ANGLIA RUSKIN UNIVERSITY

FACULTY OF SCIENCE AND TECHNOLOGY

A NOVEL LOCATION ENERGY SLEEP MODE SAVING ALGORITHM (LESMS) BASED ON CLUSTERING FOR WSN

FARIS AL-BAADANI

A thesis in partial fulfilment of the
requirements of Anglia Ruskin University for
the degree of Doctor of Philosophy

This research programme was carried out in
collaboration with the Sound and Vision Reference
Service of The British Library

Submitted: December 2017

ANGLIA RUSKIN UNIVERSITY

ABSTRACT

FACULTY OF SCIENCE AND TECHNOLOGY

DOCTOR OF PHILOSOPHY

A NOVEL LOCATION AND ENERGY SLEEP MODE SAVING ALGORITHM (LESMS) BASED ON CLUSTERING FOR WSN

FARIS AL-BAADANI

December 2017

Due to sensors being distributed in an *ad-hoc* manner and their ability to sense different environmental conditions, Wireless Sensor Networks (WSNs) have become very popular in various real time applications and Internet of Things (IoT). The WSN is widely used in monitoring harsh environments and taking decisions on the basis of collected reports. Clustering in WSNs is a proven technique to avoid redundant transmissions to the sink. This allows for better utilisation of scarce network resources such as energy. Most of the clustering algorithms proposed in the literature involve a high number of message exchanges, which results in unnecessary energy consumption.

This thesis proposes a new cluster head (CH) selection protocol based on the average energy and the location of the nodes. Five different approaches for the CH selection were examined and were statistically analysed: (i) random selection; (ii) selecting the node closest to the arithmetic mean node coordinate; (iii) selecting the node at the medoid coordinate; (iv) selecting the node nearest to the region centre; (v) selecting node nearest the BS. The mean distance of all nodes to their local CH was the dependent variable. Furthermore, three node grouping schemes were further studied and compared to the *k*-means clustering: (i) all nodes in the same group; (ii) dividing the

sensor field into four rectangular quadrants and allocating nodes accordingly; (iii) dividing the sensor field into eight sectors and allocating nodes accordingly. *T*-Test and one way ANOVA were used for the *P*-value analysis.

The sleeping mode technique was implemented. The residual energy level was then used as a main factor, for the CH selection. The average energy value and the minimum distance value both became the weighting factors to select the CH. The Assistant Cluster Head (ACH) technique was introduced to the LESMS to act as a backup system in case of the CH power levels reach critical levels or total loss of the CH node .

Finally, a new technique was introduced to perform a periodic CH health check-up to monitors the energy level of the CH. Simulations were performed using MATLAB and Network Simulator 2 (NS2). Results show that selecting the node closest to the arithmetic mean as CH has outcome other approaches. Also, the sensor field dividing scheme has a shorter distance of all the nodes compared to the *k*-mean. The LESMS, under a mixture of workload environments resulted in low power consumption, low delay, low packet drop, and low control overhead. It also showed high throughput, high packet delivery ratio (PDR) and an increase in the total residual energy.

Keywords: Wireless Sensor Networks, Internet of Things, Clustering, Cluster Head, Base Station, Location and Energy Sleep Mode Saving Algorithm, Arithmetic Mean, *k*-mean, Residual Energy, Assistant Cluster Head.

ACKNOWLEDGMENT

To begin with, I would like to express my truthful appreciation, which words will not be enough to describe, to every individual who has helped in making this work achievable.

I start with my supervision team Prof. Sufian Yousef, Dr. Laith AL-Jabouri, as well as the internal examiner Dr. Ian van der Linde whom without their help, guidance and advice; it would become impossible to get to this stage. I am really grateful for all that you did for me through your patient supervision.

All thanks to every member of my whole family starting with my parents and my wife who has given me all kind of support. My parents were the main reason to start this journey and they are the main encouragement to keep me going. Also my brothers, sisters and every member of my family who has helped in his/her own way.

I would like to thank all my friends who have helped and stood beside me in this project, without giving any names. Thank you ever so much, every one of you, especially if you are reading this.

Last but not least, I would like thank the Telecommunication Engineering Research Group (TERG) and UK-India Education and Research Initiative (UKIERI) for making me part of their teams as this has improved my skills and supported me through my education journey.

TABLE OF CONTENTS

FIGURES

EQUATIONS

TABLES

ABBREVIATIONS

ACH	Assistance Cluster Head
AODV	*Ad-hoc* On-demand Distance Vector
AWK	Aho, Weinberger and Kernighan
BP	Belief Probabilities
BS	Base Station
CBMD	Connectivity, Battery, Mobility and Distance
CBR	Constant Bit Rate
CDMA	Code Division Multiple Access
CG	Cluster Gateway
CH	Cluster Head
CM	Cluster Member
CSMA	Carrier Sense Multiple Access
DARPA	Defence Advanced Research Project Agency
DSDV	Destination Sequenced Distance Vector
DSN	Distributed Sensor Network
DSR	Dynamic Source Routing
ET-DRIVEN	Energy-Threshold-Driven Based Reconfiguration
FTP	File Transfer Protocol
GloMoSim	Global Mobile Information Systems Simulation
GPS	Global Positioning System

IDE	Integrated Development Environment
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IoT	Internet of Things
IP	Internet Protocol
J-Sim	Java Simulator
JVM	Java Virtual Machine
LAN	Local Area Network
LESMS	Location Energy Sleep Mode Saving
LID	Lowest ID
LSD	Location Standard Deviation
LSDS	Location Standard Deviation Sleeping
MAC	Medium Access Control
MAN	Metropolitan Area Network
MANET	Mobile *Ad-hoc* Network
NAM	Network AniMator
NS2	Network Simulator version 2
NS3	Network Simulator version 3
OMNeT++	Object Modular Network Test bed in C++
OPNET	Optimal Network Simulator
OTCL	Object Tool Command Language
PC	Personal Computer

PDA	Personal Digital Assistant
QoS	Quality of Service
RREP	Route Reply
RREQ	Route Request
RSSI	Received Signal Strength Indication
SD	Standard Deviation
TCL	Tool Command Language
TCP	Transport Control Protocol
TR	Text Trace File
TRMSim-WSN	Trust and Reputation Models Simulator for Wireless Sensor Network
UDP	User Datagram Protocol
VB	Visual Basic
VB	Virtual Backbone
VB.Net	Visual Basic Network
VBS	Virtual Base Station
VINT	Virtual Internet Testbed
VoIP	Voice over Internet Protocol
WAN	Wide Area Network
WCA	Weighted Clustering Algorithm
WCSim	Weighted clustering simulation Tool
WLAN	Wireless Local Area Networks
WN	Wireless Network

