ENERGY EFFICIENT DATA COLLECTION SCHEME USING RENDEZVOUS POINTS AND MOBILE ACTOR IN WIRELESS SENSOR NETWORKS

by

ABDULLAH MOHAMMED ALOMARI

Submitted in partial fulfilment of the requirements
for the degree of Master of Science

at

Dalhousie University
Halifax, Nova Scotia
August 2012

© Copyright by Abdullah Mohammed Alomari, 2012

DALHOUSIE UNIVERSITY

DEPARTMENT OF ENGINEERING MATHEMATICS AND INTERNETWORKING

The undersigned hereby certify that they have read and recommend to the Faculty of Graduate Studies for acceptance a thesis entitled "ENERGY EFFICIENT DATA COLLECTION SCHEME USING RENDEZVOUS POINTS AND MOBILE ACTOR IN WIRELESS SENSOR NETWORKS" by Abdullah Mohammed Alomari in partial fulfilment of the requirements for the degree of Master of Science.

Dated: August 17, 2012

Supervisor: ______________________________

Readers: ______________________________

____

DALHOUSIE UNIVERSITY

DATE: August 17, 2012

AUTHOR: Abdullah Mohammed Alomari

TITLE: Energy Efficient Data Collection Scheme Using Rendezvous Points and Mobile Actor in Wireless Sensor Networks.

DEPARTMENT OR SCHOOL: Department of Engineering Mathematics and Internetworking

DEGREE: MSc CONVOCATION: October YEAR: 2012

Permission is herewith granted to Dalhousie University to circulate and to have copied for non-commercial purposes, at its discretion, the above title upon the request of individuals or institutions. I understand that my thesis will be electronically available to the public.

The author reserves other publication rights, and neither the thesis nor extensive extracts from it may be printed or otherwise reproduced without the author's written permission.

The author attests that permission has been obtained for the use of any copyrighted material appearing in the thesis (other than the brief excerpts requiring only proper acknowledgement in scholarly writing), and that all such use is clearly acknowledged.

Signature of Author

DEDICATION

To who…

Taught me that the ambition should not have an end, my dear parents.

Patted my shoulder to say: I trust you, my beloved wife.

Inspire my life, my lovely children, Ahmed and Mohammed.

Stand always beside me, my wonderful sisters and brothers.

To all of you, thanks for being in my life.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

A Wireless Sensor Network (WSN) is a network composed of a large number of nodes that sense, collect, transmit, and deliver data to where it is needed. Considering the variety of applications, the varied efficiency of WSNs in different environments, and their ability to interact with its surrounding, there are still many challenges to be met and problems to be solved. Overcoming these challenges requires a protocol that is tasked with providing and designing a system that is highly efficient, thus saving energy. In WSNs with Mobile-Actors, the task is first to find an effective way to decrease the length of the tour that the M-Actor follows for data gathering. Nonetheless, this short length should be with a guarantee to access all nodes in the networks to collect the sensory data. In this thesis, we propose a protocol that contributes to reducing energy consumption in WSNs by decreasing the M-Actor path and by using Rendezvous Points (RPs) that are distributed around the network. In addition, the proposed protocol increases the network lifetime by consuming less energy in comparison with a similar protocol and offers reasonable spending time for data collection. All of that with guarantee of offering an access for all nodes inside the network to exchange their data with the M-Actors by the suggested RP algorithm. One or more nodes can be represented by a single RP that provide connectivity to all nodes in it wireless range. In case where more M-Actors are used, less time is required for traversing the network for data gathering. It is shown in this research the tour time can be reduced significantly by using more than one M-Actors.

LIST OF ABBREVIATIONS USED

BS	Base Station
BE	Back-off Exponent
CCA	Clear Channel Assessment
CH	Cluster Head
KAT	K means and TSP algorithm
LAN	Local Area Networks
LEACH	Low-Energy Adaptive Clustering Hierarchy
M-Actor	Mobile Actor
MPDG	Minimum-Path Data-Gathering Problem
MRTA	Multi-Robot Task Allocation
MST	Minimum Spanning Tree
MULE	Mobile Ubiquitous LAN Extensions
QoS	Quality of Service
RP	Rendezvous Point
SHDGP	Single- Hop Data Gathering Problem
SPAT	Set Packing Algorithm and TSP
SPIN	Sensor Protocols for Information via Negotiation
TDMA	Time Division Multiple Access
TSP	Travelling Salesman Problem
TSP_GA	Traveling Salesman Problem based on Genetic Algorithm
WSN	Wireless Sensor Network

ACKNOWLEDGEMENTS

I would like to offer thanks first and foremost to the Almighty God, Allah, who has been my strength all my life.

I would like to express my sincere gratitude to my supervisor, Dr. Nauman Aslam, for his help, support, guidance, valuable advice, and friendship. Dr. Aslam is an example of the sincere person who provides all of his knowledge to his student. Without his inspirational guidance, encouragements, selfless help, I would never acquire this achievement. My special thanks go to the committee members, Dr. William Phillips and Dr. Jack Ilow for spending time for reading my thesis, and providing me with the valuable comments.

Many friends were helpful to me during my study. Many thanks to Sayed Saad, who stood with me on learning Matlab, and for his kindness and generosity to help whenever I needed. I am glad to know him. Thanks to Mohammed Baljoun for his interesting discussion during our study in the field of WSNs. My deep gratitude to my nephew, Hassan Alomari, who has helped me during the stressful times. My great appreciation to the person who stood with me since I was a beginner in the field of computer science, assisted me to choose my major, and encouraged me during all of my studies, my brother-in-law, Ahmed H. Alomari.

Finally and most importantly, none of this would be possible without my family' love and endless support. I have no words to express my gratitude to my wonderful sisters and brothers. I will be in debt for the rest of my life to them for their unlimited support, and generous love. My wife, Fatimah, was the change in my life. I am appreciative to her for every single moment of love, care, patience, and never unfailing faith in me. God's gift, my twin kids, Ahmed and Mohammed who I apologize to them for my absence during their first year. You have been my source of inspiration during this work. I am very thankful to God for having you. Last but not least, I would like to give my heartfelt acknowledgment to my beloved parents. A big kiss on their forehead for their prayer, wisdom, love, and care. I will love and appreciate you forever. I pray to God to save, protect, and bless all of you.

CHAPTER 1 INTRODUCTION

1.1 INTRODUCTION

A Wireless Sensor Network (WSN) is formed from a large number of tiny nodes deployed in a particular area to sense, monitor, and measure certain events via wireless communication. Each node has a microprocessor, a radio chip and a power source. The specifications of these nodes vary, depending on the requirements and the application. Typically, the nodes are low-cost, low-power, limited in terms of memory, and programmable. Such specifications enable the WSN to be applied in many fields and environments, and to work well wherever it is used. However, at the same time, WSNs pose a variety of challenges and difficulties. The typical WSN consists of two main components: sensors and a sink. Essentially, the sensors' task is to sense and collect the desired data and then send it to where it is needed. The sink, or base station (BS), is the place where the gathered data is received and then delivered to the user, whether directly, through the Internet, or through another gateway [1, 2]. In some cases, a third party can act as a medium between the sensors and the sink by carrying the data, buffering it, and then delivering it to the BS [3]. This third party can be either a mobile element or another sensor node. The wireless capability of sensors allows them to cache an event simultaneously even if they are at considerable distance from that event.

Building a suitable WSN requires a consideration of many factors that may help the sensor to perform better. One factor is the deployment form of the nodes. Two types of forms are normally used: well-planned deployment and random deployment [4]. With well-planned deployment, studying the network's requirements and applications can help to achieve a high level of performance. In this case, the nodes mainly remain fixed and their location is known in advance. When the nodes are distributed randomly, deeper analysis is required to ensure that the network will work well. Issues such as connectivity, coverage, security, quality of service, and localization are more likely to be discussed in relation to the random deployment of nodes [4].

1.2 Applications

The simplicity of the WSN enables it to be applied in different environments and for various applications with ease. Furthermore, the flexibility of a WSN could enable important advances in terms of monitoring, acting, and protecting. Here are some examples of these possible advances:

1.2.1 Security and Military applications

A WSN can help to provide a clear information in terms of battlefield surveillance, or battlefield damage assessment [26]. Moreover, deploying sensors in an area that contains nuclear or chemical activities would help provide a secure and safe environment. Gas leak detectors can be used, for instance, to detect possible danger [5].

1.2.2 Disaster relief applications

This is one of ways in which a WSN is most useful. WSNs are used in places where it is difficult or impossible for humans to be. If quick and accurate information is required to respond effectively to a disaster, a WSN could help convey that information. Regarding forest fires, for example, a WSN can help the firefighters to evaluate and analyze the situation.

1.2.3 Medicine and Healthcare

Dispensing wired devices in intensive care units and attaching wireless sensors to patients' bodies or placing them in their rooms can help by improving the level of organization and with providing better service [35]. Monitoring visitors and preventing persons from entering certain departments or facilities that are off-limits are other possible usages for WSNs in healthcare facilities.

1.2.4 Facility Management

Restricting access to certain areas inside a facility can also be done using a WSN. Key-less entrance is another application in this regard, as well as monitoring the temperature, ventilation, or air-conditioning within a facility [5].

1.2.5 Other applications

WSNs can also be used in other areas, such as agriculture, roads, and the environment.

1.3 General Metrics and Challenges

Considering the wide range of applications for WSNs, some performance metrics are more important than others. The selection of performance metrics depends on the user's needs and the application. For example, in natural disasters, the least amount of delay is needed to deliver the sensed data, with no regard for the other metrics [5].

In fact, in some cases, focusing on one performance metric may enhance another and increase its efficiency. For instance, guaranteeing good coverage and connectivity for the network may increase its energy efficiency [5]. On the other hand, if a high quality of service is wanted, that often consumes more energy, which means decreasing the energy efficiency. Next, we list some of the performance metrics of WSNs.

1.3.1 Power Factors and Lifetime

Sensors are expected to gather, process, buffer, and transmit sensed data; thus, more energy would be consumed in these processes. Since most sensors have a limited power source, such as small batteries, this issue becomes one of the substantial matters in WSNs. The first priority in most WSN applications is to adapt the power source to feed the sensors inside the network as much as possible.

1.3.2 Node Costs

Although considering the node cost as a performance metric is a debatable, it is a fact that the node cost may affect the WSN indirectly especially in applications where a limited budget amount is available. Wireless sensors present suitable solutions to many demands. While they are small in size and have a limited communication range, the probability of needing a large number of nodes is high, which subsequently increases the network cost. Another concern is the cost of maintaining and replacing the power source periodically. Depending on the application needs, the node cost can work beside the other metrics to determine the proper nodes. For example, in the healthcare applications, high quality of sensory data is required which demands high efficient- components, which lead to increase the node cost. In high-density application, low cost is required to have more sensors to cover the monitoring area.

1.3.3 Scalability

Some networks have a limited number of sensors. However, numerous WSNs consist of a large collection of sensors. Thus, the network’s ability to be enlarged to handle more nodes without influencing its performance is an important factor.

1.3.4 Coverage and Connectivity

High node density in networks may help them remain connected. However, this does not prevent the breaking of connectivity for several reasons, such as environmental factors or factors related to the communication range or energy required. Hence, this gives an image about the difficulty of coverage the field that is monitored.

1.3.5 Quality of Service

In some cases in WSNs, the accuracy of sensed data is more important than other metrics. The scalability between the accuracy of data and the energy consumed is required for the network to work as expected.

1.4 Mobility in Wireless Sensor Networks

The traditional WSNs employ methods that use fixed nodes to collect data. However, they consume a great deal of energy in comparison to those WSNs that use mobile nodes [16]. Integrating mobile collectors into WSNs not only increases the efficiency, but it also provides higher quality and a longer lifetime for the involved nodes and the entire network. Choosing the proper method to collect data by mobile collectors becomes a challenge due to their mobility and their limitations in terms of energy and range. Recent research indicates that the mobility adds many advantages to WSNs, such as providing a high security connection, saving energy, and improving the network coverage [6, 7]. Moreover, other research has presented other benefits, including, but not limited to, increasing network lifetime [8].

1.5 Problem Statement

By leveraging mobility, significant gains are possible regarding WSNs. Many studies show that mobility in WSNs can improve the network performance in various ways such as lifetime, energy consumption, latency etc. However, introducing mobility in WSNs presents many challenges that need to be solved. Defining a reasonable technique for a mobile node to gather data is a difficult task that many researchers have faced. Finding an optimal set of data collection points is important in developing a data gathering technique. On the other hand, energy consumption is one of the most significant concerns in WSNs. Since nodes have to be ready for receiving and transmitting most of the time, a significant amount of energy will be consumed during the node's operations. Most of the recent research that considers the energy issue relies on one of two techniques for delivering data: either using multi-hop communication from source to destination or using a clustering technique that allows each node to handle some of its neighbors' tasks. In this research, we propose an energy efficient technique for data gathering in WSNs using a mobile actor, or M-Actor, which is a mobile collector that is working to collect sensed data from the sensors and delivers that data to the base station (BS). The proposed technique relies on computing a set of rendezvous points (RPs) for data collection from sensor nodes. The computation of RPs is motivated by the fact that an M-actor only needs to be within the radio transmission range of a sensor node. Visiting the location of the sensor node is not necessary. Hence, one RP may serve as the data collection point for many sensor nodes. However, the computation of RPs requires prior knowledge of the location and topology of sensor nodes. We developed a simple heuristic that allows the M-actor to build a topological map of the network using the information exchanged through *Hello* messages in the network discovery phase. Once the RPs are computed, the information is fed to a GA-based travelling salesman problem (TSP) algorithm to find the optimal tour.

1.6 THESIS OUTLINE

In this section, a general overview of a WSN, including its applications, metrics and challenges, and the issues of mobility and data collection, was presented.

Before we go further, it is a chance to mention that a part of this thesis had been presented and published as part of the IEEE International Conference on Communication ICC 2012 that had been held in Ottawa [9].

The rest of this thesis is organized as follow:

Chapter 2: Background and Related Work

In this chapter, a literature survey of some of the background and related works is presented. Different aspects of data collection techniques and mobility in WSNs are examined with more details, and an overview of the communication mechanism in WSNs. Furthermore, the chapter will offer a number of routing techniques that were suggested in WSNs. Mathematical path determination will be outlined with figures. At the end of this chapter, an overview of several protocols will be explored to provide a clarification about the problem.

Chapter 3: Energy Efficient Data Collection Scheme Using Rendezvous Points and Mobile Actor in Wireless Sensor Networks

Chapter 3 presents our proposed scheme for data gathering using rendezvous points and mobile actors. It includes the suggested system model and assumptions in addition to the clustering pattern. Moreover, it provides a description of the energy model that is used to evaluate the proposed algorithm and scheme.

Chapter 4: Simulation and Results

Chapter 4 presents the methods and the simulation of our protocol and K means and TSP (KAT) protocol [29], followed by a discussion and evaluation of the results and a comparison of the protocols.

Chapter 5: Conclusion and Future Work

Chapter 5 summarizes the thesis and concludes with some possibilities for further research in the future.

CHAPTER 2 BACKGROUND AND RELATED WORK

2.1 Introduction

This chapter provides a background to help the reader understand the following chapters. More details and highlighted points concerning various aspects of WSNs are presented. Data collection and mobility in WSNs are discussed, outlining the communication mechanism, routing techniques categories, the kinds of mobility in WSNs and their advantages, all with examples and brief explanations. Moreover, in this chapter, some mathematical tour optimization methods are shown with their uses and some current protocols that follow these methods. In the last section of this chapter, a survey on some recent related protocols is presented.

2.2 Data collection and Mobility in Wireless Sensor Networks

Data collection is the main task of using wireless sensor networks. This task can be done by different ways, and depends on different methods and techniques. In this section, a revision of data collection and mobility will be shown from several aspects.

2.2.1 Communication Mechanism in WSNs

A sensor node may deliver its sensory data to a destination either directly or indirectly by using a single hop or multi-hop sensor node. The strength of radio transmission power may be modified in order to route packets to the final destination either through a larger number of smaller hops or a smaller number of larger hops. One of these two methodologies is selected based on energy efficiency, which is discovered by examining the power consumption of the different operating modes and calculating the path loss between a sender and a receiver [37].

In order to create a WSN to cover a large area using relay nodes, multi-hop routing must be applied. However, if mobile collectors are applied in the network, as done in our research, multi-hop routing will not be necessary and single hop routing can achieve network connectivity. In Figure 2.1, both methods are illustrated to show the difference between them. On the left side, where single-hop communication is applied, the sensors in each cluster head (CH) are linked directly to it, and at the same time the CH itself can reach the BS with no need to communicate with another CH to relay its data. On the right side, a multi-hop communication is set up, and it appears that some sensors are far away from the CH; thus, other sensors are relied on to reach the CH. Nonetheless, some CHs would be far away as well, so the data would be sent to the closest CH within range.

