simulation results for case 4, scenario 3

Area	Number of Nodes	Sensing range	Coverage	Shortest path sample	Shortest path length
600*600	130	0.8	25%	[(node 16) (node 51) (node 20) (node 35) (node 5) (sink 130)]	33.351
		0.9	31%	[(node 38) (node 31) (node 34) (node 118) (node 17) (node 46) (sink 130)]	41.866
		1	39%	[(node 54) (node 57) (node 13) (node 64) (sink 130)]	28.842
		1.2	56%	[(node 125) (node 28) (node 64) (sink 130)]	20.938
		1.4	77%	[(node 105) (sink 130)]	4.646
		1.6	100%	[(node 59) (node 48) (node 89) (sink 130)]	19.750
		1.8	100%	[(node 10) (node 46) (sink 130)]	10.914
		2	100%	[(node 96) (sink 130)]	8.649

Figure (4.35) demonstrates the shape of the results in table (4.17) with their expected (Estimated) values. Equation (4.16) is created based on the tabular results for the coverage area.

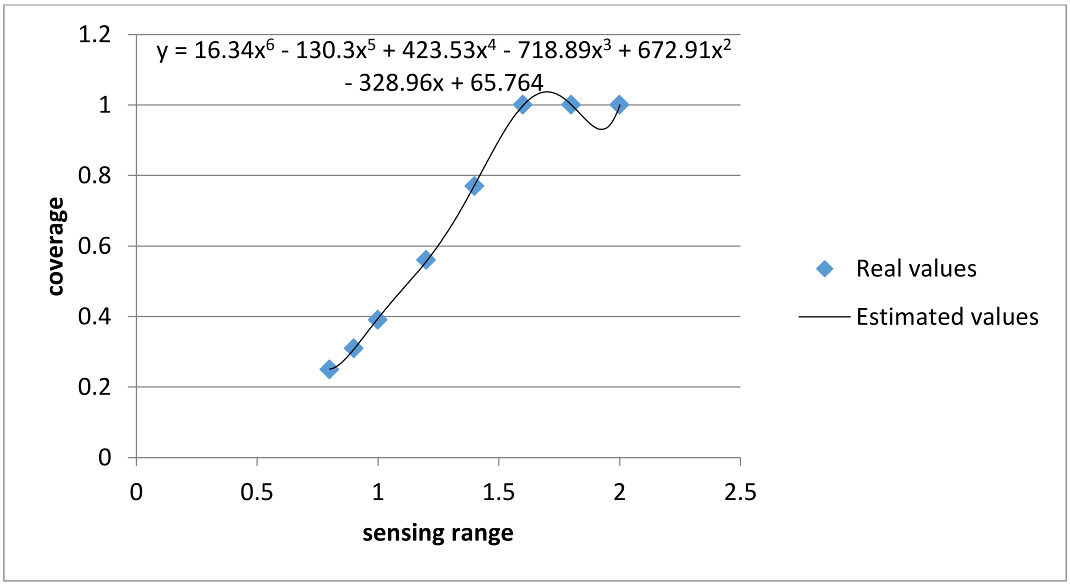


Figure (4.35): sensing range and coverage area for case 4, scenario 3

The variable y stands for the coverage area and the variable x in equation (4.16) stands for the sensing range

$y = 16.34x^6 - 130.3x^5 + 423.53x^4 - 718.89x^3 + 672.91x^2 - 328.96x + 65.764$…. (4.16)

Example:

- When the sensing range = 0.8 then y =0.25
- When the sensing range = 0.9 then y = 0.31
- When the sensing range = 0.85 then y = 0.27
- When the sensing range = 1.3 then y = 0.65

Figure (4.36) shows the simulation results for the five performance metrics.