WRP Wireless Routing Protocol

WSNs Wireless Sensor Networks

ZRB Zone Routing Protocol

CHAPTER 1:

INTRODUCTION

1.1 WSN OVERVIEW

According to Kang et al. (2017), *"A wireless sensor network (WSN) can be defined as a network of devices, denoted as nodes, which can sense the environment and communicate information gathered from the monitored field through wireless links. The data are forwarded, usually via multiple hops, to a sink".* WSNs are often defined as collections of nodes organised into a cooperative network. They are self-configured and lack infrastructure, and contain multiple sensors (often hundreds or thousands) to form networks, which can cooperatively monitor large or complex physical environments, as shown in Fig. 1.1. Factors they can monitor within these environments include pressure, temperature, motion, vibration and sound. The acquired information is then wirelessly communicated to a base station (BS) for data observation and analysis. This system propagates the information to remote devices for storage, analysis, and processing. The BS (or the sink) is therefore essentially the interface between the network and users of the network, and queries can be entered into the sink in order to obtain results.

Wireless sensor nodes consist of computing and sensing devices, as well as power devices. Individual nodes have functional limitations, which include their communication bandwidth, storage capacity, and speed processing. Despite this, they are deployed into various environments where they are able to self-organise and obtain required information. In addition to this, wireless sensor nodes can respond to inputs and queries from the end user, and provide information samples as required. Sensor nodes can therefore be either event driven, or their function can be continuous.

Increasing research, over the last decade, into optimising the activity of wireless sensor networks is driven by the requirement for lower energy consumption in order to prolong the lifetime of the network, as well as the requirement for reduced complexity of the devices. It is therefore essential to maintain a balance between data processing and communication (Chinara and Rath, 2009).

Much of the current researches (Afsar and Tayarani, 2014) focus on optimising energy-efficiency through generating algorithms. This research, however, will focus on the effects of grouping of wireless sensor nodes into clusters, and cluster formations. Combining this with a WSN protocol will potentially allow prolonging the network lifetime to produce an efficient and quality-aware protocol.

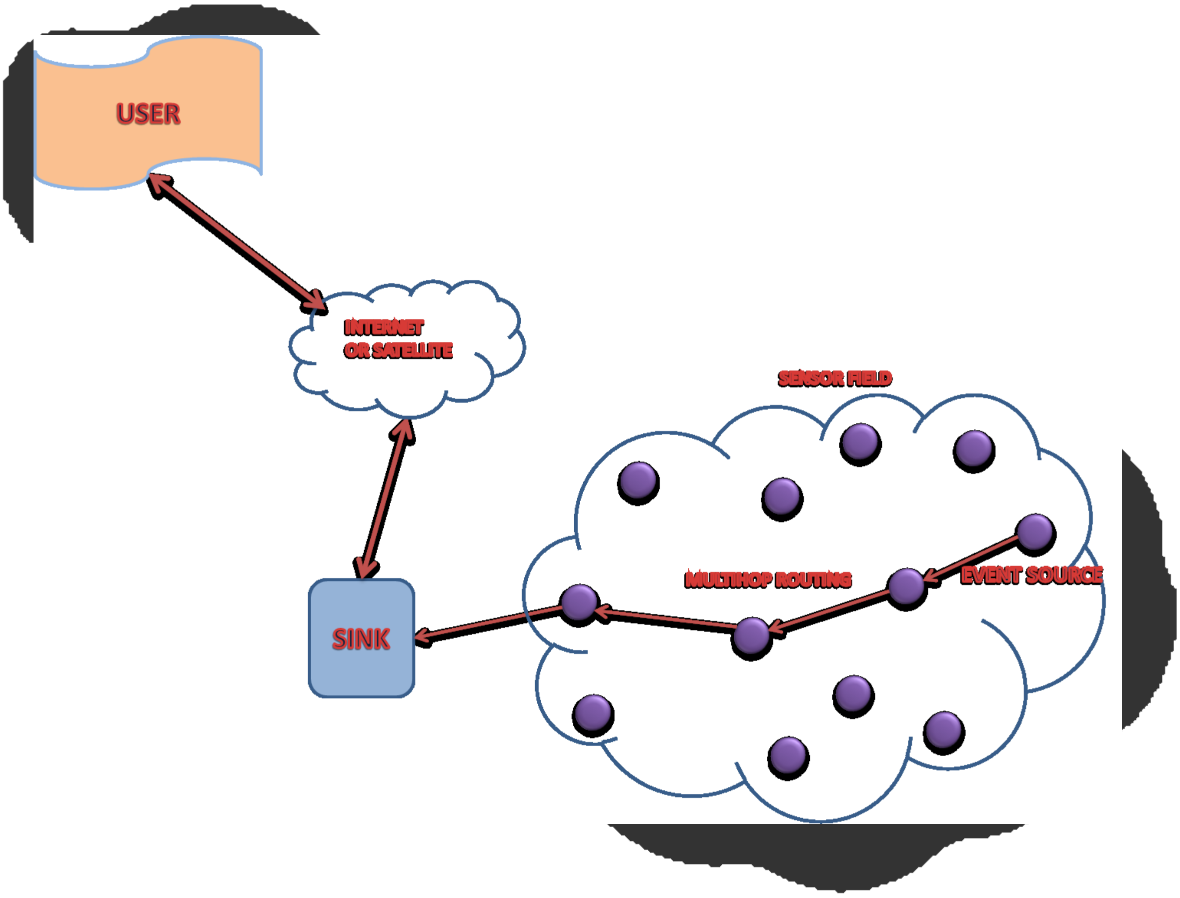

Fig. 1.1 Typical functional architecture for Wireless Sensor Network

1.2 Research Questions

- What cluster head (CH) selection approach in WSNs that can produce the best Quality of Service (QoS), shortest delay in routing and less overhead when adopted as part of an efficient protocol?

- Is the new protocol able to include more than one parameter in selecting the CH and would energy be used as a criteria?

- How will the new protocol support multiple heterogeneous applications when the MAC and transport layers are used together to produce an efficient and quality-aware protocol?

- Will the new protocol make a difference in QoS?

- Does the new protocol make a difference in the energy consumption, which is the main challenge of WSN?

1.3 Research Objectives

- Study, simulate and analyse the performance of the CH selections approaches in the WSNs to obtain the best QoS, shortest delay and minimum overhead.