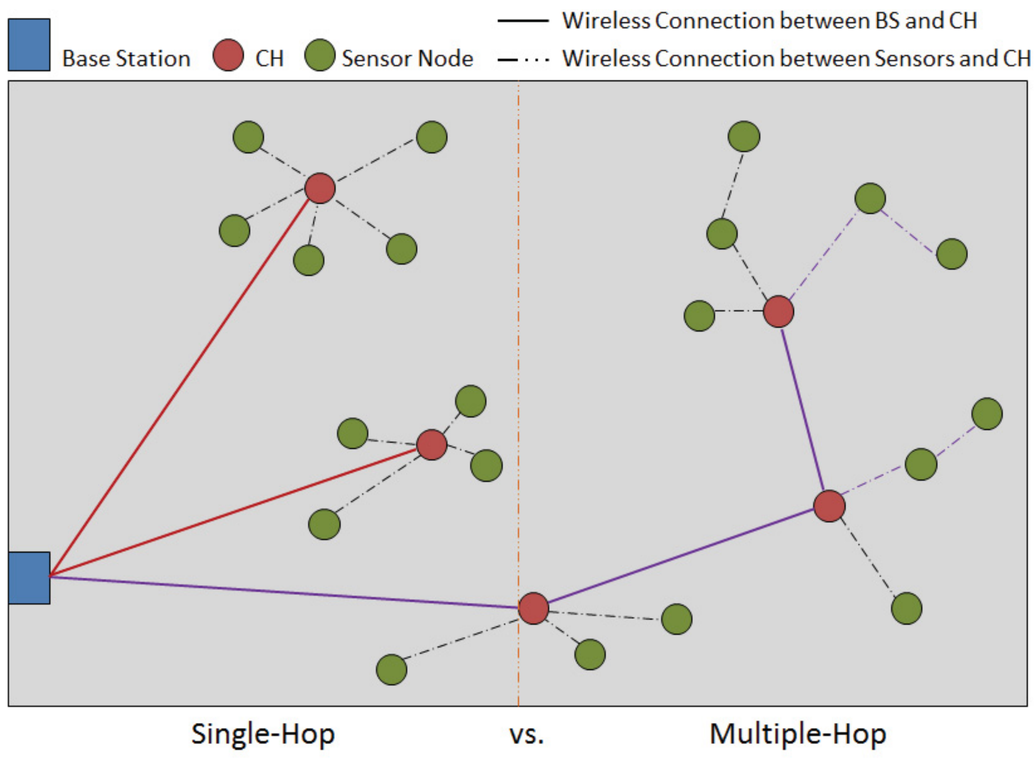


Figure 2.1 The differences between Single-hop and Multi-hop communication scheme.

2.2.2 Data Collections and Routing Techniques Categories in WSNs

In terms of network structure, generally, the methodologies of data collection used in WSNs can be classified into three main schemes, flat, hierarchical, and location-based [10]. The aim of these schemes is not only to gather data, but to do so while also improving the performance and other metrics as well, or at least without having a negative influence.

Here, we provide a short summary of each of the three data collection schemes, with examples for each type and the metric that each focuses on.

2.2.2.1 Flat Schemes

In this category, all nodes inside the network are equal and share the same roles and tasks. To determine the optimal routing path for data exchange, the source node keeps sending query messages until they reach the destination node. Then, both the source and destination decide which path is the best.

In the flat scheme, receiving the data is more important than knowing who sent it, and for that reason there is normally no need to assign an ID to each node.

Many protocols have been designed to follow this scheme for different reasons. Some examples are SPIN [11], Directed Diffusion, [12], and Rumor routing [13]. Here, we chose one of them, SPIN, to discuss.

- SPIN

In Sensor Protocols for Information via Negotiation (SPIN), each sensor that intends to send data to another needs to undergo a negotiation process to determine the optimal routing path for sending that data. In addition, all sensors have to maintain their energy modes to conserve energy during the network operation. SPIN works with three types of messages and all of its data exchanges depend on them. If a node has data to transmit, it sends an advertisement message to all nodes within its range, to inform them about its intent to send a message. The ready sensors will reply to it by sending back a request message for it to transmit data. Then, the sender will start transmitting a part of the whole data, called meta-data, which includes the size of the actual data, destination, and other data depends on the application specifications. The receivers will check the provided information, and repeat the same procedure as in the previous instance until the destination node is reached. Upon this occurring, the best routing path will then be created. Figure 2.2 shows the different operations and message stages in SPIN.

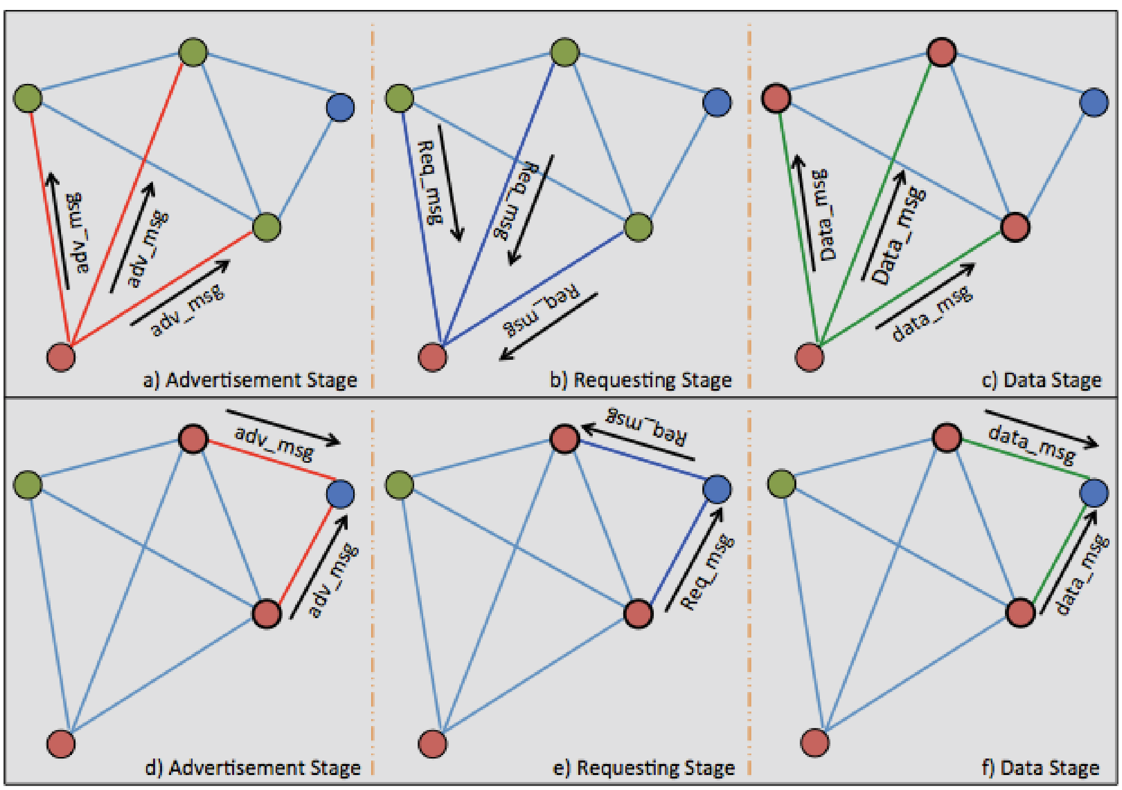


Figure 2.2 SPIN's operations and messages' stages [11].

2.2.2.2 Hierarchical Schemes

The hierarchical protocols derived their name from their task and the method they employ to collect and route data. In such protocols, each network is divided into small subnetworks and assigns a particular node to work as an administrator for that sub-network, thus handling the major function of receiving and routing the data. This type of protocol has brought considerable changes to WSN energy consumption. Next, we present two hierarchical techniques that have been demonstrated to be successful, while also discussing an example of each technique.

A- Clustering Technique

The network is sectioned into a set of subnetworks, each of which is called a cluster. In each cluster, one node represents the others and is called the cluster head (CH). Since the CH is responsible for all of the events inside its cluster, the probability of it to being down is high, and when that happens, its task will be moved to another node which will then become the CH, depending on its essential algorithm. One of the notable protocols that follows this technique is LEACH [14].

- LEACH

Since LEACH was first introduced, a huge development in WSNs has occurred. The LEACH (Low-energy Adaptive Clustering Hierarchy) protocol, presented by Heinzelman et al., is a well-known protocol that follows the clustering scheme. LEACH assumes that the nodes inside the network are fixed, and that the base station is far from them. All nodes in the network are homogenous and energy constrained. However, in LEACH protocol, some nodes consume more energy than others, which leads, over time, to some nodes being disconnected from their network. The main idea behind the LEACH protocol is to divide the network into a number of sub-networks called clusters. Each cluster consists of a fixed number of sensors, which, by voting, elect a representative node, the CH. Each sensor inside the cluster should send its data to the CH, which is responsible for delivering data to the sink. As mentioned above, this process can lead the CH to handle the most significant tasks inside the cluster, which means the CH loses its energy faster than the other nodes. LEACH is a Time Division Multiple Access (TDMA) -based protocol that provides time slots for each node to exchange data between the member nodes and their CH. Moreover, the TDMA measure the probability of each node becoming a CH, depending on its energy in a round algorithm. The rounds are the cornerstones of the LEACH protocol and algorithm. Each round is composed generally of two phases: setup phase and a steady-state phase. During the setup phase, the cluster is created and the CH is elected. The second phase is responsible for distributing the time slots among the network nodes. Figure 2.3 illustrates the LEACH operation phases and protocol [5].

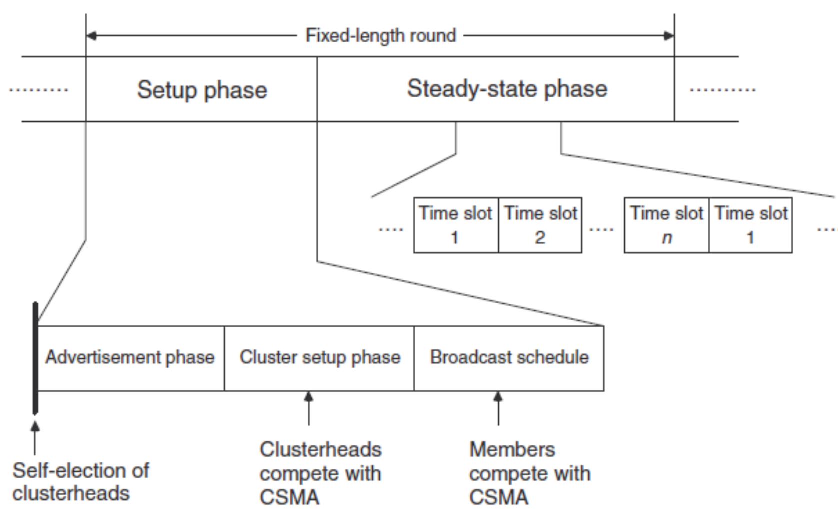


Figure 2.3 Operation Phases in LEACH [5]

B- Rendezvous Points Technique

This technique differs from the previous one insofar as there is no node to handle the cluster operations. Simply put, the network consists of a set of sub-networks and each sub-network has a centroid point that, when the mobile element stops at it, it can reach all nodes inside the cluster and gather data from them. In the case of multi-hop communication, the mobile element is not required to reach all nodes, but it has to guarantee that each node belonging to the same subnetwork has the ability to access it through another node. Since the mobile sink, or actor, does not need to travel to all of the sensors, but only moves to the RPs, a significant decrease in the data gathering delay will be achieved. Xing *et al*. presented a proposal of this type that we chose as an example of this technique [15].

- Rendezvous Design Algorithms for Wireless Sensor Networks with a Mobile Base Station [15]

The authors of this protocol assumed a mobile sink that has to move around a network to collect data. Two algorithms are produced: one for clustering the network into a subset of smaller networks and to assign an RP for each cluster, and the other to provide an essential method for the sink to optimize its travel around the RPs. The protocol in the two phases uses an integration of the Minimum Spanning Tree (MST) in the first one, and TSP in the second phase to treat the problem of having a large number of nodes for data gathering, thus decreasing the mobile sink path.

Details concerning the TSP and MST are shown in section 2.3 of this chapter.

2.2.2.3 Location-based Schemes

Sensors used in this technique depend on their location to communicate with each other. The distance between sensors can be measured either by estimating the signal strength or by exchanging the information when the sensors are able to determine their location. Most of the protocols that follow this scheme aim to achieve an improvement in energy efficiency. Changing the power modes and the roles between the sensors is popular when using the location-based routing structure.

2.2.3 Influences of Mobility in WSNs

As shown above in section 1.4, mobility enhances many factors in WSNs and may help to cover some weak points in a network. We list some of these improvements below. Then, we present the different types of mobility in WSNs. Moreover, information is introduced concerning some algorithms and protocols that have been suggested to work with these types of mobility.

Mobility in WSNs can improve the network in terms of:

- **Increasing Network Lifetime**

In the ordinary WSN, where sensors are static networks components, data delivery faces many obstacles, such as the distance between sensors and the BS, or even between the sensors themselves. A sensor's wireless range is limited, and in large, sparse networks where sensors cannot reach each other easily, multi-hop method of communication must be used. One of many concerns with multi-hop communication, however, is that some nodes handle more than their assigned task. This problem leads to a situation where these nodes consume more energy, which can result in battery failure. Mobility can fix this problem by reducing the need for multi-hop communication by retrieving collected data from static nodes and delivering it to the sink. This decreases the load on some sensors and allows them to save their energy, which increases their lifetime.

- **Reliability**

Since the M-actor moves to retrieve the data from its source frequently, the connection reliability may increase due to the decrease in the packet loss rate. In a static WSN, the packet loss rate can be higher because of the nature of the connection required for multi-hop communication to deliver data.

- **Connectivity**

In sparsely deployed WSNs where the nodes are located at far distance and disconnected from each other, an M-actor can work as a link between them, making dense networks dispensable.

- **Security**

Mobility can provide a highly secure connection for such networks. Giving authority and privileges to some mobile nodes to reach certain levels of data from sensors, instead of sharing that data through the whole network, is a necessity in some scenarios where data must be protected.

2.2.4 Types of Mobility in WSNs

The mobility in WSNs can be represented in three ways:

A- Sink Mobility

The sink, or BS, is the last station for data delivery. In cases where a mobile sink exists, data will normally be gathered using one of two methods: either by moving around the networks and visiting each sensor to retrieve its data, or by moving through certain points distributed around the network or some relaying nodes [3]. More details and examples of protocols using this type are included in [16], and [17]. Figure 2.4 illustrates the different methods.

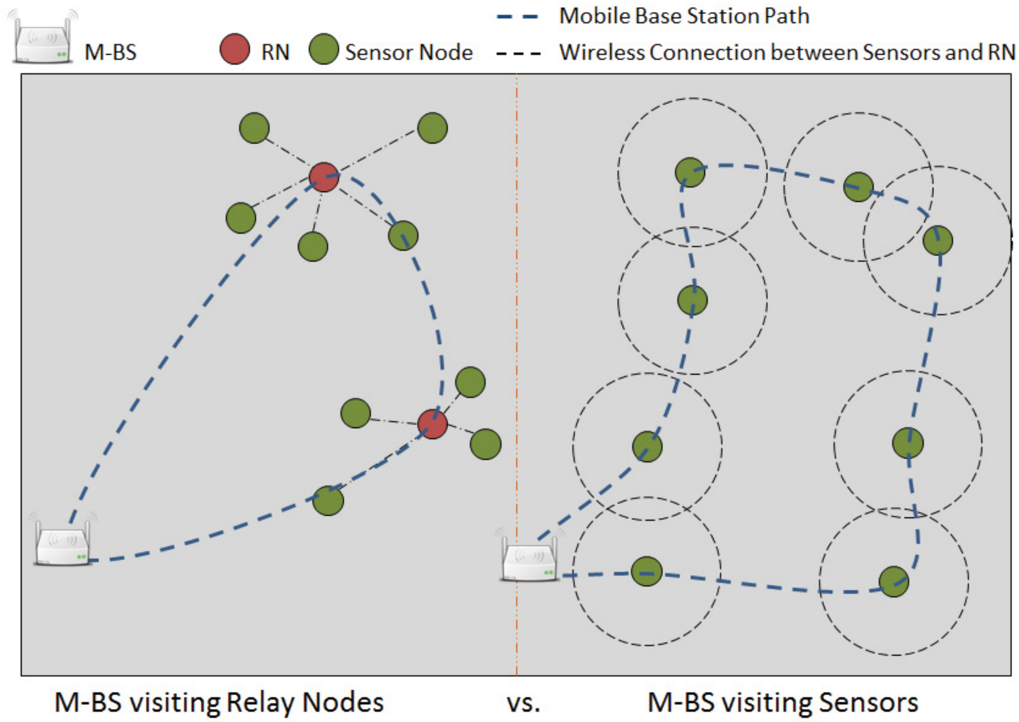


Figure 2.4 A mobile sink visits relay nodes, versus mobile sink visits sensors to gather data

B- Sensors Mobility

In this type of mobility, the sensor has the ability to move freely because it is attached to a mobile element to sense and collect data and react to that data. Typical uses involve robots and actuators. More details are available in reference [18], and [19].

C- Element & Actors Mobility

This type of mobility works as an intermediary between the two previously discussed types. It is neither a data source nor a data destination [3]. Its main task is to retrieve the data from the place where it was sensed and then deliver it to where it is needed, the sink. Generally, this type of mobility has a higher level of performance and more abilities than sensors have. In many cases, this type of mobility is essential to the network working properly. The absence of this type of mobility may lead the entire network to disconnect.