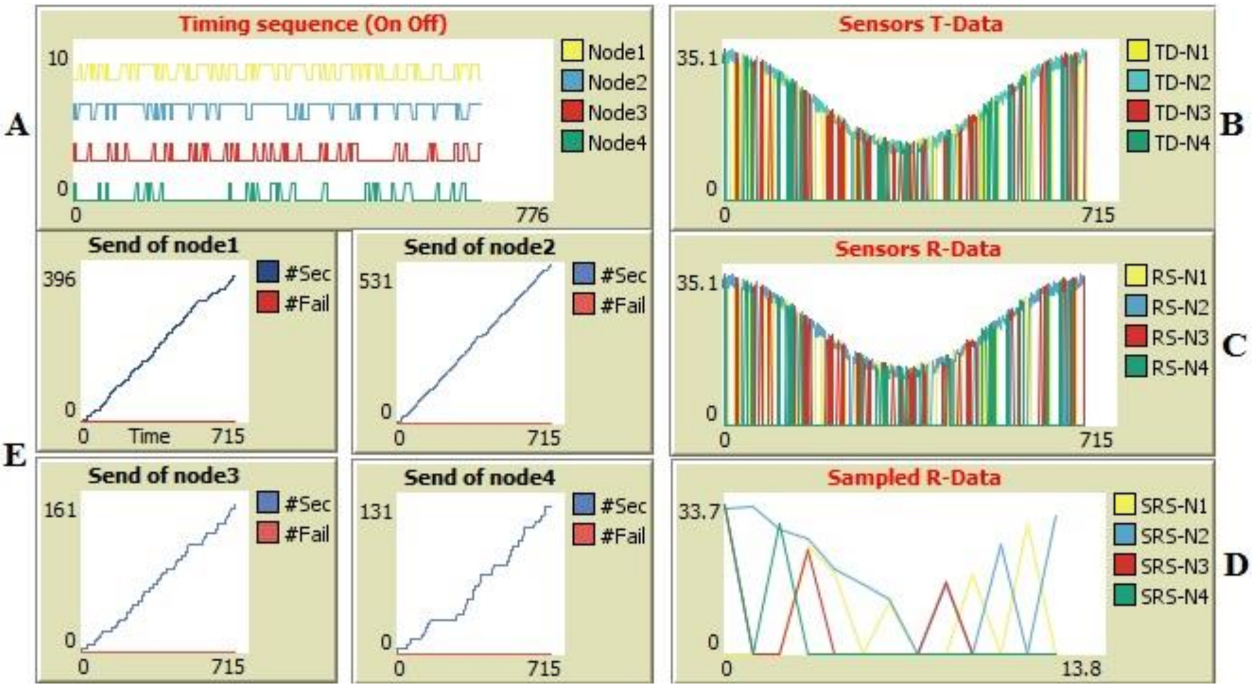


Figure (4.36): performance metrics for case 4, scenario 3.

The behavior of the results in figure (4.36) shows stability and clearly presents the effect of number of nodes.

D. scenario 4: finally 150 nodes are distributed in a randomly in a 600*600 m area and the sink node at the center point. The sensing range is as indicated previously. Table (4.18) contain the covered area value while using a variable numbers of sensing ranges.

Table (4.18): Simulation results for case 4, scenario 4

Area	Number of Nodes	Sensing range	Coverage	Shortest path sample	Shortest path length
600*600	150	0.8	29%	[(node 54) (sink 150)]	8.093
		0.9	36%	[(node 107) (node 75) (node 106) (node 56) (node 2) (node 47) (sink 150)]	44.705
		1	45%	[(node 75) (node 106) (node 56) (node 2) (node 47) (sink 150)]	36.136
		1.2	65%	[(node 130) (node 124) (node 149) (node 87) (sink 150)]	24.550
		1.4	89%	[(node 19) (node 1) (sink 150)]	16.018
		1.6	100%	[(node 101) (node 67) (node 115) (sink 150)]	21.834
		1.8	100%	[(node 121) (node 54) (sink 150)]	13.296
		2	100%	[(node 83) (node 130) (node 124) (node 149) (node 87) (sink 150)]	32.391

Figure (4.37) displays the shape of the results in table (4.18) with their expected (Estimated) values. Equation (4.17) is created based on the tabular results for the coverage area.