- Find the optimum CH selection method.

- Test the connectivity of the new protocol with respect to the energy consumption and develop an assistant technique to reduce energy consumption.

- Test and develop a new protocol to be implemented in real-life applications.

1.4 Challenges in WSNs

Due to the use of WSNs in critical applications, it is important that the few limitations that remain are addressed in order to carry out efficient data gathering. A key limitation that still needs to be addressed is security (Zia et al., 2006). Without security, WSNs are vulnerable to attacks as a result of being unattended, and the accessibility nature of wireless communication. Currently, WSN security is ensured through authentication, which is a preventative measure. This measure alone will not be sufficient to ensure that the WSN is not susceptible to attack, for example from people who possess the authentication key (Sharma et al., 2011).

Sensor nodes can often fail, either by themselves or as a result of their deployment into harsh conditions, such as deserts, volcanoes, or warzones. Similarly, they can be destroyed by outside means, such as by humans, animals or fire. Commonly, sensor nodes fail due to energy depletion. Thus, those maintaining the nodes require information regarding the status of the node, or its imminent failure. A node that had failed will no longer transfer information or communicate with other nodes effectively and hence, makes the network further susceptible to attacks. In comparison to wired or infrastructure-based networks, the wireless networks' nodes failures are usually much higher. This further emphasises the need for the immediate detection of these failures in addition to the maintenance of the overall system.

In the context of a routing protocol, the requirement of the availability of alternate packet rerouting paths is also emphasised, as the differences in the environments of deployment can result in different requirements when considering fault tolerance (Han et al., 2010).

Another issue that requires consideration in maintaining the performance of a sensor network is the scalability of the network. As the scale of sensor networks can range from a few nodes to hundreds or thousands of nodes, it is important that nodes in a high density network can reach a level in which a large number of nodes are within the range of the transmission. Considering this, sensor network's protocols must be scalable in order to achieve optimal performance at high-density levels (Akkaya and Younis, 2005).

Sensor units require the following components in order to function appropriately (Matin and Islam, 2012);

- A power supply,
- A transceiver unit,
- A processing unit and,
- A sensing unit.

Nodes can also have additional components such as localisation systems or built-in sensor devices for routing that require location awareness. These additional features, however, increase the power consumption and the size of the node, thus increasing their cost. Sensor nodes are considered to be disposable devices in several models of deployment, therefore, if the nodes can be produced at a low cost (approximately £1) and in large volumes, sensor networks can be used as a more efficient alternative to existing methods for obtaining information. The balance between functionality and nodes' components needs to be carefully considered against the cost and energy-efficiency of the nodes (Akyildiz et al., 2002).

The QoS of wireless packet networks is also an important aspect of network design. This has become an increasingly important consideration over the last decade. Due to increased demand on the media connectivity alone will not suffice to ensure that the network can efficiently deliver the information. For techniques such as Voice over Internet Protocol (VoIP) to function appropriately, additional considerations need to be taken, such as bandwidth, jitter and packet losses. These considerations, however, are complex in WSNs, and remain an issue that needs to be overcome in the design of these networks (Matin and Islam, 2012).

1.5 Motivation

Over the last decade, the use of WSNs has vastly grown, indicating their importance in numerous environments. In the near future, small intelligent devices can be deployed in homes, streets, oceans, rivers, and highways to monitor the environment in an efficient, energy-saving and cost-effective manner (Su and Akyildiz, 2005). They can be routinely used in environmental applications or in surveillance, where they can continuously obtain data regarding light, temperature, pressure, humidity, and many more, whilst being left unattended. The flexibility, fault tolerance, high sensing reliability and low-cost of sensor networks creates many new and exciting application areas for remote sensing. The wide range of WSN application areas is pointing towards sensor networks becoming an integral part of our lives. High scalability is important in maintaining the efficiency of a system, and as a result, nodes have often been grouped into non-overlapping clusters.

Despite the many advantages of using WSNs, a limitation that is becoming more widely taken into consideration is the short life span of sensor nodes (Matin et al., 2011). WSNs comprise networks of battery powered sensor nodes. These provide limited, on-board, processing and storage capabilities. Due to the difficulty of finding or replacing batteries following deployment of nodes, applications and protocols have been designed such that the network is more energy efficient, this would prolong the network lifetime (Ndiaye et al., 2017).

More awareness about a cluster’s characteristics can aid in the understanding of energy consumption and its distribution in WSNs. Therefore, investigation of cluster characteristics is a sensible starting point in the optimisation of energy efficiency and network scalability in WSNs. Clustering in WSNs can resolve the issue of energy consumption as well as the issue of network scalability through dividing the network into small groups 'clusters'. All local nodes will communicate with their own CH, and other CHs will in turn communicate with other CHs, therefore, allowing less communication between all nodes, hence more reliable networks. In other words, this will reduce the overhead messages and the overall power consumed by the node. The result will be prolonging the lifetime of the network.

1.6 Contributions

This research proposes to carry out performance analyses by simulation, using NS2 open source software (NS2, DARPA). These analyses would be used to establish the advantages and the drawbacks of existing protocols. One protocol, that suits the format of clustering in the wireless sensor networks, will be adopted. This unique sensor network protocol would then be developed to support multiple heterogeneous applications, where the MAC and transport layers can be used together to provide an efficient and quality-aware transmission mode by managing of the number of exchanged messages between the nodes, retransmission, losses and priority, in an intelligent manner.

Clustering in WSN is an important part for this research where CH selection is the core of this research. The CH selection emphases are on selecting the CH based on more than one parameter. Since energy use is a huge challenge in WSN, it is regarded as one of the main parameters that are used to select the CH. The other factor is the shortest node distance to the arithmetic mean.

The theoretical work proposed in this thesis has motivated the researcher in applying it in real-life scenarios, where it has involved hardware as well as software. This research has taken into consideration how complex the work can be when the theory is implemented in real-life. Thus, this part of work will focus on relatively small real-life network scenarios. The hardware experiments will demonstrate the single node architecture, which is an independent module that can sense, analyse and transmit information. It would also demonstrate the multi node architecture using an application with three sensor nodes connected to each other.