2.3 Tour Optimization in Wireless Sensor Networks

Obtaining an optimal route for mobile collection in a WSN can positively affect many metrics. One of these metrics is the tour length. Many mathematical solutions were investigated that show useful improvement for enhancing the operation of data gathering when using mobile collectors. Here, we show two well-known problems that are used to optimize the path problems.

2.3.1 Travelling Salesman Problem (TSP)

TSP assumes that when an n number of cities are given, and each should be visited by a salesman (one and only once for each city), the challenge is to find the shortest tour that involves the shortest distances at the lowest cost. The TSP is considered an NP-Complete problem, where it is not guaranteed that the optimum solution will be reached. TSP has been used in a great deal of research in different fields and its success has been demonstrated. Replacing the cities with nodes and the salesman with the mobile collector, the problem has been used in many WSN applications and protocols to optimize the tour for gathering data in WSNs.

Generally, TSPs are categorized into two types: symmetrical and asymmetrical [20]. The symmetrical TSPs are the cases where the distance (or cost) from starting point i to end point j is equal to the distance (cost) from j to i. In this type of TSP, when n number of nodes is given, there are always $(n-1)!$ visible solutions. The task is to find which one of these $(n-1)!$ solutions is the shortest in distance unit or least in cost unit. On the other hand, with the second type of TSP, the asymmetrical ones, the distance (cost) between two nodes differs by the direction. Thus, the distance (cost) from i to j is *not equal* to the distance from j to i. In this case, when n number of nodes is given, there will be $\frac{(n-1)!}{2}$ solutions. TSP usually assumes that the salesman can move freely from any city to any city, no matter which city is the starting point [20,21].

Over the course of history and especially in the last few decades, many algorithms have been developed to reach an adequate solution to the TSP.

2.3.2 Minimum Spanning Tree (MST)

A Minimum Spanning Tree (MST) aims to find the minimum total cost of moving around a set of nodes, starting from one point and ending at another one. The MST has been used widely in many applications, and it is also a typical solution for communication network connectivity problems. In WSNs, a MST is used to guarantee the movement of an M-element from one point to another within the network or to help to distribute the nodes effectively in the network. Examples of how MSTs have been used in WSNs are shown in [22], [23], [24]. Here, in Figure 2.5 and 2.6, we show two different algorithms that have been proposed for solving and illustrating an optimal solution in the MST. Figure

2.7 provides an example of a network of five nodes with eight different edges and cost, as well as its MST solution [21].

1. Initialize ***S*** = { }
2. Pick the edge with smallest weight and enter to the MST. Add the vertices in ***S***.
3. For every vertex in ***S***, find the edge with minimum weight connecting ***i*** in ***S*** to ***j*** in ***N-S***. Let ***i-j*** be the minimum weight edge among these. Add vertex ***j*** to the set ***S***. Include edge ***i-j*** in the MST.
4. Repeat step 3 till all vertices are in {***S***}.

Figure 2.5 The Prim's Algorithm [21].

1. Initialize ***n*** edge **= 0** $_{i\text{-}j}$. Arrange edges in increasing (nondecreasing) order of weights.
2. Pick the first edge and ignore it if it forms a loop with the existing edges in the MST. Enter this to the MST if it does not form a loop. ***n*** edge + 1. Remove edge ***i-j*** and Repeat 2.
3. If n edge = ***N***-1 STOP. Else go to Step 2.

Figure 2.6 The Kruskal's Algorithm [21].

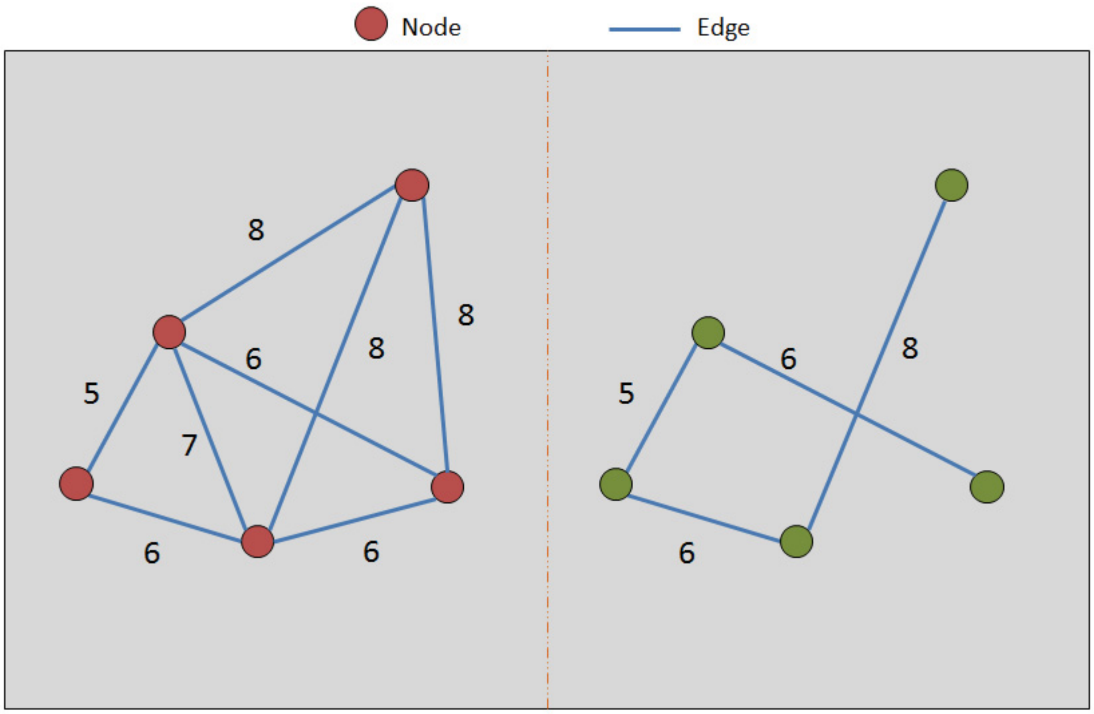


Figure 2.7 An example of MST problem with five nodes and eight edges [21].

Offering a simple explanation of the difference between TSP and MST, Figure 2.8 gives an example of five different nodes linked together with their edges costs. The TSP tour starts from one node, and ends at the same node, while the MST tour goes around all of the nodes no matter where its starting point was or where its ending point will be.

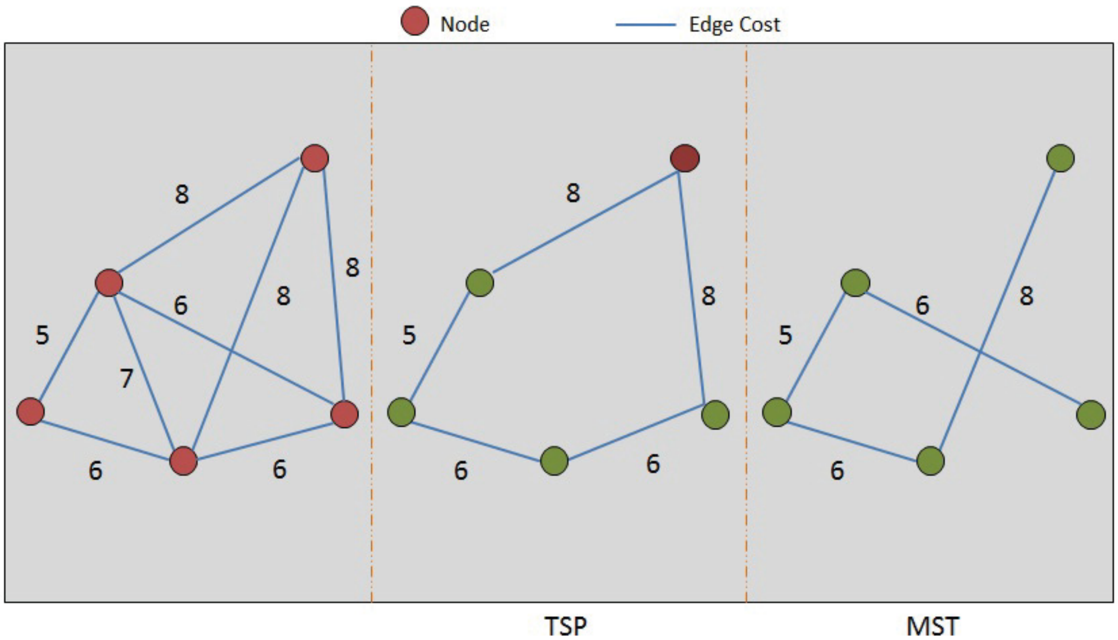


Figure 2.8 The differences between TSP and MST with the same topology.

2.4 Review of Different Mobility Algorithms and Protocols *[9]

One of the earliest schemes, proposed by Shah et al., is Mobile Ubiquitous LAN Extensions (MULEs) [25]. In this research, the network architecture is divided into three tiers for collecting data. The MULE, which is the middle tier, moves around the network to retrieve data from the sensors, which represent the bottom tier, to store that data and then upload it to the top tier, which can be a sink or an access point. MULEs can be vehicles, humans, or even animals, depending on the application. Due to the limitation of the transmission range of the sensors, the MULEs tour increases by the number of sensors or the sensors that are too far to reach. The main drawback of this proposal is in regards to latency. However, the MULEs added a significant improvement to WSNs.

With the Single- Hop Data Gathering Problem (SHDGP) algorithm [26], Ma and Yang propose a new protocol for gathering data in large-scale networks. The aim of this proposal is to find the shortest path possible to traverse the network and collect sensed data. In the SHDGP, the sink is static, while one or more collectors called M-Collectors represent the mobility. The M-Collector starts its tour from the sink and returns to it after the data has been gathered. In the first tour only, the M-Collector travels around the network and stores sensor positions in order to locate them during future tours. This tour is called the exploration tour. In the following tours, when required data is sensed, the M-Collector can collect it by creating stop points, which are called polling points, and then gathering the data from these points. One or more nodes can share the same polling point. The TSP is used as a reference to compute the optimal path for traveling around the points. The main objective of this research is to increase network lifetime. In [27], Fang-Jing Wu et al. propose a new protocol for collecting data in large scale WSNs, Minimum-Path Data-Gathering Problem (MPDG). Due to various factors, the WSN can be separated into multiple subnetworks. These factors include natural obstacles, mountains, etc., or emergencies such as fire, floods etc. The Fang-Jing Wu *et al.* study shows how data can be gathered in an optimal manner using mobile MULE and TSP to compute path length. Table 1 shows a comparison between different protocols that use mobility in WSNs.

Table 1 Comparison between different protocols that use mobility in WSNs

Protocol	Network Size	Static Elements	Mobile Elements	Sensors Location	Topology	# of M-Elements	# of hop	Tour Computing	Performance
MULEs	Large	Sink - Sensors	MULEs	Pre-known	Sparse Networks	Single - Multiple	Single	-	Energy Conservation
MPDG	Large	Sink	MULE	Pre-known	Sub networks	Single	Multi	TSP	Minimizing Path Length
SHDGP	Large-Scale	Sink - Sensors	Collectors	Exploration Tour	Spanning covering tree	Single - Multiple	Single	TSP	Increasing Lifetime
KAT	Large-Scale	Sensors	Sink	Pre-known	Clusters	Single - Multiple	Single	TSP	Energy Conservation
SPAT	Large-Scale	Sensors	Sink	Pre-known	Clusters	Single - Multiple	Single	TSP	Fairness, Connectivity

In the case of the Set Packing Algorithm and TSP (SPAT) [28], the goal is to minimize the data collection tour. The protocol procedure consists of four phases. The first phase is to generate the clusters of sensors. After that, the number of clusters is decreased by eliminating and filtering them. In the third phase, the clusters are reorganized to guarantee that they cover all sensors using a minimum number of clusters. Finally, the TSP is used to computes the path that should be taken. In this protocol, the authors guarantee that all nodes in the network are visited. In the fault-resilient sensing in WSN proposal, the authors introduce a new scheme for collecting sensed data in WSNs. The method, KAT [29], is based on the K-means clustering and TSP algorithm. The idea here is to divide the field into a set of clusters where each cluster can contain a number of sensors. The sensors in this model are fixed, while the sink itself represents the mobility. The role of the sink is to traverse the networks to collect data, buffer the data, and upload it to the access points. The authors argue that their proposed scheme is useful even when some parts of the network are being attacked or lost. The network is assumed to be administrated by a person whose task is to monitor and localize the actual sensors in the field. The One Phase Pull model, which is a version of Directed Diffusion, is adopted. In this model, the sink sends a query to all nodes that can be reached, which they then resend to their neighbors until the source is reached. Then, the source responds to the message along all paths. After that, the optimal path can be selected from among all possible paths. In some cases, such as on a battlefield, the network topology can change at any time, especially when some parts are being attacked. To solve this problem, the authors suggest that the sink be mobile and move randomly around the network with a certain velocity.

The sink moves to central points in clusters depending on the path computed by the TSP. In their evaluation, the authors confirm the effectiveness of their work by creating a simulator to accommodate their idea. This simulator measures the model efficiency using different transmission ranges and different amounts of data. The energy efficiency is expressed in the KAT model as:

$$R\ [\mathrm{KB}] = \frac{\text{Received Bytes by all Mobile Sinks}}{N \times M} \tag{2.1}$$

Where R represents the average received bytes by all services (sinks, sensors), N is the number of sensors, and M is the number of mobile sinks.

Secondly, the average consumption over the number of mobile sinks can be calculated as:

$$C\ [\text{mWHr}] = \frac{\text{Consumed energy by all Sensors}}{M} \tag{2.2}$$

Then, KAT energy efficiency is calculated as:

$$E\ [\text{KB/mWHr}] = \frac{R}{C} = \frac{\text{Received Bytes by all Mobile Sinks}}{\text{Consumed energy by all Sensors} \times N} \tag{2.3}$$

2.5 Conclusion

In this chapter, we provided background information concerning various aspects of WSNs. We began by briefly discussing data collection mechanisms and data gathering techniques with examples of protocols. Then, we discussed the influences on using mobility within WSNs from different perspectives. We then explained the kinds of mobility, and offered some mathematical problems for tour optimization. At the end of the chapter, we explained and criticized some recent protocols that are closely related to our particular area of research.

CHAPTER 3 ENERGY EFFICIENT DATA COLLECTION SCHEME USING RENDEZVOUS POINTS AND MOBILE ACTOR IN WIRELESS SENSOR NETWORKS

3.1 INTRODUCTION

In this chapter, we propose a new technique for gathering data in WSNs using an M-Actor. The proposed technique relies on computing a set of rendezvous points for data collection from sensor nodes. The computation of RPs is motivated by the fact that an M-Actor be only within the radio transmission range of a sensor node. Visiting the location of a sensor node is not necessary. Hence, one RP may serve as the data collection point for many sensor nodes. However, the computation of RPs requires prior knowledge of the location and topology of sensor nodes. We developed a simple heuristic that allows the M-Actor to create a topological map of the network using the information exchanged through 'Hello' messages during the network discovery phase. Once the RPs are computed, the information is fed to a GA-based TSP algorithm in order to discover the optimal tour. Figure 3.1 illustrates the architecture of the proposed framework where the sensor nodes are deployed in a 2D plane. A set of RPs is computed and used by the M-Actor for data gathering.

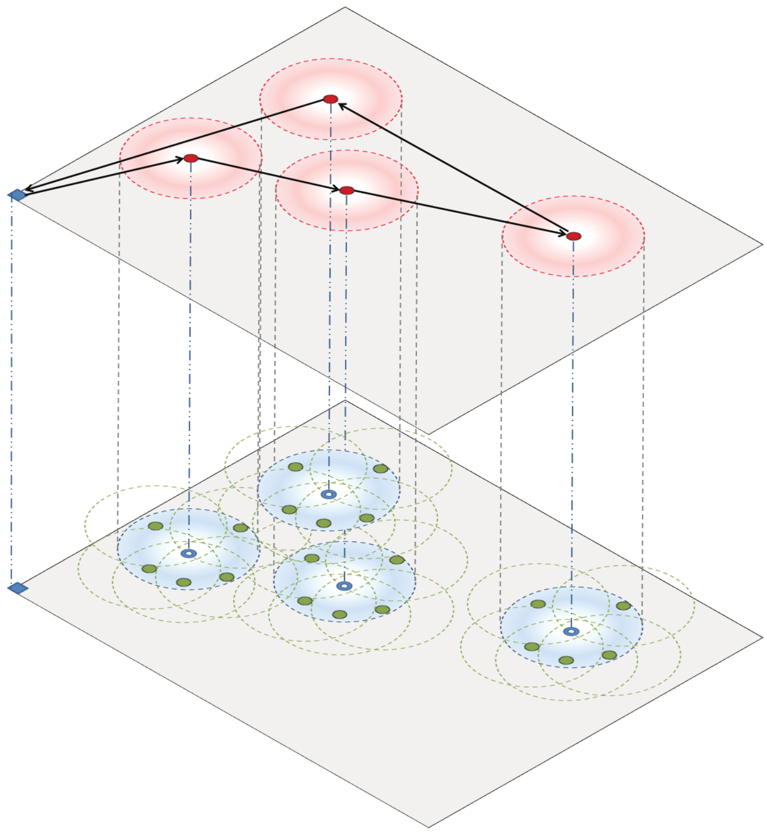


Figure 3.1 Proposed Network Model

3.2 System Model and Assumptions

This section details our system model and the assumptions used for the current framework.