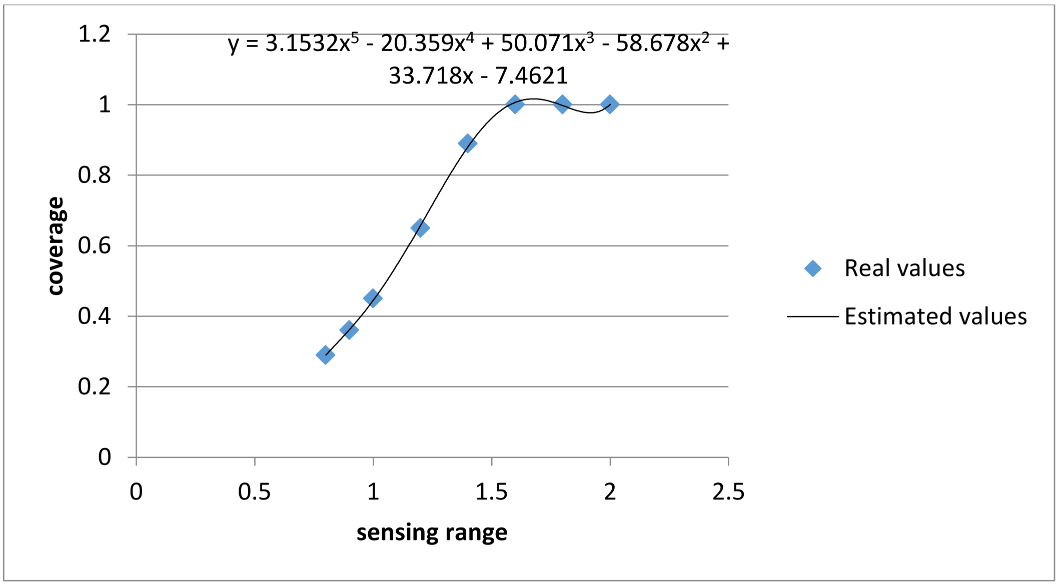


Figure (4.37): sensing range and coverage area for case 4, scenario 4

The variable x in equation (4.17) used to represent the sensing range and the variable y used to represent the coverage area.

$y = 3.1532x^5 - 20.359x^4 + 50.071x^3 - 58.678x^2 + 33.718x - 7.4621$ ……..(4.17)

Example

- When the sensing range = 0.9 then y =0.36
- When the sensing range = 1.2 then y = 0.65
- When the sensing range = 0.95 then y = 0.40

- When the sensing range = 1.5 then y = 0.95

Figure (4.38) shows the simulation results for the five performance metrics.