The research will compare different approaches for the CH selection, this will then be examined by simulation and statistically analysed. The research will compare five approaches in order to select the optimum CH, these are: (i) random selection; (ii) selecting the node closest to the arithmetic mean node coordinate; (iii) selecting the node at the medoid coordinate; (iv) selecting the node nearest to the region centre; (v) selecting node nearest the BS. The dependent variable will be the mean distance of all nodes to their local CH. In addition, three node grouping schemes will be further investigated: (i) all nodes in the same group; (ii) dividing the sensor field into four rectangular quadrants and allocating nodes accordingly; (iii) dividing the sensor field into eight sectors and allocating nodes accordingly. The schemes proposed will be compared to the *k*-means clustering where full analyses of the obtained results will be provided and analysed, in the light of the P-value, using t-test and one way ANOVA.

One of the main challenges in WSN is the ‘energy consumption’, due to the limitation of the power source (battery). This research will introduce a Location Energy Sleep Mode Saving (LESMS) as a new protocol where the sleeping mode techniques will be implemented in order to overcome the excessive power consumption. This technique requires member participating nodes to go into a temporary sleep-mode when there is no data to be sensed. However, the node sensing device will always be awake to insure a reliable sensor network. Furthermore, the energy will be introduced as a main factor, as well as the arithmetic mean in the CH selection. This will be achieved by taking the average residual energy form all the nodes as an indicator of the minimum battery level required for any node to be CH. The average energy value and the minimum distance value will be the parameters weighting factors to select the CH.

An Assistant Cluster Head (ACH), will be introduced and implemented to take over the main CH duties when the CH battery reaches low levels. The final stage of LESMS will be the creation of the health check-up technique, where the CH is able to pass its duties to the ACH before it dies.

1.7 Thesis structure

This thesis embodies the development and the designs of a unique sensor network protocol, through the quantification of the advantages and drawbacks, of several algorithms that are routinely used in simulations. The thesis is structured as follows:

Chapter 1: Introduction and Motivation

This chapter describes the rationale and motivation behind this research, highlighting the issues and concerns that are in need of deeper research and how these issues may be addressed.

Chapter 2: Background of WSNs and Clustering

Background information is given regarding the WSNs and sensor nodes and their functions in current networks. This covers the description of the sensors nodes, their network topologies, which include star, mesh and hybrid. In addition, the WSN architecture, protocol stacks, and the relationship to the *ad-hoc* networks are provided. An overview of clustering, objectives, advantages and classification of its algorithms are discussed in this chapter.

Chapter 3: Literature Review

In this chapter the literature that relates to the area of WSNs is compared and discussed. Different simulation approaches and some of the main algorithms, used in system simulations, are also described.

Chapter 4: Methodology and Simulation Design

This chapter describes the current research methodology used in the simulation design and the implementation of new algorithms. An overview of the simulation is discussed as well as what is expected from simulations. Then, information about WSN simulation tools are presented. Furthermore, the practical and real-life aspects of the work are shown in the hardware experiments.

Chapter 5: Location-aware Protocols

This chapter is aimed to compare different approaches for the CH selection. These will be presented in simulations and statistically analysed. In addition, three node grouping schemes will be further investigated and compared to the *k*-mean algorithm. *T*-Test and one way ANOVA will be used and the P-value will be obtained to show the differences.

Chapter 6: LESMS Protocol

This chapter includes the introduction of the energy, as a main factor, as well as the closest node to the arithmetic mean, in selecting the CH. Following that the discussion of an Assistant Cluster Head (ACH) is created as a new technique that will be used to back up the CH with the networks' heath check-up.

Chapter 7: Conclusions and Future Work

This chapter includes the thesis contributions, where summaries of the findings are presented, as well as possible considerations for future work.

CHAPTER 2:

WIRELESS SENSOR NETWORKS

2.1 BACKGROUND

Networks are commonly used for the distribution of information in most companies and organisations. As a result, networks form the basis of most programs in that they are the main concept behind data transfer and data availability for members of a given network, without the limitation of the location of the users of the network. There are several types of computer networks which vary in what they can do, these include Wide Area Networks (WANs), which connect several computers that are located over a large geographical area such as a country. Local area networks (LANs) are local area networks, which connect computers that are located within a smaller area, such as in a building. Metropolitan area networks (MANs) are metropolitan area networks, which connect computers that are located in a city or a town. Finally, wireless networks, (WNs) connect computers that use wireless media, such as infrared or radio transmission. When a wireless network (WN) covers larger areas, it becomes a 'wireless WAN', and if it covers smaller areas, it becomes a 'wireless LAN' (Tanenbaum, 2003).

Wireless Local Area Networks (WLANs) are becoming the most highly used networks by individual users as well as corporate organisations, where nowadays, worldwide, WLANs are used by frequenters of large establishments such as shopping malls and airports (Eren, 2005). Devices that are used by individuals, such as laptops and mobile phones require wireless communications between both sender and receiver of information, in order to exchange information between each other or to connect them to the internet. This data transfer using microwaves and radio waves, and the lack of limitation of the geographic location of the user, is often what defines wireless communication (Shorey et al., 2006). Nowadays, wireless technologies are highly popular and are used in a large number of types of devices, ranging from mobile phones to tablets and laptops. However, despite this, there is still a requirement for optimisation of wireless routing and communications protocols (Shorey et al., 2006).

2.2 HISTORY

Research into WSNs originates from DSNs (distributed sensor networks), which were originally founded in 1980 at the Defence Advanced Research Projects Agency (DARPA) (Chong and Kumar, 2003). Until WSNs became more widely used in research, the 'Aparnet' was being used in several institutions, before the Internet was founded. An advantage of using DSNs was their low cost nodes, which could function autonomously and freely to any given node that was available to process the information at a given time (Chong and Kumar, 2003). Of course, DSNs were used at a time when laptops and personal computers were not yet in use, and 'minicomputers' were the main devices used by DSNs. In 1978, the components for the technological application of DSNs were added in a DSN workshop. Of these components, algorithms, software and acoustic sensors were assigned, as well as the 'Accent' operating system, which was designed to be communication oriented (Rashid and Robertson, 1981).

Accent allowed full access to distributed resources. An example of DSN being used in real-life scenarios is its previous use in helicopter tracking (Myers et al., 1984). This system used acoustic microphones to detect and match signals, and was founded at the Massachusetts Institute of Technology (MIT). However, there were limitations involved when developing DSN networks, the most significant being the large size of sensors used in a DSN. Their size proved to be a limitation to the number of applications that could be processed by the system. More recently, there has been a dramatic advance in electromechanical technology, allowing for WSN research to be more like the network that was envisioned at the time of DSN development.