1) Sensors are placed in a 2-dimensional field of a known area *A*. The placement of sensor nodes is assumed to follow a uniform distribution.

2) Sensor nodes remain static and do not move after placement. All nodes are homogeneous in terms of energy, communication and processing capabilities.

3) All nodes can work in different power modes (receive, transmit, idle, sleep) and can change their mode whenever necessary.

4) The M-Actor in not energy constrained and is able to move freely within the deployment area.

5) A polling based scheme is assumed for data collection. Once the mobile collector stops at an RP, it sends a poll message to each node for data collection.

6) The M-Actor is able to move in a straight line from one point to another within the deployment area. For simplicity's sake, we assume that there are no obstacles in the deployment area that would restrict movement in a straight line.

7) Data is delay tolerant and does not require any quality of service guarantees.

8) Sensor nodes are able to determine their location in the deployment area.

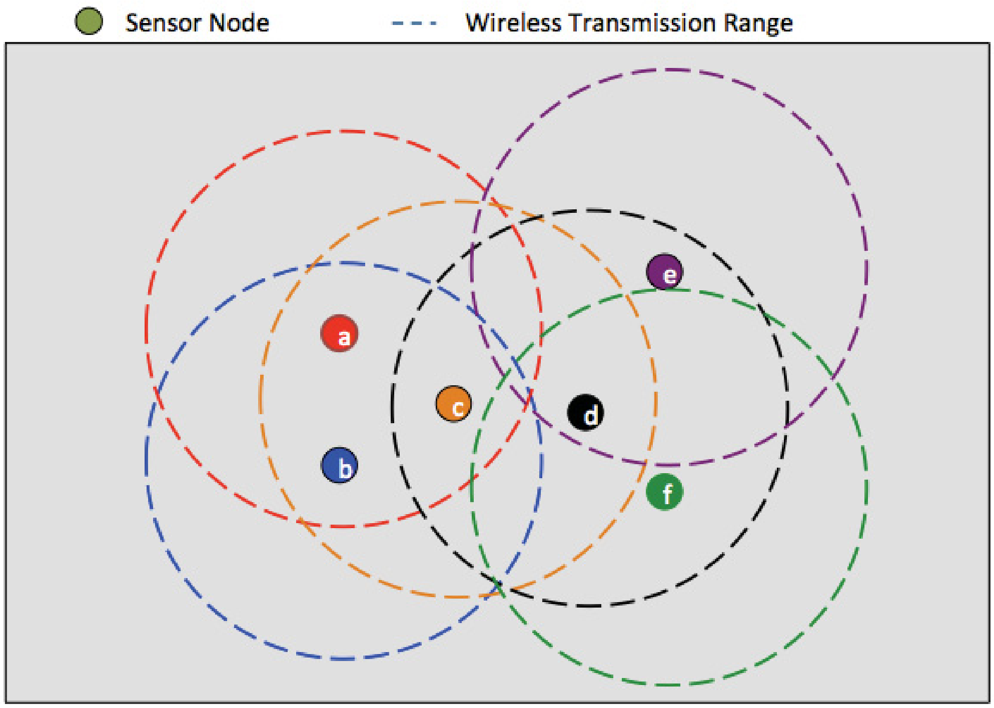
9) The data storage capacity of an M-Actor is large, so it is able to collect and store data in its local buffer during path traversal.

3.3 RP DATA GATHERING SCHEME

This section provides details for finding the shortest path for an M-Actor, which traverses through the deployment area so that each sensor node is able to transmit its collected data to an M-Actor. The proposed scheme consists of the following steps:

3.3.1 Neighbor Discovery

In this step, each node establishes a two-hop neighbor information base. At the network initialization phase, this information can be established by exchanging 'Hello' messages. In each 'Hello' message, a node will advertise its ID, location, energy, and number of neighbors. In order to make sure that each node is able to receive a 'Hello' message from all neighbors, we implement a timer. As long as the timer does not expire, nodes will process any 'Hello' message received. In processing 'Hello' messages, the neighbor field will be examined and the node will update its database. Implementing this timer also ensures that nodes are allowed sufficient time to complete the discovery process.



$$N_a = \{b,c\}, N_b = \{a,c\}, N_c = \{a,b,d\}$$
$$N_d = \{c,e,f\}, N_e = \{d\}, N_f = \{d\}$$

Figure 3.2 Distributed nodes with their transmission range

3.3.2 Building Virtual Clusters

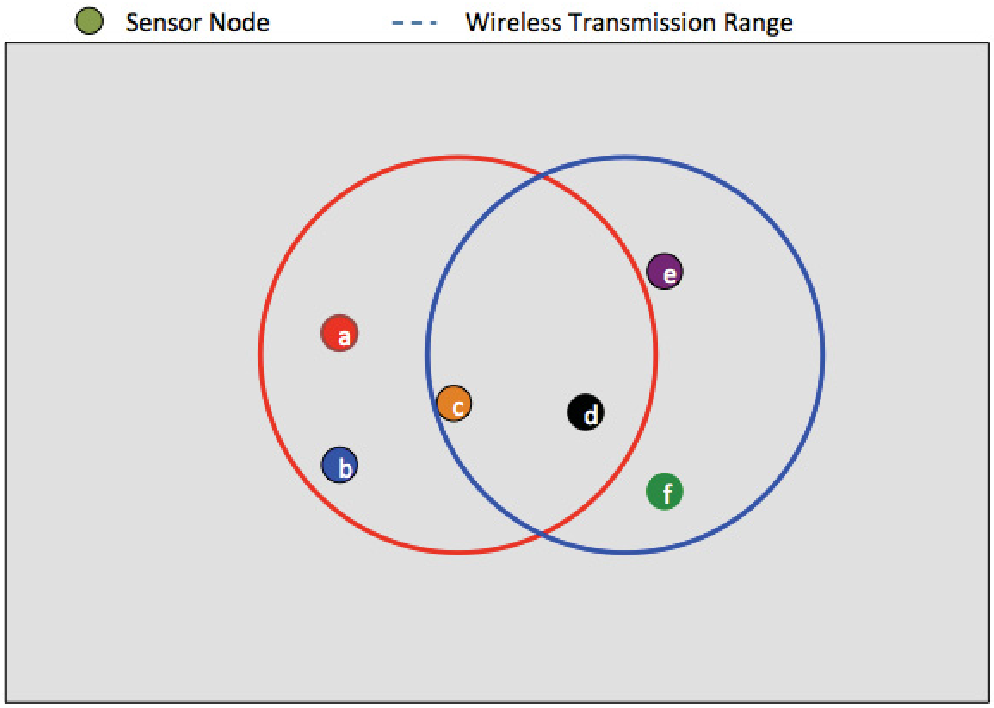
Before an M-Actor starts its tour, it needs to compute RPs for data collection. However, the computation of RPs depends on building a topological map of all sensor nodes, which requires obtaining location, energy and neighborhood information related to each active sensor node in the deployment area. Now if each sensor node were to send its information to the M-Actor, a considerable overhead would be incurred. In order to avoid this problem and gather the topological information, some researchers have suggested sensing area exploration tours prior to data collection [10]. While exploration tours seem like a good starting point, they cannot guarantee that the data collection points cover all sensor nodes in the deployment area. We therefore introduce a new concept, where virtual clusters are formed so that only a subset of nodes relays neighborhood information to the M-Actor. Consider Figure 3.2, which provides a sample scenario where six nodes are deployed in an area A. We say that a sensor node i is a neighbor of j if a wireless link exists between i and j. Let R_{Tx} denote the wireless transmission range of an arbitrary sensor node ni. The neighbor set Ni consists of all nodes that are its radio transmission range R_{Tx}.

$$N_i = \{n_j\}, \forall j, d_{ij} \leq RT_x \tag{3.1}$$

where d_{ij} is the Euclidean distance between i and j. During the 'Hello' message exchange, a sensor node builds its neighbor information base as shown in Figure 3.2. The nodes form virtual clusters after the neighbor discovery process in an autonomous manner as in Figure 3.3.

In our example scenario, node a and b have neighbor sets $\{b, c\}$ and $\{a, c\}$ respectively. Node c has neighbor set $\{a, b, d\}$. When node a and b receive neighbor information from node c, they observe that their ID is listed in the neighbor set of node c; therefore, they decide to relinquish the advertisement of topological information to node c. Now, node d also receives the neighbor set information of node c. Upon inspection of its neighbor set, it observes that nodes f and e are not neighbors of node c. It therefore sends the topological advertisement packet to the M-Actor. The remaining nodes, d and e, refrain from sending the topological advertisements for the same reason as discussed for node a.

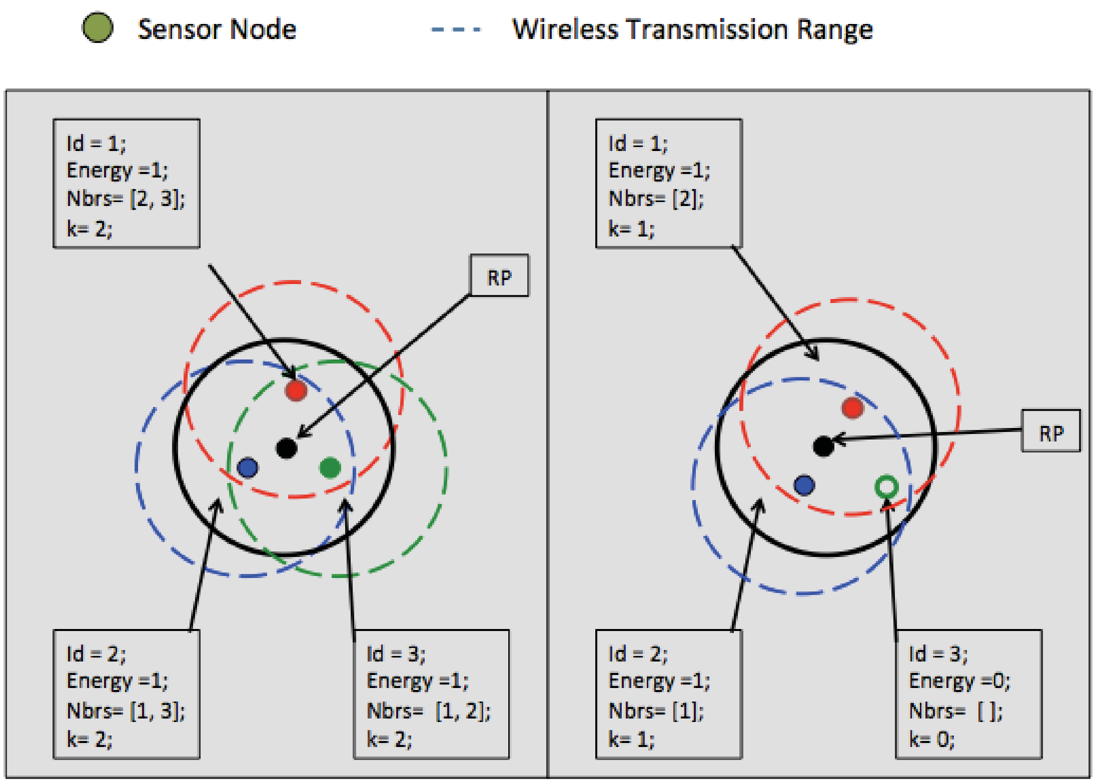
By using this simple scheme, nodes c and d assume the role of virtual cluster heads (CHs) to advertise the topological information to the M-Actor.



$$Virtual\ Cluster\ 1 = \{a, b, c, d\}$$
$$Virtual\ Cluster\ 2 = \{c, d, e, f\}$$

Figure 3.3 Virtual Clusters Formation

In the following tours, and in the case of a node becoming discharged, its neighbours would not receive the advertisement message, and they will thus exclude it from their list and from the calculation. Figure 3.4 gives an example of three different nodes with their ranges and RP. The (a) section represents the topology and the table list at the initial phase when all nodes have full energy and the same number of neigbours. Now, in (b), when a sensor loses its energy, its table will be updated with the new energy; thus, it will not be able to communicate with others. At this stage, the remaining sensors will exclude it from the RP's calculation. A new RP will be formed at the centroid point of the live sensors.



(a) (b)

Figure 3.4 (a) all sensors are alive. (b) a sensor's energy is lost, and a new RP is formed.

3.3.3 Calculation of Rendezvous Points

After the M-Actor receives all topological information advertisements from the virtual clusters, the next step is to calculate the set of RPs. The RP set denoted by $\Re$ should be computed so that each sensor node is covered by at least one RP. Figure 3.5 shows the proposed RP's selection architecture.

Determination of the RPs can be summarized as following these steps:

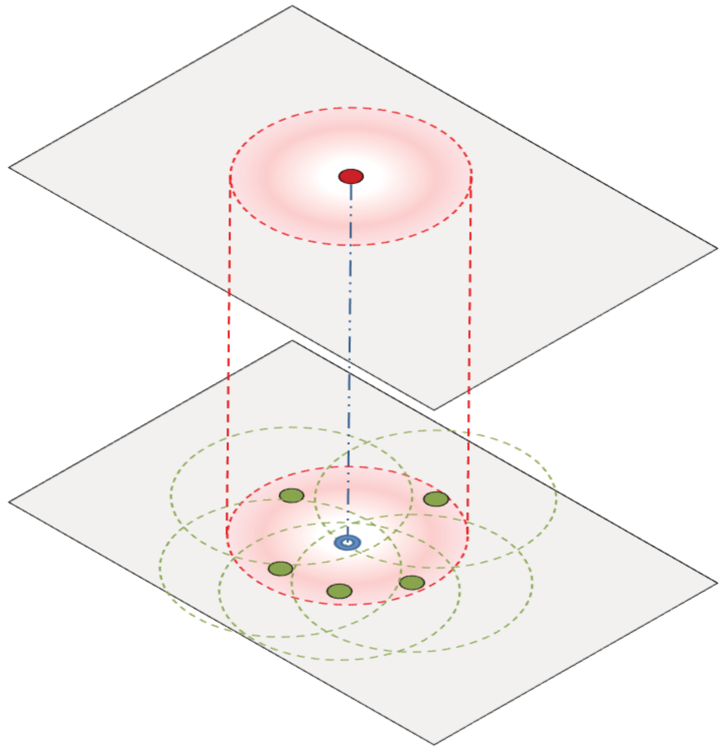


Figure 3.5 RP's Selection architecture

1) Initialize structures in the local storage for $\Re$ and a cover set denoted by C.

Initialize $C, \Re \leftarrow \phi$

2) The M-Actor examines all topological advertisement messages and creates an active nodes list $\mathcal{S}$.

3) Define the stopping criteria as $C == \mathcal{S}$ for the following iterative steps.

4) Select a node $v_i \in \mathcal{V}$ with the largest node degree and highest amount of energy. The degree here represents the number of reported active neighbors. In cases where more than one node has the same degree, break the tie in favor of the node with the lowest ID.

5) For the v_i selected above, examine its neighbor set N_i. For each node listed in N_i, check if it is a member of C. If the result is negative then update C and store the information in temporary list $\mathcal{T}$. This procedure is followed until the end of list N_i.

6) Compute the centroid using the location coordinates of all elements stored in $\mathcal{T}$ using the following expression;

$$(Cx, Cy) = \left(\frac{\sum_{i=1}^{n} x_i}{n}, \frac{\sum_{i=1}^{n} y_i}{n}\right) \tag{3.2}$$

7) Update $(Cx, Cy) \rightarrow \Re$ and set $\mathcal{T} \leftarrow \phi$

8) For any node that has a degree of zero (no neighbors), add its coordinates to $\Re$.

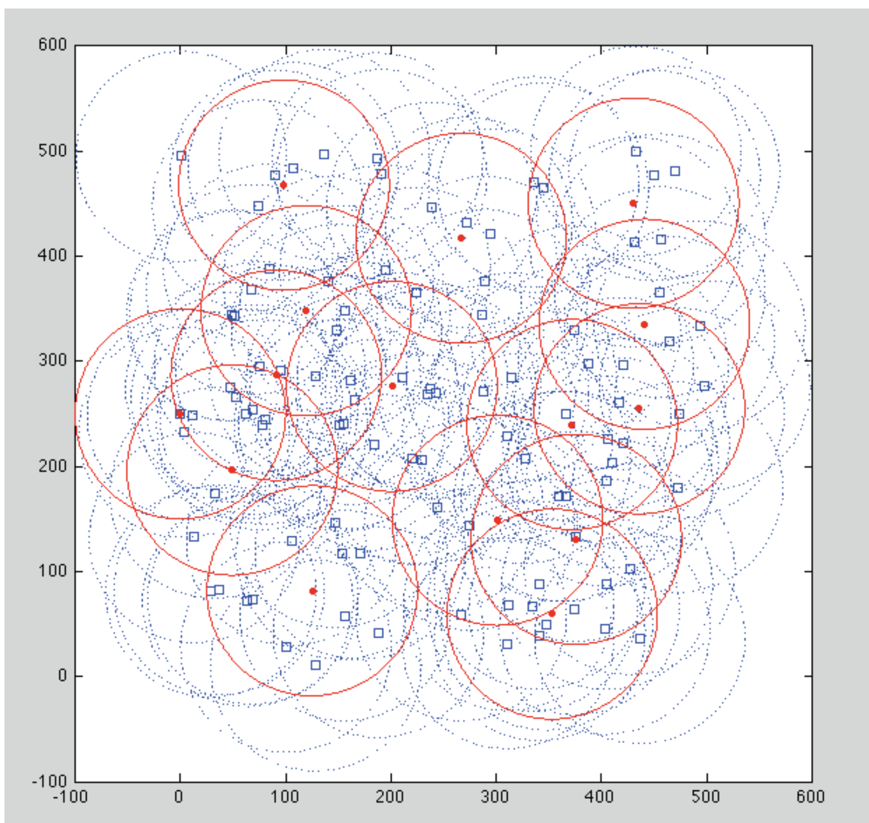


Figure 3.6 The RPs formation among the nodes (example of using 100 nodes)

Figure 3.6 illustrates a network topology of an area of 500 * 500 m contains 100 nodes, which creates 15 clusters and RPs. The blue squares represent the sensors with the circles as their ranges of communication, and the red points indicate the RPs location and their ranges.