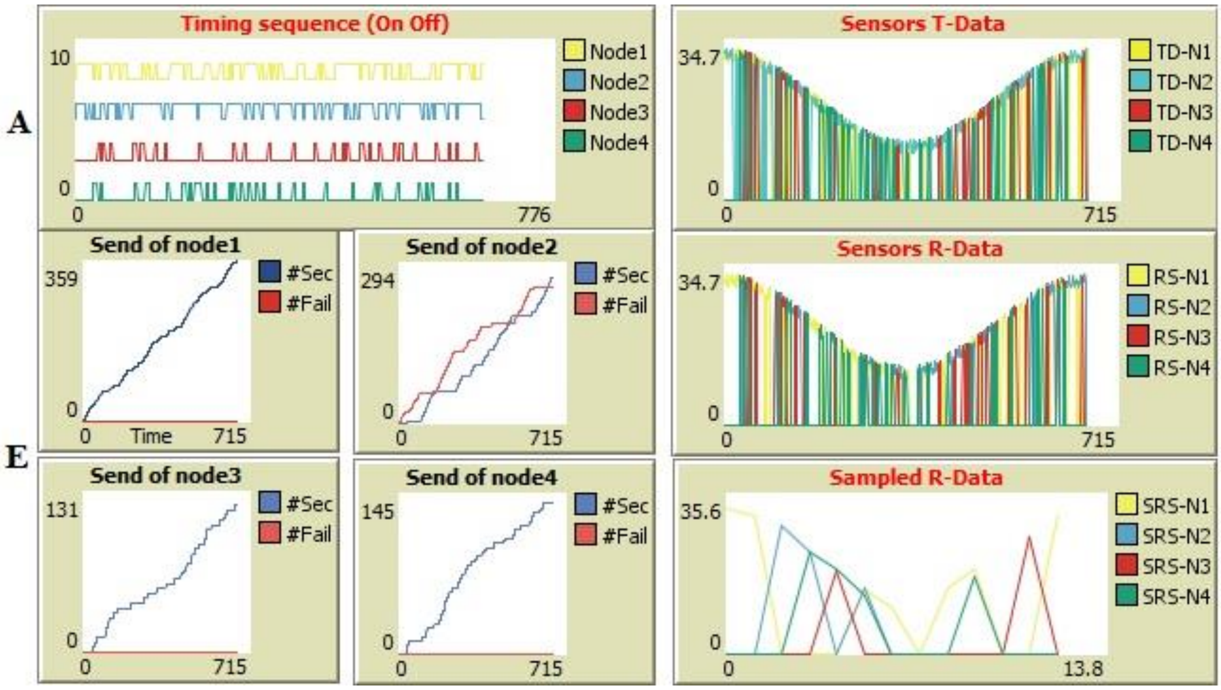


Figure (4.38): performance metrics for case 4, scenario 4.

The behavior of the results in figures (4.32, 4.34, 4.36 and 4.38) shows stability and clearly presents the effect of number of nodes.

CHAPTER FIVE

Conclusions and Future Works

Chapter five

Conclusions and future works

5.1 Conclusions

1. Sensor's coverage area is significant in creating any WSN to ensure certain level of the target observation. Modeling and analyzing its effects on other network metrics and parameters are useful.

2. The fundamentals of graph theory can be utilized to study and present the network topology and the coverage models in WSNs to improve their performance.

3. Considering the sensors deployment, transmission and sensing ranges will improve the sensors ability as well as the network reliability in detecting any event inside the monitored area.

4. Regression models can be created and efficiently utilized to estimate the thresholds, coverage and sensing ranges for different WSN environments.

5. The created equations in this thesis are suitable to be used in current and future studies for different WSNs applications.

6. Increasing the sensing range value will result in an increased coverage level of the selected area.

7. Increasing the number of used sensor nodes will result in an increased coverage level of the selected area.

5.2 Future works

There are several aspects which are not handled in this study:

1. Analyze and evaluate the coverage models in linear wireless sensor networks.
2. Mention the effect of the node failure on the same proposed network
3. Apply multi objective optimization approach to evaluate the WSN coverage and connectivity.
4. Apply the same approaches to evaluate the wireless vehicular networks.
5. Mention to the security issues on the same proposed network.
6. Test another simulator and compare the results.

References

REFERENCES:

Abbasian Dehkordi, S., Farajzadeh, K., Rezazadeh, J., Farahbakhsh, R., Sandrasegaran, K., & Abbasian Dehkordi, M. (2020). A survey on data aggregation techniques in IoT sensor networks. *Wireless Networks*, *26*(2), 1243-1263.

Abo-Zahhad, M., Farrag, M., Ali, A., & Amin, O. (2015, March). An energy consumption model for wireless sensor networks. In *5th International Conference on Energy Aware Computing Systems & Applications* (pp. 1-4). IEEE.

Akbarzadeh, V., Gagné, C., Parizeau, M., Argany, M., & Mostafavi, M. A. (2013). Probabilistic sensing model for sensor placement optimization based on line-of-sight coverage. *IEEE Transactions on Instrumentation and Measurement*, *62*(2), 293–303. https://doi.org/10.1109/TIM.2012.2214952

Akyildiz, Ian F., and Mehmet Can Vuran. 2010. *Wireless Sensor Networks*. Vol. 4. John Wiley & Sons.

Amutha, J., Sharma, S., & Nagar, J. (2020). WSN Strategies Based on Sensors, Deployment, Sensing Models, Coverage and Energy Efficiency: Review, Approaches and Open Issues. *Wireless Personal Communications*, *111*(2), 1089–1115. https://doi.org/10.1007/s11277-019-06903-z

Arkes, J. (2019). *Regression analysis: A practical introduction*. Routledge.

Aslam, S., Farooq, F., & Sarwar, S. (2009, December). Power consumption in wireless sensor networks. In *Proceedings of the 7th International Conference on Frontiers of Information Technology* (pp. 1-9)

Aziz, N. A. A., Aziz, K. A., & Ismail, W. Z. W. (2009). Coverage strategies for wireless sensor networks. *World Academy of Science, Engineering and Technology*, *38*(2), 145–150.