WSNs became largely popular around the globe, and there has been a renewed interest in their development. The techniques that are being applied to improve this technology aim to produce very dynamic *ad-hoc* sensor nodes. As a result, nodes have become smaller and cheaper, allowing for their use in many more applications, including environmental sensing (AL-Karaki and Kamal, 2004).

As mentioned previously, DARPA played a role in network development, and another program designed by them was the SensIT initiative (Kumar and Shepherd, 2001), which resulted in an increase in the capabilities of sensor networks, allowing dynamic querying and tasking, multitasking and reprogramming, and *ad-hoc* networking.

2.2.1 Sensor Nodes

A sensor is an object that performs a sensing task; converting one form of energy in the physical world into electrical energy, to gather information about physical objects or areas (sensing).

Sensor nodes are typically small, wireless devices, consisting of five general components as shown in Fig. 2.1:

- processing (small microcontroller)
- Communication (radio transceiver or other wireless communications device)
- Energy source (or means of generating power)
- Sensor
- Memory

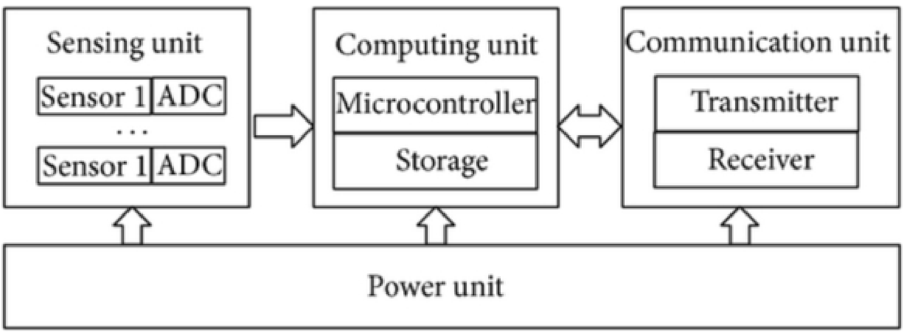

Fig. 2.1 the components of a sensor node (Jiang et al., 2015)

Depending on the application that the node is used for, it can also comprise additional components such as a power generator or a location finding system. These can be re-programmed or replaced based on the network requirements, which allows for a range of different sensors that can be used in each node.

Remote nodes can acquire data commanded by a base station, and the firmware of the node can be upgraded through the WSN. Microprocessors within nodes have several roles in the functionality of the node, including:

- power management
- radio network protocol
- data collection management
- managing the interface between sensor data and radio layers

To minimise the energy consumption of a system, data is only sent over a radio network when required, due to the radio subsystem requiring large amounts of power. This is decided upon application of an algorithm, which assigns the time-point at which data can be sent, based on sensed stimuli. To further minimise energy consumption of the node, hardware may be designed to give the microprocessor control over energy used by the radio sensor, and the sensor signal conditioner. (Akyildiz et al., 2002).

Fig. 2.2 shows some of the sensors in real-life, comparing the size of the sensor to the size of human hand where such sensor can be used to monitor room temperature or fire flame.

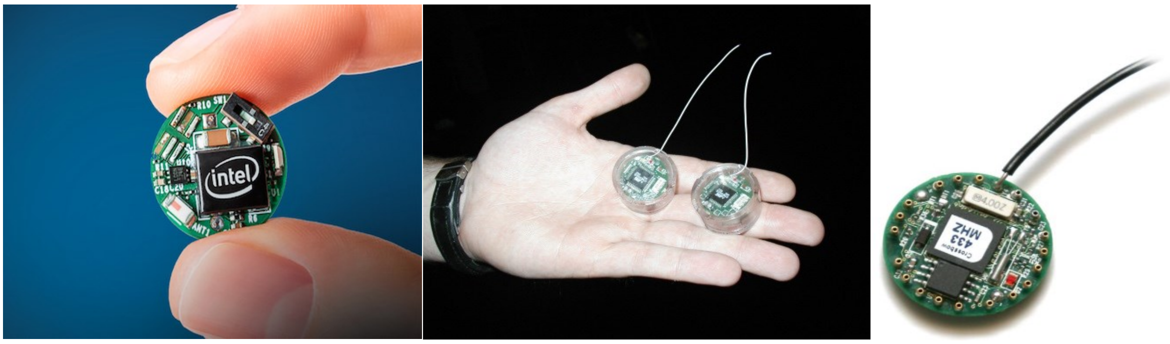

Fig. 2.2 Sensor nodes from real-life scenarios

2.3 WSN NETWORK TOPOLOGIES

The general WSN structure can incorporate various topologies in radio communication. Some of the various topologies are described in the following sections.

2.3.1 Star network (single point-to-multipoint)

The communication topology of a star network involves the transfer of packages/messages from a single base station to several remote nodes (Fig. 2.3). All remote nodes are connected to one main node (gateway), but are not connected individually to each other. As a result, individual nodes can only communicate with each other through the main node, which acts as a 'mediator'. This type of network is advantageous, in that it is a simple network, able to reduce the power consumption of remote nodes. Similarly, this type of network reduces remote node and base station communication latency. A disadvantage of the star network is that it is not robust, due to its requirement that all individual nodes should be within radio transmission range of the base station. As a result, this network also relies on a single node, which can also make the system less robust (Reina et al., 2013).

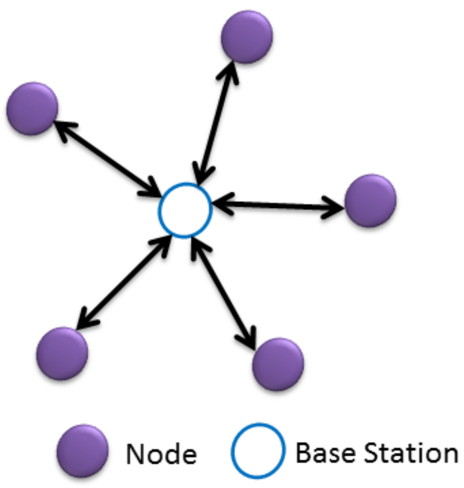

Fig. 2.3 An example of star network topology with 5 nodes and 1 base station

2.3.2 Mesh network

The main identifying feature of a mesh network is that it is made up of nodes that are connected to each other, such that they can transmit data to each other within the range of radio transmission. As a result, multi-hop communications can occur when a node requires the transmission of data to a node out of the radio communications range, by using an intermediate to communicate to another node as shown in Fig. 2.4.