Table 2 shows the node`s base table after discovering its neigbours and forming the RPs. At this point, each node has its own table with the same fields as other nodes, including the node`s ID, the x and y-axis to represent the location, the degree, which stands for the number of neihbours, and the neighbours' IDs.

Table 2 The node identification table with each node's location, neighbours, and neighbouring degree (an example using 30 nodes).

ID	**xd**	**yd**	**Neighbours**	**Degree**
1	295.69	17.13	13, 23	2
2	467.62	204.41	4	1
3	331.69	434.03	5, 6	2
4	452.19	190.73	2, 9, 21, 26	4
5	348.58	476.98	3, 6, 24	3
6	390.14	481.79	3, 5, 24	3
7	253.44	216.77	30	1
8	212.02	484.34	12, 19	2
9	448.41	92.01	4, 11, 26, 28	4
10	273.39	317.16	15, 22	2

ID	xd	yd	Neighbours	Degree
11	446.57	44.70	9, 26, 28	3
12	113.82	494.76	8	1
13	229.32	9.67	1, 14, 17	3
14	178.30	33.16	13, 17	2
15	232.08	379.24	10, 22	2
16	129.70	279.15	20, 22, 25	3
17	198.68	81.95	13, 14	2
18	30.61	398.76	29	1
19	224.97	495.78	8	1
20	143.82	221.72	16, 22, 25, 30	4
21	360.32	194.88	4, 26	2
22	186.23	305.81	10, 15, 16, 20, 25	5
23	333.21	25.88	1	1
24	443.62	457.84	5, 6	2
25	171.39	290.57	16, 20, 22	3
26	400.12	107.28	4, 9, 11, 21, 28	5
27	434.60	316.89	-	0
28	407.85	98.18	9, 11, 26	3
29	0.43	471.16	18	1
30	175.82	189.02	7, 20	2

3.3.4 Data Distribution

In this phase, each node could sense an event and retrieve a random amount of data. However, that amount of data should not exceed the node's buffer size. Another condition here is that each node should check its energy to determine if it can handle that amount of data or not. If the node is unable to retrieve the whole data, it will calculate how much data can be uploaded depending on its energy level, and then retrieve that amount of data.

3.3.5 Computing the Minimum Path Tour

In this step, the information from $\Re$ is used to compute the minimum tour length. Similar to [28], the problem is modeled using the well-known TSP. The operation of the M-Actor is mapped onto that of a salesman, and RPs are mapped onto cities. The M-actor will start its tour at the sink, visit each RP in sequence according to the computed path, and then ultimately return to the sink. A (TSP_GA), Traveling Salesman Problem (TSP) Genetic Algorithm (GA), is used to determine the minimum path tour.

3.3.6 Data Collection

After computing the optimal route using the TSP, the M-Actor starts its tour, visiting the RPs one by one. At each RP, the M-Actor will stop to collect the data for a time that is equal to:

$$T = \left(\sum_{i=1}^{n} data\right) / DR \tag{3.3}$$

Where i to n represent the nodes in each RP, *data* is the collected data for each node, and the *DR* is the data rate.

3.3.7 Tour Time Calculation

At this point, the M-Actor can calculate its tour time by taking the last two points into account. Each tour time can be represented as

$$T_t = (\min\_dis / MA\_v) + (data / DR) \tag{3.4}$$

Where min_*dis* is the minimum distance of the TSP, *MA_v* is the M-Actor velocity by the distance unit over the time unit, *data* is the collected data for each RP, and the *DR* is the data rate for each node.

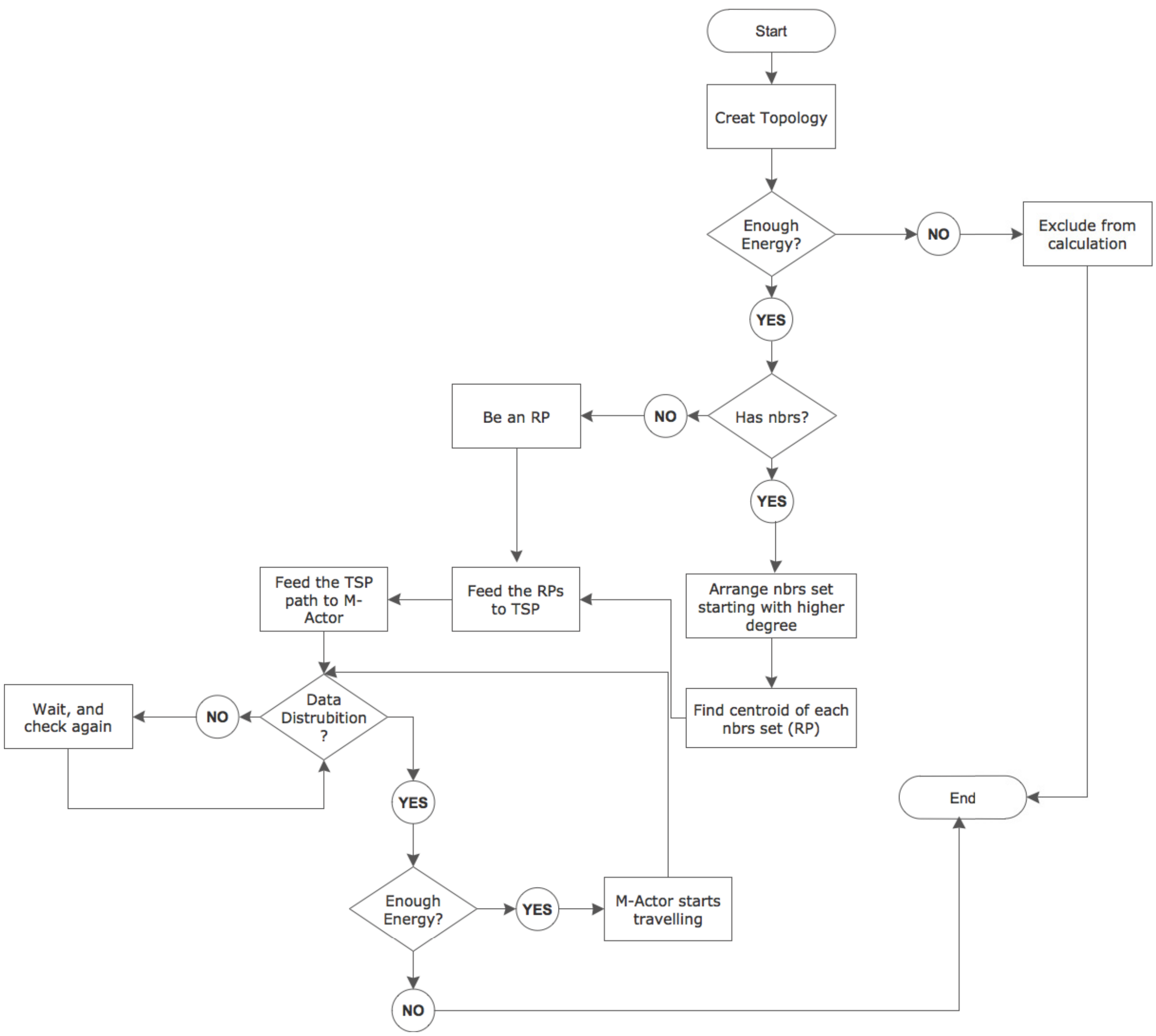


Figure 3.7 Flow chart of the algorithm's operations

Figure 3.7 shows the whole process of the algorithm's operations, from distributing sensors around the field, through data sensing, to computing the route and collecting the data.

3.4 Energy Model

Since energy consumption is the base for any protocol's success, we tried to take this factor into account as we developed our model. To demonstrate the efficiency of our proposed scheme, we applied an energy model to help us to evaluate our proposal. It is a new consumption proposal that was first revealed by Zhou *et al.* in 2011 [31]. The proposed research is divided into four different models that concern about node energy modeling. Since data exchange and energy consumption is our main concern, we chose the Transceiver Energy Model (TEM) to use in our protocol.

3.4.1 Energy Consumption

The Transceiver Energy Model (TEM) assumes that each sensor is capable of working in different energy modes and moves from one mode to another when necessary. There are six different modes in the TEM: receiving, transmitting, idle, sleep, CCA/ED, and off. However, since there is no critical consumption in the off mode, we did not use it in our applied model. The energy consumption model can be expressed as:

$$E = E_{\mathrm{TX}} + E_{\mathrm{RX}} + E_{\mathrm{Idle}} + E_{\mathrm{Sleep}} + E_{\mathrm{CCA}} \tag{3.5}$$

Where the E_{TX} is the energy consumption in the transmitting mode, E_{RX} is the energy consumption in the receiving mode, and E_{Idle}, E_{Sleep}, and E_{CCA} are the consumption in idle, sleep and CCA/ED modes, respectively.

This can be expressed as follows:

$$= \sum_{i=1}^{N_{TX}} P_{TX} L_i / R + \sum_{i=1}^{N_{RX}} P_{RX} L_i / R + P_{Idle} T_{Idle} + P_{Sleep} T_{Sleep} + P_{CCA} T_{CCA} \quad (3.6)$$

Where P_x, L_i, R, and T are the power rate for each mode, data size, data rate, and the time for each mode, respectively. N_{TX} and N_{RX} are the number of nodes in transmitting or receiving modes.

P_x can be calculated by multiplying the voltage into the current unit for the used mode.

$$P = V * I_x \quad (3.7)$$

Thus, the final formula will be:

$$E = \sum_{i=1}^{N_{TX}} (V \, I_{TX} \, L_i) / R + \sum_{i=1}^{N_{RX}} (V \, I_{RX} \, L_i) / R + V \, I_{idle} \, T_{idle} + V \, I_{sleep} \, T_{sleep} + V \, I_{CCA} \, T_{CCA} \quad (3.8)$$

After distributing the data and implementing the energy model, the identification table will be updated and completed in all fields, and an example of this is given in Table 3.

Table 3 The node's identification table with its location, remaining energy, data (in bits), time, nighbours (nbrs), and nighbouring degree (k) after 1800 seconds (an example of using 30 nodes, with initial node energy of 100, and random data).

ID	xd	yd	Remaining Energy	Data	Time	Nbrs*	k*
1	295.7	17.13	77.60	4.89e+5	1.19	13, 23	2
2	467.6	204.4	76.70	3.69e+6	14.44	4	1
3	331.7	434.1	77.82	2.31e+5	0.90	5, 6	2
4	452.2	190.7	75.75	2.30e+6	9.01	2, 9, 21, 26	4
5	348.6	476.9	75.86	2.66e+6	10.42	3, 6, 24	3
6	390.1	481.8	78.64	3.96e+5	1.54	3, 5, 24	3
7	253.4	216.8	76.98	8.76e+5	3.42	30	1
8	212	484.3	75.53	4.12e+6	16.12	12, 19	2
9	448.4	92	76.67	3.24e+6	12.66	4, 11, 26, 28	4
10	273.3	317.1	76.45	4.07e+6	15.93	15, 22	2
11	446.5	44.7	78.77	3.61e+4	0.14	9, 26, 28	3
12	113.8	494.7	76.53	2.32e+6	9.08	8	1
13	229.3	9.67	77.81	5.75e+5	2.24	1, 14, 17	3
14	178.3	33.16	76.29	3.92e+6	15.32	13, 17	2
15	232.1	379.2	76.42	1.7e+6	6.74	10, 22	2
16	129.7	279.1	77.05	2.79e+6	10.92	20, 22, 25	3
17	198.6	81.95	76.74	5.15e+5	2.01	13, 14	2

18	30.61	398.7	77.78	7.44e+5	2.90	29	1
19	224.9	495.7	79.28	2.71e+6	10.62	8	1
20	143.8	221.7	74.67	1.63e+6	6.40	16, 22, 25, 30	4
21	360.3	194.8	77.21	3.96e+6	15.48	4, 26	2
22	186.2	305.8	78.04	4.05e+6	15.85	10, 15, 16, 20, 25	5
23	333.2	25.88	75.39	3.73+6	14.58	1	1
24	443.6	457.8	77.11	3.39e+6	13.24	5, 6	2
25	171.3	290.5	75.48	4.16e+6	16.25	16, 20, 22	3
26	400.1	107.2	76.25	2.05e+6	8.04	4, 9, 11, 21, 28	5
27	434.6	316.8	76.99	5.78e+5	2.26	-	0
28	407.8	98.18	76.86	1.28e+6	5.03	9, 11, 26	3
29	0.43	471.1	77.45	2.75e+6	10.77	18	1
30	175.8	189.1	76.40	3.96e+6	15.50	7, 20	2

3.4.2 Clear Channel Assessment (CCA/EED)

When a sensor has data to send to the M-Actor, it should first check the medium to ensure its ability to transmit the data without risk of a collision. Each time a node wants to transmit, it will wait for a randomly timed back-off period before it starts transmitting. The back-off period is limited in the range between (0 and 2^{BE} - 1) to get the Clear Channel Assessment (CCA), where BE refers to Back-off exponent.

When the node checks the channel, it will find one of two situations:

- Busy channel: then the node will wait for another back-off time, and retransmit again.
- Idle: in this case the node will begin transmitting.

In [33], the initial BE value is set to the Medium Access Control Minimum Back-off Exponent (macMinBE) which has a default value equal to 3. In the worst case, that value can be calculated as in [33] as:

$$InitialBackOffPeriod + CCA = (2^{BE=3} - 1) \times aUnitBackOffPeriod + CCA \quad (3.9)$$

Where the CCA detection time is equal to 8 symbol periods, a unit backoff period is defined as 20 symbol periods, and each single period is set to 16 μs.

Thus, the calculated period will be:

$$= 7 \text{ x } 320\ \mu s + 128\ \mu s = 2.368 \text{ ms} \quad (3.10)$$

3.4.3 Measurement of Energy Efficiency

To evaluate our protocol in terms of energy efficiency, we used the performance metric that was proposed in KAT, with some modifications. We were concerned about the amount of data uploaded by all sensors and the energy consumed to retrieve that data. The average data collected by all sensors (*R*) is equal to:

$$R\ [KB] = \frac{\text{Received and transmitted data by all nodes}}{N} \quad (3.11)$$

Where the data includes both, the sensed one that are collected by the sensors, in addition to that which are uploaded from sensors to the M-Actor.

On the other hand, the average consumption of each node can be calculated using the whole energy amount divided by the number of nodes, as follow:

$$C\ [\text{mWHr}] = \frac{\text{The total energy consumption}}{N} \tag{3.12}$$

Then we can evaluate the system by finding the amount of data collected over the consumed energy:

$$E\ [\text{KB/mWHr}] = \frac{R}{C} = \frac{\text{Received and transmitted data by all nodes}}{\text{The Total energy consumption}} \tag{3.13}$$

3.5 Conclusion

In this chapter, we introduced a novel scheme for data gathering in WSNs. By following this scheme, a set of RPs is formed in accordance with the number of neighbours and the node's energy. By integrating these two metrics, our technique will decrease the number of visited nodes to less number of RPs with more lifetimes. Moreover, we implemented a generic TSP to achieve the shortest M-Actor tours among all of the RPs. Finally; we applied an energy model to measure our technique’s efficiency in terms of energy consumption.

CHAPTER 4 SIMULATION, EVALUATION, AND RESULTS

This chapter provides an evaluation and analysis of our protocol. We use three different metrics to view the results, and compare some of the results with another relevant protocol. Since the aim of this thesis is to propose a new technique that presents a sufficient solution for data gathering, we chose tour length, energy consumption, and network lifetime as our metrics. At the end of this chapter, we use another metric that needs more work in the future. It is the duration of the tour length when using multiple M-Actors. The proposed model was implemented in the Matlab environment to validate its performance, and all results shown are based on an average of 10 running times.

4.1 TOUR LENGTH

We assume that N sensors are deployed randomly in a 2-D plane (500m x 500 m) following a uniform distribution. We assume that the M-actor starts it tour at the point (250,0), the location of a BS. During its tour, the M-actor stops at each RP, collects the data from all sensor nodes in its radio transmission range, and then moves on to the next RP. As mentioned in Chapter 3, a polling based mechanism can be used to avoid contentions. After visiting all RPs, the M-actor returns to the BS to offload the data.