Balamurali, R., & Kathiravan, K. (2015, July). An analysis of various routing protocols for Precision Agriculture using Wireless Sensor Network. In *2015 IEEE Technological Innovation in ICT for Agriculture and Rural Development (TIAR)* (pp. 156-159). IEEE.

Boualem, A., Dahmani, Y., Maatoug, A., & De-runz, C. (2018). Area coverage optimization in wireless sensor network by semi-random deployment. *SENSORNETS 2018 - Proceedings of the 7th International Conference on Sensor Networks, 2018-Janua*(Sensornets), 85–90. https://doi.org/10.5220/0006581900850090

Chang, M., & Pakzad, S. N. (2014). Optimal sensor placement for modal identification of bridge systems considering number of sensing nodes. *Journal of Bridge Engineering, 19*(6), 04014019.

Chen, Y. N., & Chen, C. (2018). Sensor deployment under probabilistic sensing model. *ACM International Conference Proceeding Series*, 33–36. https://doi.org/10.1145/3234664.3234683

Djedouboum, Asside Christian, Ado Adamou Abba Ari, Abdelhak Mourad Gueroui, Alidou Mohamadou, and Zibouda Aliouat. 2018. "Big Data Collection in Large-Scale Wireless Sensor Networks." *Sensors* 18(12):4474.

Elhoseny, M., Tharwat, A., Farouk, A., & Hassanien, A. E. (2017). K-Coverage Model Based on Genetic Algorithm to Extend WSN Lifetime. *IEEE Sensors Letters*, *1*(4), 1–4. https://doi.org/10.1109/lsens.2017.2724846

Fawziyah Waleed Khazeel ,"Modeling a suggested Modified Predictive Wakeup Medium Access Control on WSN", An MSc. Thiess submitted to the college of Sciences-University of Babylon –Iraq, In partial Fulfillment of the Requirement for the degree of Master of sciences in Computer Sciences,2014.

Fen, L., Kai, D., Fanzhi, M., & Weiyan, Z. (2016). *Coverage in Wireless Sensor Network Based on Probabilistic Sensing Model*. www.worldscientific.com

Goodall, P., Sharpe, R., & West, A. (2019). A data-driven simulation to support remanufacturing operations. *Computers in Industry, 105,* 48-60.

Guo, J., & Jafarkhani, H. (2016). Sensor Deployment With Limited Communication Range in Homogeneous and Heterogeneous Wireless Sensor Networks. *IEEE Transactions on Wireless Communications*, *15*(10), 6771–6784. https://doi.org/10.1109/TWC.2016.2590541

Hossain, A. (2014). Sensing and Link Model for Wireless Sensor Network : Coverage and Connectivity Analysis. *ArXiv:1406.1275*, *1*, 1–5.

Hossain, A., Biswas, P. K., & Chakrabarti, S. (2008). Sensing models and its impact on network coverage in wireless sensor network. *IEEE Region 10 Colloquium and 3rd International Conference on Industrial and Information Systems, ICIIS 2008*, *January*. https://doi.org/10.1109/ICIINFS.2008.4798455

Hossain, A., Chakrabarti, S., & Biswas, P. K. (2012). Impact of sensing model on wireless sensor network coverage. *IET Wireless Sensor Systems*, *2*(3), 272–281. https://doi.org/10.1049/iet-wss.2011.0101

IEEE Systems, M., & Institute of Electrical and Electronics Engineers. (2016). *Advanced Mechatronic Systems : Nov. 30 - Dec. 3, 2016, Melbourne, Australia*. 412–416.

Institute of Electrical and Electronics Engineers., & IEEE Antennas and Propagation Society. (n.d.). 2012 8th International Conference on Wireless Communications, Networking and Mobile Computing : WiCOM2012 : September 21-23, 2012, Shanghai, China.

Jindal DAV College, V. (2018). History and Architecture of Wireless Sensor Networks for Ubiquitous Computing. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, *7*(2), 2278–1323.

Kassim, M. R. M., & Harun, A. N. (2016, October). Applications of WSN in agricultural environment monitoring systems. In *2016 international conference on information and communication technology convergence (ICTC)* (pp. 344-349). IEEE.