The mesh network has high scalability and high redundancy, meaning, that if one node were to fail, other nodes retain the ability to communicate with other nodes that are in its range, and as a result, the message can be forwarded to a specific location. The range between single nodes does not limit the range of the mesh network, as it can be extended by simply adding more nodes. A disadvantage of the mesh network is its higher power consumption; as nodes within a mesh network generally implement multi-hop communications, which requires higher energy usage, which in turn would limit the lifetime of the network. Also, increasing the number of nodes to this system would increase the length of time required to transmit messages (Wilson, 2004).

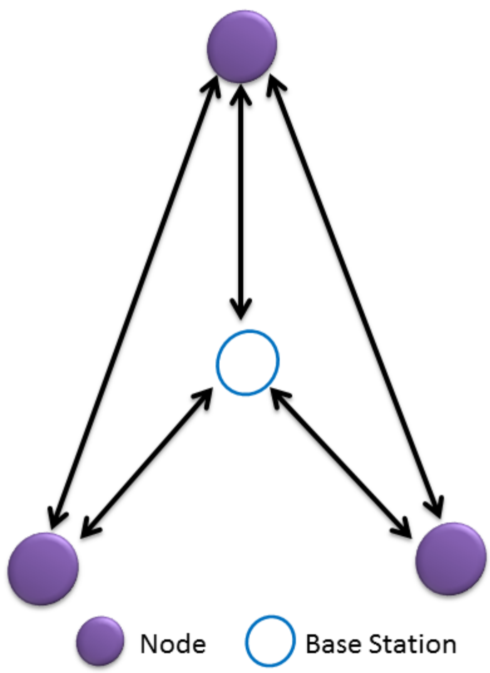

Fig. 2.4 An example of mesh network topology with 3 nodes and 1 base station

2.3.3 Hybrid star – Mesh network

The hybrid star-mesh network is a compromise between the star and mesh networks, combining the advantageous aspects of both (Fig. 2.5). This network can significantly reduce the power consumption of nodes. Using this network topology, low-power nodes are not able to forward a message, which in turn maintains the minimal power consumption throughout the system. Other nodes, however, are given multi-hop capabilities, which allows the forwarding of data from nodes that have low-power status to other nodes. 'ZigBee', the high level network communication protocol, implements the hybrid star-mesh network (Townsend and Arms, 2005).

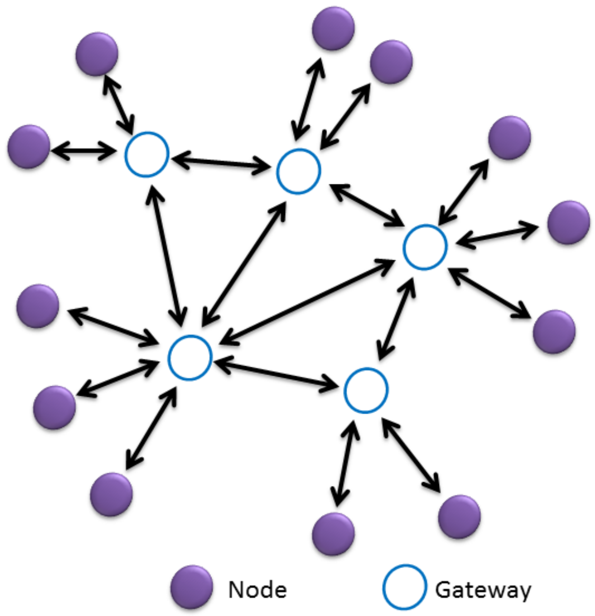

Fig. 2.5 An example of hybrid star – mesh network topology with 12 nodes and 5 base station

2.3.4 Tree Network topology

The communication topology of a tree network involves the transfer of packages/messages from a leaf node to its parent nodes. In turn, a node, which receives data from a child node, will then send the data to another parent node (receiver) after aggregating the received data with its own data. In this way, data flow from leaf nodes to the core node, which normally acts as the sink. The main idea behind the tree technique is that data can avoid flooding by having the data sent using unicast as an alternative of broadcast. As a result power consumption can be saved using the tree topology. Fig. 2.6 shows a typical layout of the tree topology. The arrows show the data flow from a leaf node to the base station node (Mamun, 2012).

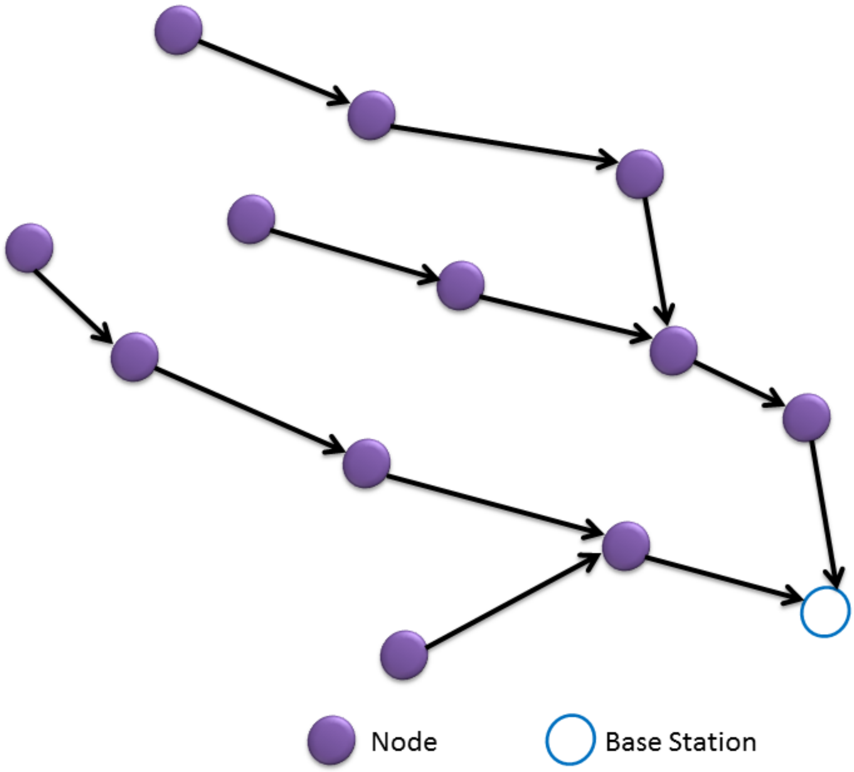

Fig. 2.6 An example of tree network topology with 12 nodes and 1 base station

2.4 NETWORK ARCHITECTURE

In a WSN, several sensor nodes make up a communications network. These nodes have communications, computing and sensing capabilities, and are deployed when desired into a sensing field, to monitor the physical world. The end user can obtain the data via the internet. A WSN sketch map is shown in Fig 1.1.