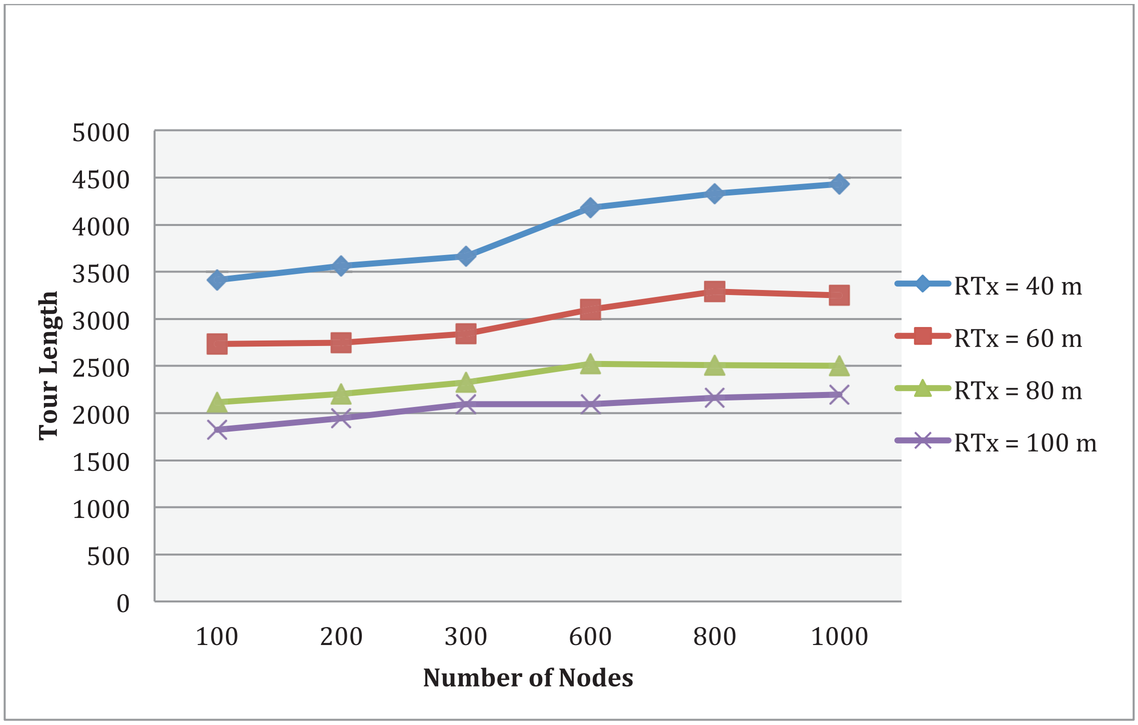


Figure 4.1 Tour Length vs. Network Size for radio transmission range of 40, 60, 80 and 100 m

First, we evaluated the tour length for an M-actor when the network size is increased from 100 to 1000 nodes, as illustrated in Figure 4.1. We observed that in low network densities the tour length was affected significantly by the radio transmission range. For example, for a sparse network consisting of 100 nodes, using an R_{Tx} of 40 m yielded a tour length of 3412 m that was reduced significantly by 2678 m, 2300 m, and 1591 m with R_{Tx} values of 60 m, 80 m, and 100 m, respectively. For sparse networks, it is highly likely that most nodes have very few (in some case zero) neighbors. Therefore, the algorithm computes the RP ensuring that all nodes are covered, thereby increasing the tour length. As the radio transmission range is increased, fewer RPs are required, thus reducing the tour length. However, as the network density (no. of nodes) was increased, we observed an increase in the tour length. The tour length increased because more nodes needed to be covered by RPs. However, the increase in tour length remained relatively small until the network size reached a saturation point. These results are in line with the number of RPs versus the number of nodes shown in the Figure 4.2.

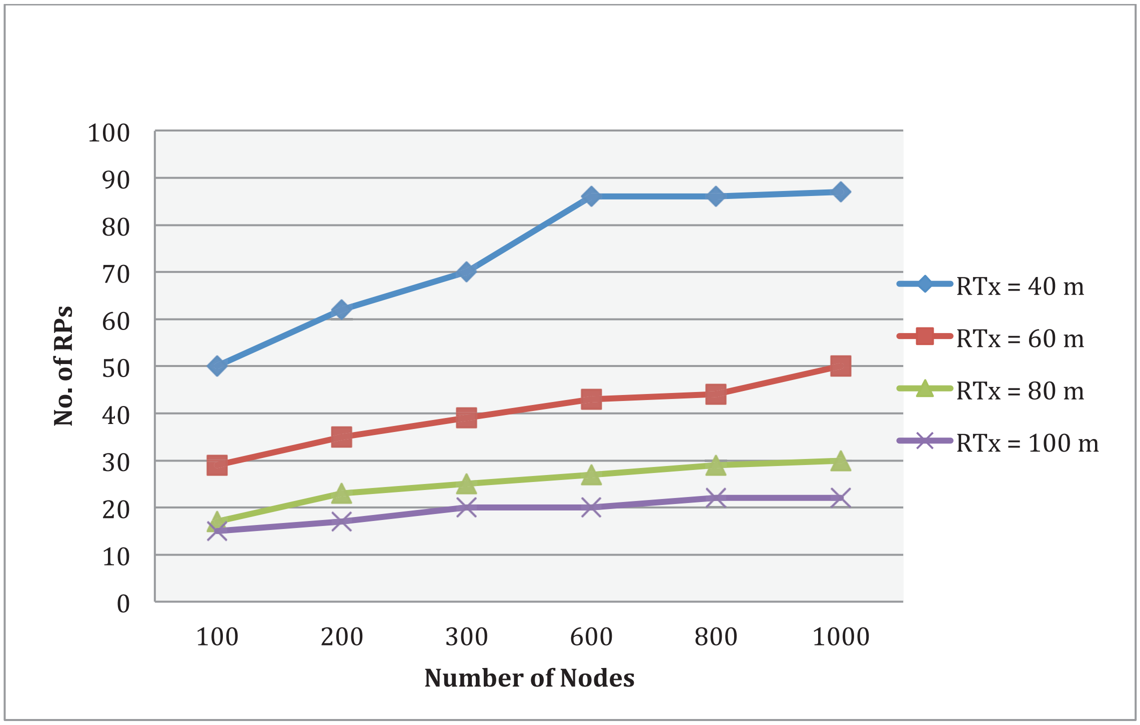


Figure 4.2 No. of RPs vs. Network Size for radio transmission range of 40, 60, 80 and 100 m

Figures 4.3 and 4.4 provide example outputs of the tour generated by our RP-based algorithm when 200 nodes are spread within a 500 x 500 m^2 area with a transmission range of 40 m. As a result, 69 RPs are created.

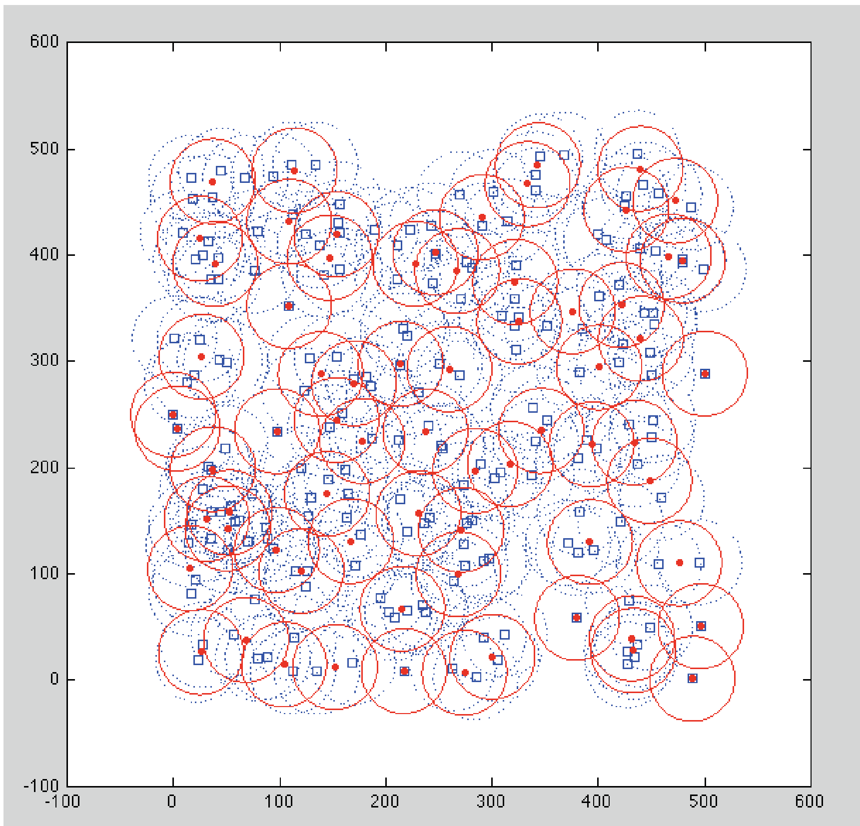


Figure 4.3 Example output of RPs formation for N=200, RTx = 40 m

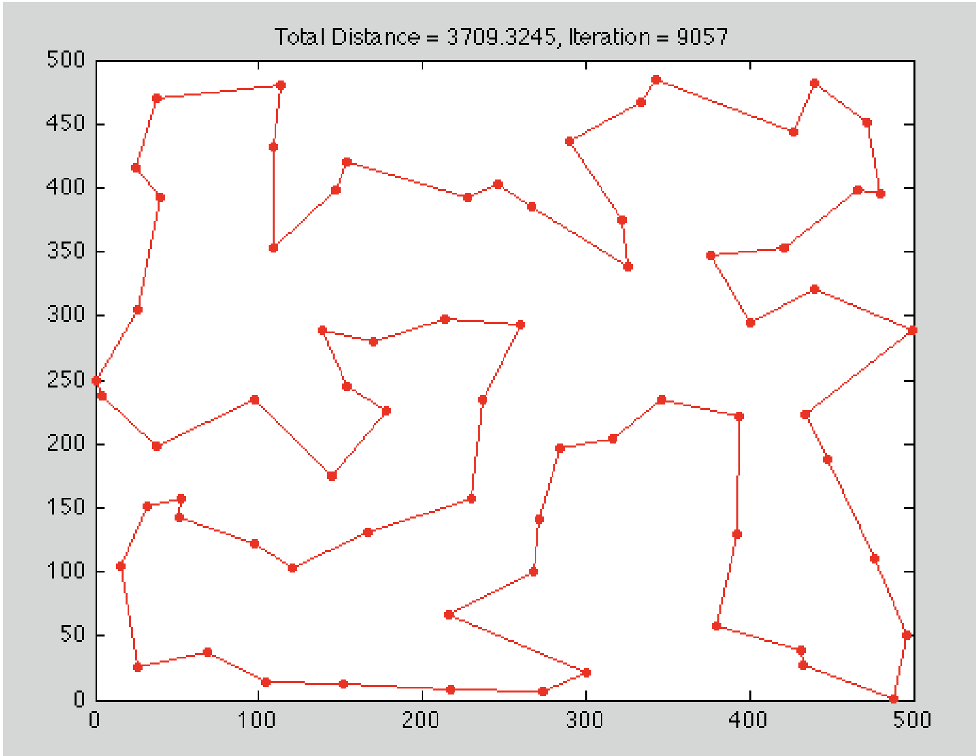


Figure 4.4 Example output of M-Actor tour for N=200, RTx = 40 m

When the transmission range is increased to 80 m, fewer RPs are formed, which leads to a decrease in the tour length. Figures 20 and 21 show the same network in terms of area and number of nodes, and using an 80 m transmission range, which produces 22 RPs. About 68% of the stop points were decreased.

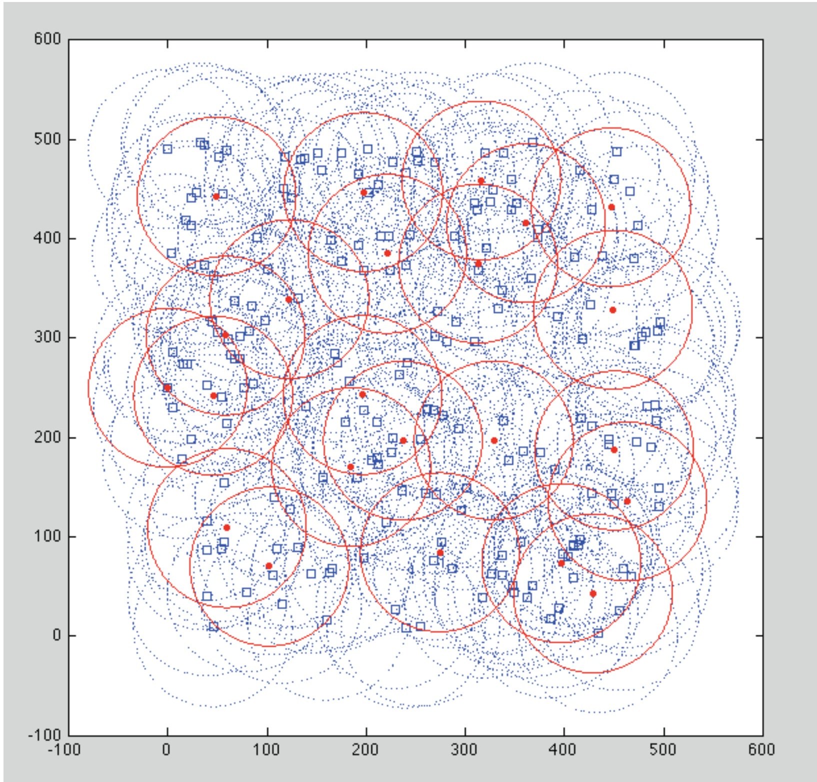


Figure 4.5 Example output of RPs formation for N=200, RTx = 80 m

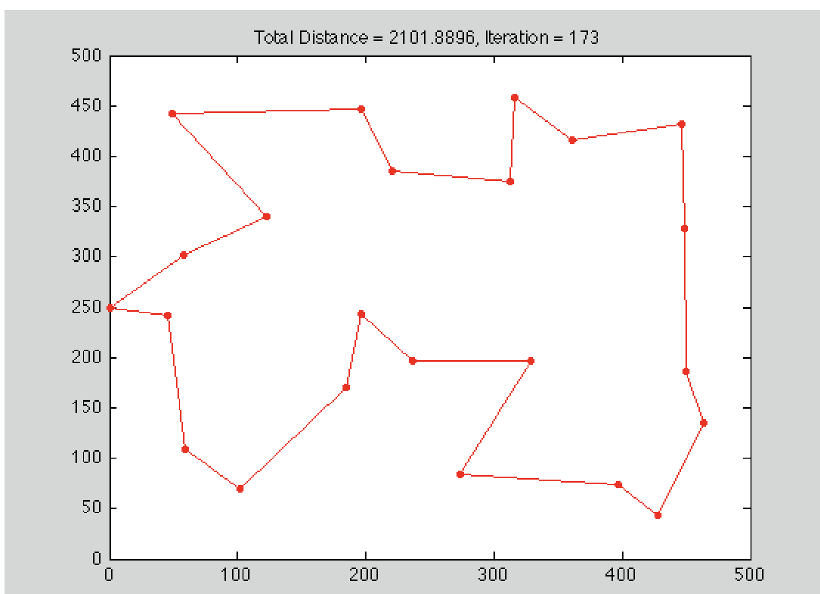


Figure 4.6 Example output of M-Actor tour for N=200, RTx = 80 m

4.2 Energy Consumption

Since energy consumption is one of the critical issues in WSNs and our model is expected to provide an adequate solution regarding that issue, we chose energy consumption as the second metric for our evaluation section. In the energy consumption part, two different scenarios were applied to demonstrate our scheme's efficiency. We used two different kinds of parameters: one for the network topology and the other one for the node's specifications. In order to make a fair comparison, we simulate the KAT model that is discussed early in chapter 2 and compare it to our model.

4.2.1 Scenario 1

The purpose of the first scenario was to test the model with respect to energy consumption and efficiency. We used the same equations that are shown in 3.5. A network size of 5000 x 5000 m was created with various numbers of nodes from 80 – 200. A single M-Actors starts its tour from the base station point at (0, 250) with a random velocity between 10 – 30 m per second. When the M-Actor returns back from its tour collection, it stops for 20 seconds at the base station before starting its next tour. We set up a simulation time of 80 minutes for testing both algorithms. Table 4 summarizes the network parameters for this scenario.

Table 4 Network Initial Parameters

Parameters	Value
Network Size	5000 x 5000 m
Base Station Location	(0, 250)
Number of nodes	80 - 200
Number of M-Actors	1
Velocity of M-Actor	Random between 10 – 30 m/sec
Simulation Time	4800 sec
Pause Time (M-Actor)	20 sec

Since we were concerned about the energy consumption with regards to different modes, we selected a new mote to use in our simulation. The IRIS mote [32] is able to operate in different modes to conserve some of its energy. It is able to work in sleep, active, receive, or transmitting modes. Each of these modes' energy consumption is different than the others. When a node intends to transmit, it will first move into active mode. Then, it will check the channel availability by waiting for a random time, as was described in 3.5.2. If it manages to find an open channel, it will start transmitting. The IRIS motes' parameters were taken from a datasheet as summarized in Table 5.