Kim, J., & Yoo, Y. (2020). Sensor node activation using bat algorithm for connected target coverage in wsns. *Sensors (Switzerland)*, *20*(13), 1–24. https://doi.org/10.3390/s20133733

Kocakulak, M., & Butun, I. (2017, January). An overview of Wireless Sensor Networks towards internet of things. In *2017 IEEE 7th annual computing and communication workshop and conference (CCWC)* (pp. 1-6). Ieee.

Makowski, D., Ben-Shachar, M. S., Patil, I., & Lüdecke, D. (2020). Methods and algorithms for correlation analysis in R. *Journal of Open Source Software*, *5*(51), 2306.

Mini, & Pal, A. (2021). Estimating the coverage performance of a wireless sensor network considering boundary effects in the presence of sensor failure. *Evergreen*, *8*(3), 601–609. https://doi.org/10.5109/4491652

Parwez, Md Salik, Danda B. Rawat, and Moses Garuba. 2017. "Big Data Analytics for User-Activity Analysis and User-Anomaly Detection in Mobile Wireless Network." *IEEE Transactions on Industrial Informatics* 13(4):2058–65.

Priyadarshi, R., & Gupta, B. (2020). Coverage area enhancement in wireless sensor network. *Microsystem Technologies*, *26*(5), 1417-1426

Qin, N. N., & Chen, J. le. (2018). An area coverage algorithm for wireless sensor networks based on differential evolution. *International Journal of Distributed Sensor Networks*, *14*(8). https://doi.org/10.1177/1550147718796734

Railsback, S., Ayllón, D., Berger, U., Grimm, V., Lytinen, S., Sheppard, C., & Thiele, J. C. (2017). Improving execution speed of models implemented in NetLogo.

Ramson, S. J., & Moni, D. J. (2017, February). Applications of wireless sensor networks—A survey. In *2017 international conference on innovations in electrical, electronics, instrumentation and media technology (ICEEIMT)* (pp. 325-329). IEEE.

Savitha M, Shantala Devi Patil ,"A Survey on Clustering Techniques in Wireless Sensor Networks", International Conference on Advances in Computer and Communication Engineering(ACCE-2014). www.ijetae.com

Seah, W. K., Eu, Z. A., & Tan, H. P. (2009, May). Wireless sensor networks powered by ambient energy harvesting (WSN-HEAP)-Survey and challenges. In *2009 1st International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology* (pp. 1-5). Ieee.

Shan, A., Xu, X., & Cheng, Z. (2016). Target coverage in wireless sensor networks with probabilistic sensors. *Sensors*, *16*(9), 1372.

Sharma, V., Patel, R. B., Bhadauria, H. S., & Prasad, D. (2016). Depl oyment schemes in wireless sensor network to achieve blanket coverage in large-scale open area: A review. *Egyptian Informatics Journal*, *17*(1), 45–56. https://doi.org/10.1016/j.eij.2015.08.003

Singh, M. K., Amin, S. I., Imam, S. A., Sachan, V. K., & Choudhary, A. (2018, October). A Survey of Wireless Sensor Network and its types. In *2018 international conference on advances in computing, communication control and networking (ICACCCN)* (pp. 326-330). IEEE.

Tarnaris, K., Preka, I., Kandris, D., & Alexandridis, A. (2020). Coverage and k-coverage optimization in wireless sensor networks using computational intelligence methods: a comparative study. *Electronics, 9*(4), 675.

Toth, C., & Jóźków, G. (2016). Remote sensing platforms and sensors: A survey. *ISPRS Journal of Photogrammetry and Remote Sensing*, *115,* 22-36.

Tripathi, A., Gupta, H. P., Dutta, T., Mishra, R., Shukla, K. K., & Jit, S. (2018). Coverage and Connectivity in WSNs: A Survey, Research Issues and Challenges. *IEEE Access*, *6,* 26971–26992. https://doi.org/10.1109/ACCESS.2018.2833632

Veličković, V. (2015). What everyone should know about statistical correlation. *American Scientist*, *103*(1), 26-29.

Walpole R. E. Myers R. H. Myers S. L. & Ye K. (2017). *Probability & statistics for engineers & scientists* (Ninth edition Global). Pearson Education South Asia Pte.