Two network topologies are shown. Sensor nodes can form a flat network topology where nodes transfer data to sinks via multi-hop routing. In this topology, the nodes act as routers. Similarly, they can also form a hierarchical topology, using relays (mobile or fixed) to transfer data to the sink. In this topology, clustering is used to group the nodes. (Akyildiz et al., 2002).

2.5 PROTOCOL STACK OF WSNS

Fig. 2.7 shows a WSN stack protocol (Akyildiz et al., 2002), which is similar to the mobile *ad-hoc* network (MANET), in terms of internet protocol (IP) and transmission control protocol (TCP) network protocols. The main difference between the stack protocol and the aforementioned is that mobility, task management and power planes operate across all network layers. The layered approach has been accepted in MANETS for a length of time, however, due to the high application specificity of WSNs, it has been found by researchers to be difficult to apply this type of protocol in WSNs (Garcia-Macias and Gomez, 2007). As a result, layered network structures are often not the best way to optimise resources and assess a wide application range (Shakkottai et al., 2003).

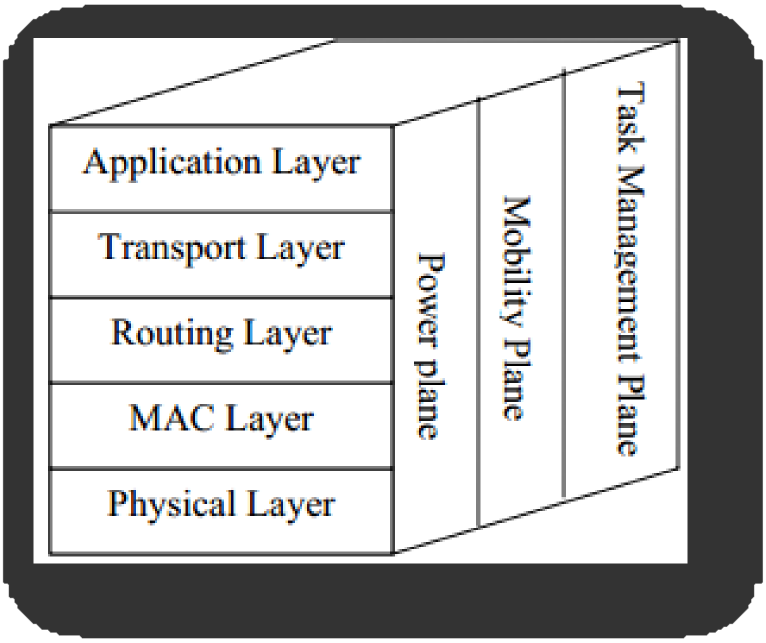

Fig. 2.7 Protocol Stack for WSNs (Matin and Islam, 2012)

This protocol stack comprises several layers, including the physical layer, medium access control (MAC) layer, routing layer, transport layer and application layer. These layers are described below:

- Physical layer: In harsh environments, robust transmission can be gained through the physical layer; which co-ordinates low-level radio interface operations that include coding, signal detection, modulation, frequency selection and transmit power. Coding and signal detection are hardware related competences like memory size and processor speed. Frequency selection control the potential ranges of wireless transmission media through radio frequency. According to the industrial scientific and medical (ISM) band, the recent frequencies used in WSN include the higher 2.4 GHz of band and the 915 MHz via IEEE 802.15.4 or IEEE 802.11 (Baronti et al., 2007). An optimum selection of a modulation scheme is a significant factor amongst sensor nodes in order to achieve an accurate information delivery. As stated by Panichpapiboon, transmit power defined as “the minimum transmit power used by all nodes necessary to guarantee network connectivity” (Panichpapiboon et al., 2006).

- MAC layer: The MAC layer controls channel access between transmitters. It switches off the radio when a sensor is not active or not receiving data, which in turn saves energy. One of the main objectives of MAC is to share the common communication resources or medium among various sensor nodes in an efficient and fair manner. Therefore, a good network performance can achieve in terms of energy consumption, throughput, and delivery latency. (Zhong et al., 2001).

- Routing layer: In a multi-hop network, the routing layer addresses nodes and routing. In MANETs, the use of unicast and multicast systems cannot be applied in WSNs, due to the routing being in different forms, including one-to-many and many-to-one (Bhagwat et al., 1996). Routing layer in WSN is responsible for routing the sensed data by a sensor node (source) to the base station node (destination).

Generally the sensor source node can transmit the sensed data to the base station either directly through a single hop or through multi-hop wireless communication (Gomez and Garcia-Macias, 2006).

- Transport layer: The transport layer addresses packet delivery; it can control congestion both inside the transport layer, where the transport layer is in charge for reliable end-to-end data delivery between sensor nodes and the base station. MANET traditionally implements the TCP/IP protocol stack, meaning MANET nodes will have IP addresses to support broadcast, unicast and multicast routing mechanisms. Due to the energy and storage restrictions of the sensor nodes, such mechanisms cannot be applied directly to sensor networks without modification. The communal development in the research community consider the nature of WSN totally different in comparison to the traditional TCP/IP networks in MANET, thus transport protocols should refrain from the TCP concepts (Zheng and Jamalipour, 2009).

- Application layer: Software created for general purposes, and given an associated class, can be used in several different applications in WSN, to reduce the time required in prototyping. This assignment of classes is created by the application layer (Murthy & Manoj, 2004).

- The power: Power-awareness is an essential consideration to be taken in and across each layer in WSNs. This power-awareness is recognised by the 'power plane' so that if a sensor detects activity, it may keep its radio on, or switch it off if it did not belong to an active route due to it not generating data. Sensors can save energy in this way, if it is running low in energy it can turn off its radio and thus save power.

- Mobility: In the event of sensor mobility, the mobility plane becomes responsible for the maintenance of sensor network operation. Although most sensing applications are static, there is a strong likelihood that mobile sensing applications will soon be in use, for example in systems where the sensor is placed on moving objects such as animals, cars etc. Mobile network routes need to be repaired periodically because they have a limited lifetime, due to node mobility. Similarly, routes may need to be changed in the event of nodes running out of power.

- Task Management: This plane is responsible for the co-ordination of all nodes, while maintaining power-awareness during normal operation. In a situation where there is sufficient sensing-redundancy from other sensors in a given region, some sensors within that region may turn off. Task management is an assists for the sensor nodes which coordinate the sensing task and also helps to lower the overall energy consumption (Matin and Islam, 2012).