Table 5 IRIS Mote's Parameters

Parameters	**Value**
Transmission Range	300 m
Buffer Size	512 KB
Transmission Data Rate	250 kbps
Current Draw (Receive Mode)	16 mA
Current Draw (Transmit Mode)	17 mA
Current Draw (Idle Mode)	8 mA
Current Draw (Sleep Mode)	8 µA
Initial Energy	30780 (2x AA Batteries) Joule
Voltage	3 V

IRIS consumes 8 µA in sleep mode, 8 mA in idle mode, 16 mA in receive mode, and 17 mA in transmitting mode. IRIS offers a good transmission range of more than 300 m at the open door environment. Nonetheless, IRIS comes with 512 K bytes of flash memory, and offers a reasonable data rate of 250 kbps. IRIS mote operates with 2 AA batteries; thus, we use a simple battery-operated model. We assume that we use a regular alkaline battery that has an average capacity of 2850 mAHr with a nominal voltage of 1.5 V [36]. Thus, the initial energy can be estimated as:

$$2.85 \text{ AHr} \times 1.5 \text{ V} \times 3600 \text{ Sec} = 15390 \times 2 = 30780 \text{ joule.} \quad (4.1)$$

These specifications were applied to our simulation for both of our model and KAT. Figure 4.7 shows the RPs formation and KAT clustering for the same topology.

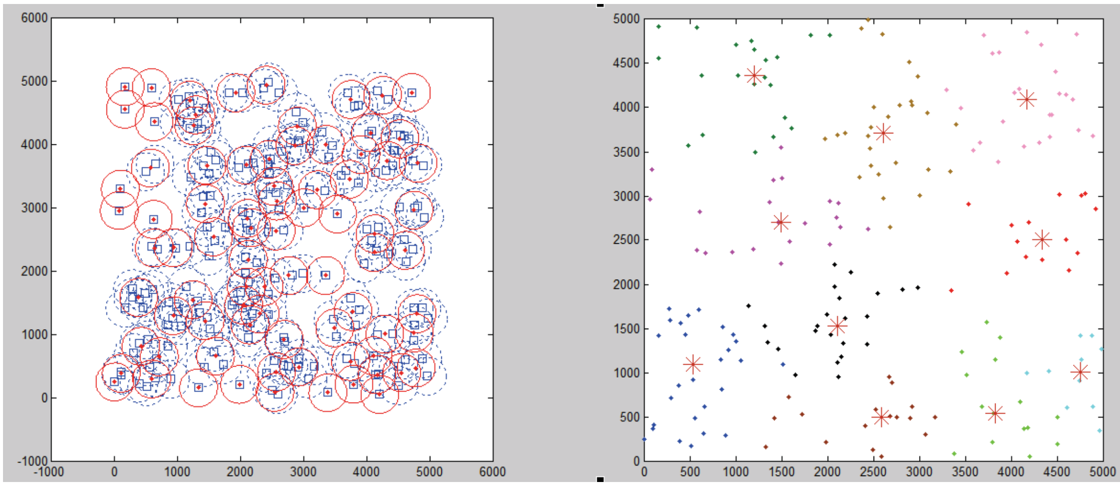


Figure 4.7 RPs formation vs. KAT clustering for 200 nodes

We first tested the two protocols starting with 80 nodes. Our protocol shows better performance in comparison with the KAT protocol by more than double. By increasing the number of nodes, we noticed that the results became closer to each other. However, our model still achieved superior efficiency. This means that our model is able to handle more data with less energy consumption. For example, when using 140 nodes, the nodes in our model received and transmitted about 6436 MB of data, and consumed about 2233 mWHr. On the other hand, KAT received and transmitted 6752 MB of data, with consumption of 3614 mWhr, with the same network size and parameters. In this case, we follow the same performance metrics that were used in KAT to evaluate the effectiveness of our model. We calculated the efficiency by dividing the amount of data in KB by the energy consumption in mWHr as it is described in 3.4.3, which gave 21.89 % for our model and only 13.26 % for KAT. Figure 4.8 presents the performance comparisons between the two mobility models with different numbers of nodes.

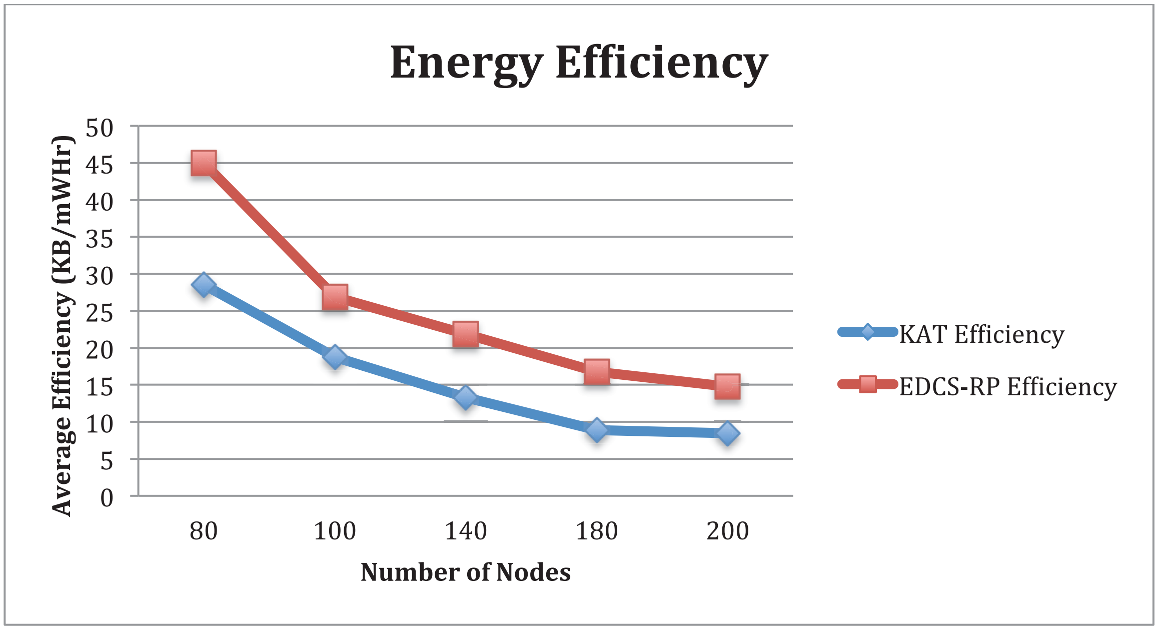


Figure 4.8 Performance comparisons between the two mobility models, EDCS-RP (Energy Efficiency Data Collection Scheme using RP) represents our model

These results were arrived at because our model is designed to guarantee data collection from all nodes in the network. This means all nodes are able to send their sensed data to the M-Actor while they are grouping into an RP, or considering themselves as RPs in cases where no neighbors exist within their range.

Otherwise, KAT model assumes to use a static number of clusters with only 10 clusters. It does not provide a clear point on how to determine the optimal number of clusters. In this case, it is difficult to guarantee that each node can be a part of one cluster, especially when using random distribution and limited transmission range. Thus, some sensors will spent their energy in receiving mode with no guarantee of delivering the sensed data to an M-Actor if it is out of reach. On the other hand, our model ensures that each node is within range of at least one RP, and in the case where no neighbors exist, that sensor's location will become an RP where the M-Actor will travel to collect data. In summary, the sensed data in our model will surely be collected while the KAT model does not guarantee data collection from all sensors. Figure 4.7 shows an example where each node in our model having an RP, where the centroid points (red stars) are too far to reach by their nodes in KAT.

4.2.2 Scenario 2

In the second scenario, we use the same parameters that were used in the first one, with some exceptions that are shown in table 6. The aim of this scenario is to evaluate the two protocols in terms of energy consumption and network lifetime. We increased the M-Actor random velocity to be between 20 – 30 m per second this time, and decreased the simulation time to 60 minutes. To simplify the model and to arrive at reasonable results, we set the node's energy at 100 joules.

Table 6 Network Initial Parameters in the second scenario

Parameters	**Value**
Network Size	5000 x 5000 m
Base Station Location	(0, 250)
Number of nodes	80 - 200
Number of M-Actors	1
Velocity of M-Actor	Random between 20 – 30 m/sec
Simulation Time	3600 sec
Pause Time (M-Actor)	20 sec
Initial Energy	100

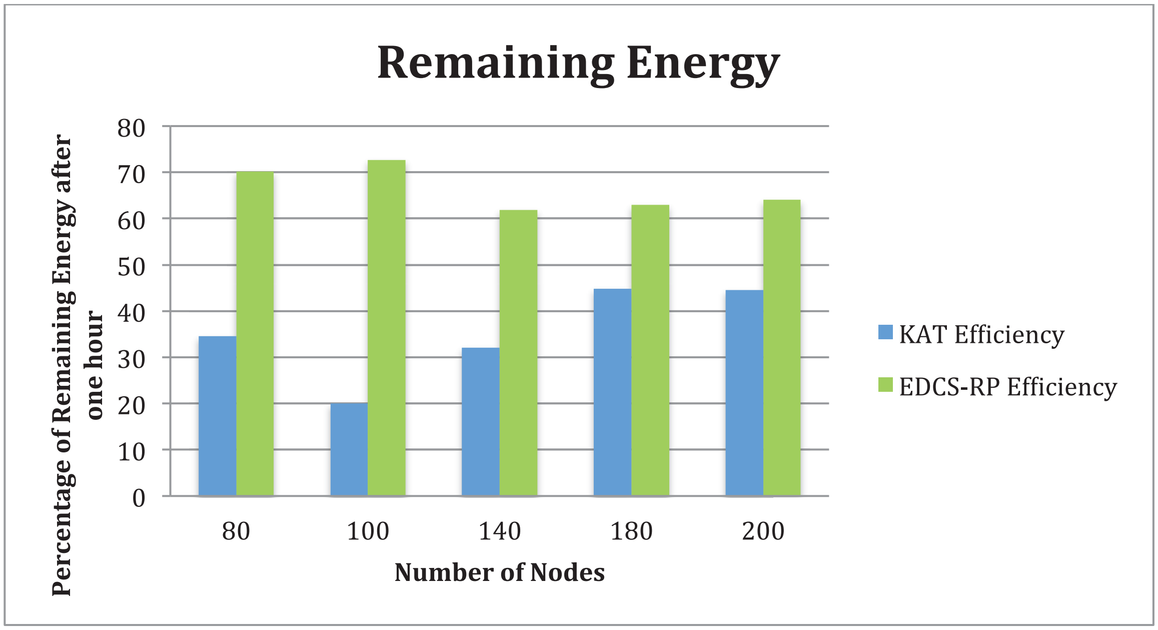


Figure 4.9 Percentage of remaining energy after running the simulation time of one hour, EDCS-RP (Energy Efficiency Data Collection Scheme using RP) represents our model

We evaluated the two experiments to find the remaining energy after 3600 seconds. Again, we used the same network size, between 80 - 200 nodes. From the results, it is clear that the percentage of remaining energy in our model is superior in comparison with the KAT model. Our model shows outstanding efficiency, with more than 70% in the case of 80-100 nodes. Furthermore, it demonstrated an excellent performance when the network size became larger. Figure 4.9 shows 62 - 64 % of remaining energy when 140, 180, and 200 nodes are used. On the other hand, with the KAT model, the experiment showed that in the most efficient case, the percentage of remaining energy did not exceed 45% in case of 100 nodes. Again, the underlying reason for this is the RPs' distribution around the network, which guarantees there will be an RP for each sensor.

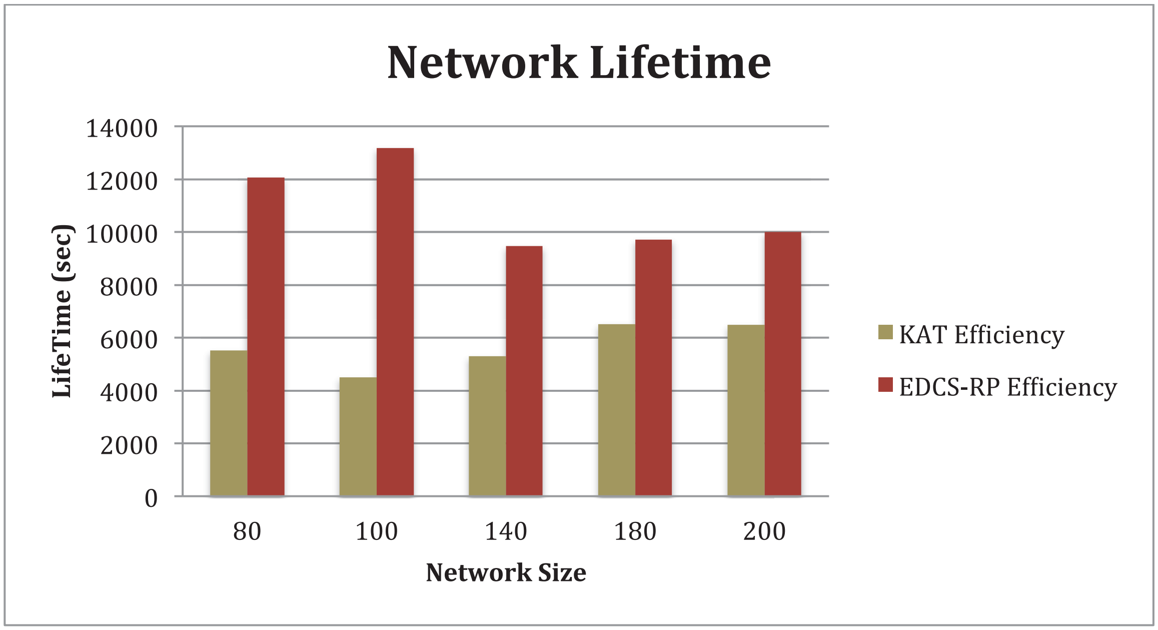


Figure 4.10 Network lifetime for the two models, EDCS-RP represents our model.

In Figure 4.10, we rate the network in terms of lifetime. We used the exact parameters that were used previously. Since we use homogeneous nodes in our model with similar specifications, we consider the network lifetime as the time when all nodes go out of energy. From the experiment, we noticed that all nodes in our model could keep their energy for a long time in comparison with the same number of nodes in the KAT model. This time varies with the network size—the test showed that when using 80 and 100 nodes, the nodes could stay alive for 12060 and 13176 seconds, respectively. In KAT, only 34.15 – 45.67 % of our model's lifetime was achieved. Figure 4.10 displays that with the shortest lifetime in our model when using 140 nodes, the nodes are able to live for 9468 seconds, while in the longest lifetime of KAT when using 180 nodes, the nodes could not live for more than 6519 seconds. This obviously confirms our model's ability to increase the network's lifetime as well.

4.3 Multiple M-Actors

Using multiple M-Actors offers more advantages over using a single M-Actor. At this time, we concentrated on the advantages of using multiple M-Actors for decreasing the tour length time. We implemented the same network topology as before, using 1, 2, and 5 M-Actors. Figure 4.11 represents the RPs formation for a network of 200 nodes in 5000 x 5000 m with the 3 different cases of using multi M-actors when using TSP.

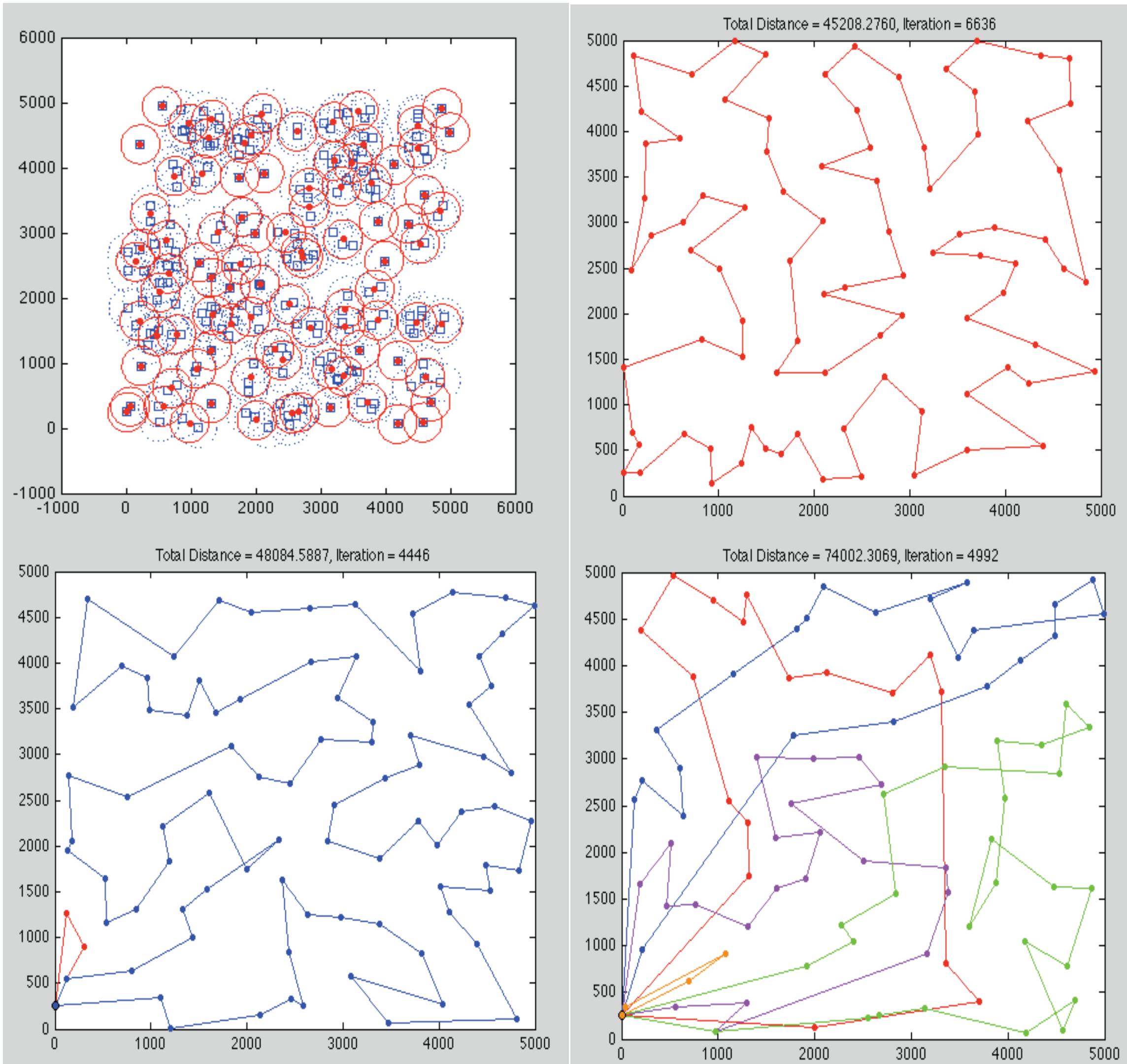


Figure 4.11 RPs formation with the TSP tour for different M-Actors starting from a single one in the upper right, two M-Actors in the bottom left, and four M-Actors in the bottom left.