Wang, Y. C., & Tseng, Y. C. (2008). Distributed deployment schemes for mobile wireless sensor networks to ensure multilevel coverage. *IEEE Transactions on Parallel and Distributed Systems*, *19*(9), 1280–1294. https://doi.org/10.1109/TPDS.2007.70808

Wilensky, U., & Rand, W. (2015). *An Introduction to Agent-Based Modeling: Modeling Natural, Social, and Engineered Complex Systems with NetLogo*. The MIT Press. http://www.jstor.org/stable/j.ctt17kk851

Xia, F. (2009). Wireless sensor technologies and applications. *sensors*, *9*(11), 8824-8830.

Xie, T., Zhang, C., Zhang, Z., & Yang, K. (2018). Utilizing active sensor nodes in smart environments for optimal communication coverage. *IEEE Access*, *7*, 11338-11348.

Yildiz, Orcun. 2017. "Efficient Big Data Processing on Large-Scale Shared Platforms: Managing I/Os and Failure."

Zhang, P., Nevat, I., Peters, G. W., Septier, F., & Osborne, M. A. (2018). Spatial Field Reconstruction and Sensor Selection in Heterogeneous Sensor Networks with Stochastic Energy Harvesting. *IEEE Transactions on Signal Processing*, *66*(9), 2245–2257. https://doi.org/10.1109/TSP.2018.2802452

Zorlu, O., & Sahingoz, O. K. (2016, July). Increasing the coverage of homogeneous wireless sensor network by genetic algorithm based deployment. In *2016 Sixth International Conference on Digital Information and Communication Technology and its Applications (DICTAP)* (pp. 109-114). IEEE.

الخلاصة

تمتلك شبكات الاستشعار اللاسلكية (WSNs) العديد من التطبيقات المهمة. تُستخدم المستشعرات بشكل أساسي لتتبع أو مراقبة أي تغييرات في المنطقة المحيطة. يمكن استخدامها للمراقبة وجمع المعلومات بعد ذلك ومعالجة البيانات المحسوسة ونقلها إلى المحطة المركزية.

منطقة التغطية هي مقياس أداء أساسي في تقييم كل WSN. إنه يمثل قدرة الشبكة على تغطية ومراقبة مجالات اهتمام معينة. تعتمد منطقة التغطية بشكل أساسي على نصف قطر استشعار المستشعر.

في هذا العمل ، تم اقتراح سيناريوهات محاكاة مختلفة لنمذجة منطقة التغطية وفقًا لحجم المنطقة ، وعدد أجهزة الاستشعار المنتشرة ، ونصف قطر الاستشعار ، ونوع النشر. يتم تنفيذ سيناريوهات المحاكاة لتحليل الاختبار وتقييم العلاقة بين منطقة التغطية ومعلمات ومقاييس الشبكة الأخرى للحالات المقترحة المختلفة. يتم إنشاء نماذج الانحدار التحليلي لهذه الحالات المقترحة المختلفة. هذه النماذج مناسبة للاستخدام بدقة دون الحاجة إلى تكرار برامج المحاكاة أو تنفيذها أو إعادة تشغيلها. تظهر المعادلات التحليلية المقترحة نتائج دقيقة أثناء التدريب وأمثلة الاختبار. تعتمد منطقة التغطية على نصف قطر الاستشعار ، وعدد أجهزة الاستشعار المنتشرة ، ونهج النشر (عشوائي أو محدد مسبقًا) ، والمنطقة المنتشرة. يتم محاكاة تأثير هذه المعلمات على منطقة التغطية واختبارها في سيناريوهات مختلفة. يتم إنشاء واختبار منحنيات الانحدار المختلفة ومعادلاتها. يمكن استخدام هذه المعادلات بدقة لتقدير قيم التغطية.

جمهورية العراق

وزارة التعليم العالي والبحث العلمي

جـامعــة بـابــل

كلية تكنولوجيا المعلومات ـ قسم الشبكات

نمذجة منطقة التغطية لشبكة المستشعر اللاسلكية على اساس الانحدار

رسالة مقدمة إلى

مجلس كلية تكنولوجيا المعلومات - جامعة بابل كجزء من متطلبات

نيل درجة الماجستير في تكنولوجيا المعلومات / الشبكات

من قبل

زينب قاسم محمد علي هداوي

بإشــراف

أ . د . سعد طالب حسون الجبوري

2022　　　　1444