Figure 4.12 illustrates the time required for different numbers of M-Actors to traverse networks of diverse sizes when using a random velocity between 20 - 30 m/sec. It is beneficial to use multiple M-Actors, as the experiments proved. For example, the required time for traversing a network of 100 nodes can be decreased to about 47 seconds when using 2 M-Actors as compared to 83.69 seconds when using a single one.

Moreover, when hiring more M-Actors, the time spent will be reduced. In another instance, 5 M-Actors required 53.54 seconds to go around 1000 nodes, in contrast with 152.9 and 84.86 seconds being needed when single and 2 M-Actors were used, respectively.

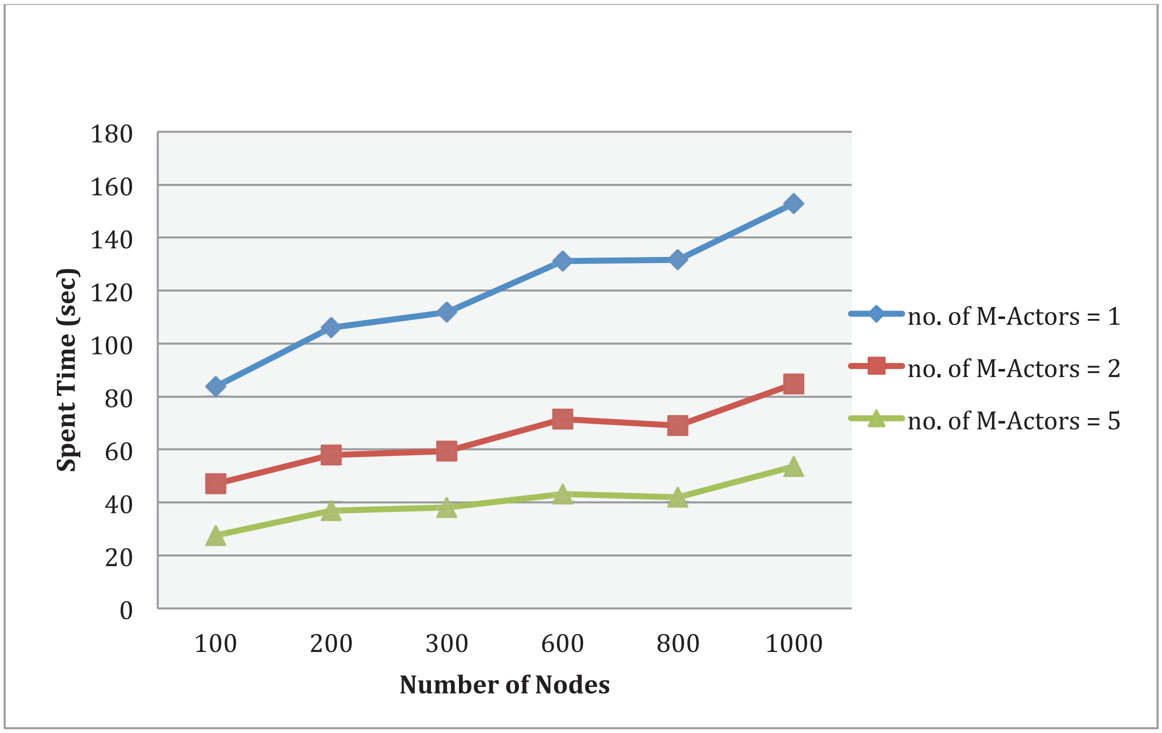


Figure 4.12 Spent Time when using different number of M-Actors starting from single one, 2, and 5 with transmission range of 80

4.4 Conclusion

In this chapter, different scenarios with different parameters were provided in details to rate the efficiency of the proposed model. Based on the results, it is clearly appears that our proposed scheme success to find a way that offering a guarantee for each node to deliver its data to the M-Actor with least tour length that is required for the M-Actor. On the same time, this algorithm shows a significant development in terms of energy consumption in comparison with another clustering scheme that depends on K means algorithm. This success can be extended when using multiple M-Actors to decrease the tour time and delivering data to BS in fastest time possible.

CHAPTER 5 FUTURE WORK AND CONCLUSION

This research aims to propose a new technique that works on integrating mobility into wireless sensor networks to enhance the energy efficiency performance, and decreased the mobile actor tour, which collect sensed data, through the sensors in the network. In this chapter, thesis contributions are summarized in 5.1 with some points that we plan to work on, in the near future as a future work in 5.2.

5.1 CONCLUSION

In this thesis we presented a new approach for data gathering in large scale WSNs using M-actor nodes. Our technique enables the M-actor to compute a set of RPs for data gathering in the deployment area. The proposed technique exploits information exchanged in the network discovery process to build a topological map. A heuristic yields a set of RPs in a manner that guarantees that all sensor nodes will be covered by at least one RP. Using the computed information concerning RPs, the M-actor calculates an optimal tour through a GA-based TSP. A simulation in Matlab was implemented to evaluate this scheme. Different parameters and scenarios were applied to get the results. The simulation results demonstrated the effectiveness of our proposed technique. Chapter 4 shows the methods that were followed to evaluate the proposed technique. Here is a summary of the points that have been gained during this work:

- Our technique shows a high level of efficiency in energy consumption when dealing with a massive amount of sensed data. This scheme provides higher energy efficiency when compare it with the KAT model, in the case where using the same parameters by guaranteeing that all sensors are able to exchange their data with the M-Actor through the RPs points. All sensors are belonging to at least one RP, or they can make themselves as an RP. Thus, each node will be visited by the M-Actor to deliver the sensory data. After delivering its data, the node will change its mode to an idle mode to conserve energy.

- A longer lifetime is also achieved by our scheme, in contrast with the KAT model. The results show that in our model, the sensor nodes can keep alive up to 65% longer in some cases than KAT. Still the well guarantees for data transferring through the RPs, and the ability of work into different modes are the most significant reason to enhance the network lifetime.
- Using multiple M-Actors would be beneficial too in our scheme to reduce the latency and time spent gathering data. The procedure of this scheme aims to minimize the number of visited nodes, which lead to minimize the tour length. When using multiple M-Actors the required tour can be allocated between the M-Actors to accelerate the data gathering process.

5.2 Future Work

While using multiple M-Actor offers outstanding improvements in terms of time spent, it still requires further investigations regarding the distribution and the division of the network between the actors. Figure 4.7 shows the TSP tour for different numbers of M-Actors. In the case of using 2 M-Actors, the illustration demonstrates that there is no equality of the RPs' allocation between the M-Actors to traverse the network. For example, in that case, one actor will travel to only 2 RPs, while the other one is going to visit the rest of the network's RPs, which number 83. This problem would be considered as a part of MRTA (Multi Robot Task Allocation) problem that is discussed in [34]. In our algorithm, when n sensors are given with m number of M-Actors, how to determine and optimize which M-Actor is responsible for which sensors for data collection. Indeed, this problem becomes more complicated when different tasks of nodes or M-Actors are applied. Moreover, reasonable allocation between M-Actors is expected to provide more energy efficiency and a longer lifetime for both sensors and M-Actors, simply by guaranteeing to offload the sensed data from sensors as soon as they can, which means keeping the sensors in idle or sleep mode for longer periods of time. In addition, using multi-hop communication might be beneficial in several aspects. It may help to decrease the number of RPs, which leads to decrease the tour length.

BIBLIOGRAPHY

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A survey on sensor networks,” *Communications magazine*, IEEE, vol. 40, no. 8, pp. 102–114, 2002.

[2] J. Yick, B. Mukherjee, and D. Ghosal, “Wireless sensor network survey,” Computer networks, vol. 52, no. 12, pp. 2292–2330, 2008.

[3] M. Di Francesco, S. K. Das and G. Anastasi. Data collection in wireless sensor networks with mobile elements: A survey. *ACM Trans.Sen.Netw. 8(1),* pp. 7:1-7:31. 2011. Available: http://doi.acm.org/10.1145/1993042.1993049.

[4] M. Zennaro. Introduction to wireless sensor networks. [[Online] http://wireless.ictp.it/wp-content/uploads/2012/02/Zennaro.pdf;]. *2012(June/22),* 2012.

[5] A. W. Holger Karl, *Protocols and Architectures for Wireless Sensor Networks.* England: John Wiley & Sons Ltd, 2005.

[6] S. Abdel-Mageid and R. Ramadan, “Efficient deployment algorithms for mobile sensor networks,” in *Autonomous and Intelligent Systems (AIS), 2010 International Conference on*. IEEE, 2010, pp. 1–6.

[7] A. Mateska and L. Gavrilovska, “WSN coverage & connectivity improvement utilizing sensors mobility,” in *Wireless Conference 2011- Sustainable Wireless Technologies (European Wireless), 11th European*. VDE, 2011, pp. 1–8.

[8] D. Denkovski, A. Mateska, and L. Gavrilovska, “Extension of the WSN lifetime through controlled mobility,” in *Wireless On-demand Network Systems and Services (WONS), 2010 Seventh International Conference on*. IEEE, 2010, pp. 151–156.

[9] A. Alomari, N. Aslam and F. Comeau, "Data collection using rendezvous points and mobile actor in wireless sensor networks," in *IEEE International Conference on Communications,* Ottawa, Canada, 2012, pp. 7227-7231.

[10] J. N. Al-Karaki and A. E. Kamal, "Routing techniques in wireless sensor networks: a survey," *Wireless Communications, IEEE,* vol. 11, pp. 6 - 28, 2004.

[11] W. R. Heinzelman, J. Kulik and H. Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," in *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking* (MobiCom '99), New York, NY, USA, 1999, pp. 174-185.

[12] C. Intanagonwiwat, R. Govindan, and D. Estrin, “Directed Diffusion: a Scalable and Robust Communication Paradigm for Sensor Networks,” Proc. ACM Mobi-Com 2000, Boston, MA, 2000, pp. 56–67.

[13] D. Braginsky and D. Estrin, “Rumor Routing Algorithm for Sensor Networks,” Proc. 1st Wksp. Sensor Networks and Apps., Atlanta, GA, Oct. 2002.

[14] W. Heinzelman, A. Chandrakasan and H. Balakrishnan, “Energy-Efficient Communication Protocol for Wireless Microsensor Networks,” Proc. 33rd Hawaii Int’l. Conf. Sys. Sci., Jan. 2000.

[15] G. Xing, T. Wang, W. Jia and M. Li, "Rendezvous design algorithms for wireless sensor networks with a mobile base station," in *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '08),* New York, NY, USA, 2008, pp. 231-240.

[16] X. Li, A. Nayak, and I. Stojmenovic, "Sink Mobility in Wireless Sensor Networks," Wireless Sensor and Actuator Networks: Algorithms and Protocols for Scalable Coordination and Data Communication, Wiley, 2010.

[17] Y. Gu, H. Liu, F. Song and B. Zhao, "Joint sink mobility and data diffusion for lifetime optimization in wireless sensor networks," in *2007 IEEE Asia-Pacific Services Computing Conference,* Tsukuba Science City, Japan, 2007, pp. 56.

[18] V. S. Gordon and Z. Matley, "Evolving sparse direction maps for maze pathfinding," in *Congress on Evolutionary Computation CEC2004,* Portland, OR, US, 2004, pp. 835-838.

[19] M. Szymanski, T. Breitling, J. Seyfried and H. Wörn, "Distributed shortest-path finding by a micro-robot swarm," in *Ant Colony Optimization and Swarm Intelligence*, M. Dorigo, L. Gambardella, M. Birattari, A. Martinoli, R. Poli and T. Stützle, Eds. Berlin/ Heidelberg: Springer, 2006, pp. 404-411.

[20] Z. H. Ahmed, "Genetic Algorithm for the Traveling Salesman Problem using Sequential Constructive Crossover Operator," *International Journal of Biometrics & Bioinformatics (IJBB),* vol. 3, pp. 96-105, 2010.

[21] G. Srinivasan, *Operations Research: Principles and Applications.* New Delhi: Prentice-Hall, 2007.

[22] M. Khan, G. Pandurangan and V. S. and Anil Kumar, "Distributed Algorithms for Constructing Approximate Minimum Spanning Trees in Wireless Sensor Networks," *IEEE Transictions on Parllel and Distributed Systems,* vol. 20, 2009.

[23] Y. Choi, M. Khan, V. S. Anil Kumar and G. Pandurangan, "Energy-Optimal Distributed Algorithms for Minimum Spanning Trees," *IEEE Journal on Selected Areas in Communications,* vol. 27, pp. 1297, 2009.

[24] C. Wu Yu, E. Chen and C. Fang, "Deploying mobile nodes to connect wireless sensor networks using novel algorithms," in *International Conference on Wireless Algorithms, Systems and Applications,* Chicago, IL, 2007, pp. 199.

[25] R. Shah, S. Roy, S. Jain, and W. Brunette, "Data mules: Modeling and analysis of a three-tier architecture for sparse sensor networks," Ad Hoc Networks, vol. 1, no. 2-3, pp. 215–233, 2003.

[26] M. Ma and Y. Yang, "Data gathering in wireless sensor networks with mobile collectors," in Parallel and Distributed Processing, 2008. IPDPS 2008. IEEE International Symposium on. IEEE, 2008, pp. 1–9.

[27] F. Wu, C. Huang, and Y. Tseng, "Data gathering by mobile mules in a spatially separated wireless sensor network," in *Tenth International Conference on Mobile Data Management: Systems, Services and Middleware*, MDM'09. 2009, pp. 293–298.

[28] H. Nakayama, Z. M. Fadlullah, N. Ansari and N. Kato, "A Novel Scheme for WSAN Sink Mobility Based on Clustering and Set Packing Techniques," *IEEE Transactions on Automatic Control,* vol. 56, pp. 2381, Oct 2011.

[29] H. Nakayama, N. Ansari, A. Jamalipour and N. Kato, "Fault-resilient sensing in wireless sensor networks," *Computer Communication,* vol. 30, pp. 2375–2384, 2007.

[30] M. Ma and Y. Yang, " Data gathering in wireless sensor networks with mobile collectors," in *IEEE International Symposium on Parallel and Distributed Processing,* Miami, FL, 2008, pp. 1-9.

[31] H. Zhou, D. Luo, Y. Gao and D. Zuo, "Modeling of Node Energy Consumption for Wireless Sensor Networks," *Wireless Sensor Network, SciRes,* vol. 3, pp. 18-23, 2011.

[32] Memsic Technology Inc., "IRIS XM2110CA Data Sheet." [Online]. Available: http://www.memsic.com

[33] Jennic., "Jennic Inc., Calculating 802.15.4 Data Rates," 2006.

[34] X. Li, I. Lille, R. Falcon, A. Nayak and I. Stojmenovic, "Servicing Wireless Sensor Networks by Mobile Robots," *IEEE Communications Magazine,* vol. July, pp. 2 - 9, 2012.

[35] J. Ko, C. Lu, M. Srivastava, J. Stankovic, A. Terzis and M. Welsh, "Wireless Sensor Networks for Healthcare," *Proceedings of the IEEE,* vol. 98, pp. 1947 - 1960, 2010.

[36] Energizer Holdings, Inc.,"Energizer E91 Data Sheet." [Online]. Available: http://data.energizer.com/PDFs/E91.pdf

[37] K. Sohraby, M. Daniel and Z. Taieb, *WIRELESS SENSOR NETWORKS: Technology, Protocols, and Applications.* Hoboken, New Jersey: John Wiley & Sons, Inc, 2007.