2.6 RELATIONSHIP TO WIRELESS *AD-HOC* NETWORKS

WSN can be considered a particular type of Mobile *Ad-hoc* Network (MANET). Large volumes of evidence regarding MANETs are being increasingly used by researchers to apply this evidence to WSNs. This is because of the auto-configurable nature of node networks in both MANETs and WSNs, the nodes of which are connected wirelessly. Similarly, current resources in connecting these wireless links are few, and commonly used networking algorithms and protocols are no longer adequate. In applying algorithms and protocols to WSNs, it is important that considerations are taken if initially designed for MANETs. Despite the similarities between WSNs and MANETs, it can often still be debated that WSNs may be thought of as an entirely separate field of research. The similarities and differences between the two network types are described below (Jindal et al., 2011; Garcia-Macias and Gomez, 2007).

2.6.1 Similarities

The US military were the main customers of both WSNs and *Ad-hoc* networks, when they were developed in the 1970s. Recently, advances are obvious as *Ad-hoc* technologies are now being used in commercial applications (Garcia-Macias and Gomez, 2007). The newer, commercial applications are very different from their initial use in the military, and as a result, need to be addressed with a different approach. For example, we can no longer use considerations such as cost awareness, scalability, single purpose-applications and radio application, in the same manner to which they were applied in *Ad-hoc* networks, for applications such as human context interaction, disaster recovery, peer-to-peer, WSNs and so on.

One of the key reasons for the resemblance of WSNs to *Ad-hoc* networks is because of the wireless distribution of both network types (Römer and Mattern, 2004). In essence, neither displays a distinct network infrastructure. Similarly, multi hop routing is often used in both, where intermediate relay nodes are required for routing between two individual nodes. Another factor is that sensor nodes and *Ad-hoc* nodes are both powered by batteries, which raises the issue of power consumption and designing ways in which it could be minimised in both networks. Another similarity between the two networks is the use of wireless channels in both, which is susceptible to interference by other radio technologies which operate in similar frequencies; due to the wireless channel being in an unlicensed spectrum. Because of the similarities in the distributions of both networks, self-management is essential in both.

There has been a recent return in *Ad-hoc* networks, due to two main considerations; the first being the recent advances in technology allowing the production of more powerful, yet smaller mobile devices. The second consideration is the recent advance in the types of *Ad-hoc* applications that are becoming available. It has been stated that the core network internet paradigm should remain as simple as possible, as it only considers data packet delivery. However, this intuition no longer suits commercial *Ad-hoc* networks. This issue is controversial, however, as it has since been debated that *Ad-hoc* networks differ greatly from traditional WLANs. Nowadays, the emergence of newer applications is changing *Ad-hoc* networks, to networks with a greater capacity for networking storage, and processing (Garcia-Macias and Gomez, 2007).

2.6.2 Differences

Despite the similarities between MANETs and WSNs, a few central differences set them apart from each other. These differences are based on the differences in the natures of both networks, for example, nodes used in MANETs are usually devices that are designed for human use, such as mobile radio terminals and laptops, while WSNs have an increased application in an environmental context. This is emphasised by the fact that WSN nodes are deployed into the environment to carry out sensing tasks, and as a result, the number of nodes that are deployed is often greatly larger than that of *Ad-hoc* networks. This requires important consideration of scalability (Dash et al., 2012).

In cases where nodes are deployed in the environment, for example in the ocean or on a volcano, it is common for nodes to become damaged or even completely fail. As a result, network topology can be affected. However, in *Ad-hoc* networks, the network topology is usually affected by node mobility. Thus, there is an increased requirement for reconfiguration techniques, such that the design of the network considers nodes that are likely to fail. Also, some of the issues that need to be considered in mobile networks, such as nodes that are placed in or onto moving objects for sensing applications, are not important for consideration in WSNs (Younis et al., 2004).

Aside from the failure of nodes, network topology is often affected by protocol cycles that allow the system to sleep/awake on demand, in the context of WSNs. These sleep/awake protocols are designed to minimise power consumption; a big limitation in WSNs. As nodes are often left unattended at length, often months and even years while running on battery power, communication ranges are often low (a few metres) and the speed of communication is also often low (Jindal et al., 2011; Garcia-Macias and Gomez, 2007).

2.7 CLUSTERING IN WSN

2.7.1 Overview

Clustering defines the arrangement and grouping of dynamic nodes in wireless sensor networks. Nodes are collectively grouped based on their proximity to each other, which in effect, also groups them based on their transmission ranges. Nodes that are within close proximity with another are able to form a bidirectional link to another node. The control frameworks of single and multi-hop clustering are determined by the diameter size of a cluster, and the distance between a node and the CH is taken into account. For example, each node is no more than one hop away from a CH, in single-hop clustering. Therefore, in one cluster, all nodes would only be (at most) two hops away from each other.

Conversely, multi-hop clustering arrangements demonstrate a restriction in the proximity of one node to another. In essence, nodes do not have to be in close proximity to the CH and as a result, they can be several k-hop distances away from the CH and still form a cluster (Angione et al., 2007). A WSN nodes in flat and cluster structure is shown in Fig. 2.8.

Individual nodes of a WSN are shown as small circles, joined by lines, which represent the single hops that link the nodes wirelessly. Nodes are numbered from 1 to 16, and in the case of Fig. 2.8(A), each node has the same capability as a router to forward data packets to all other nodes in a WSN expressing a flat architecture.

According to (Akyildiz et al., 2002), a better protocol performance of the MAC layer can result in increased scalability, power saving and spatial use. Clustering helps with the routing within the network, allowing for a smaller routing table, as well as overheads of transmission. As each node can only store small amounts of data at a given time, clustering helps to improve the overall topology of the network. (Er & Seah, 2004). Described below, are three types of nodes which play different roles in a clustering network.

CH nodes:

Connected CH from the 'backbone' or the support of network operation, functioning to control all applications within the network, including power control, routing, channel access and bandwidth allocation. (Ohta et. al, 2003). Through their role, CH nodes control the overall efficiency of the cluster, as they connect directly to each other or indirectly through gateway nodes. The CH also have the role of forwarding information internally between nodes. Firstly, an ordinary node sends packets to its CH, and the CH can then send the packet back. CH can also route the packets to a different CH which is connected to a receiver, which then directs it again to its end point. (Chen & Liestman, 2003). CHs therefore play essential roles in the stability of a network.

Cluster Gateway Nodes:

A gateway node forms a connecting terminal for individual CHs. As shown in Fig. 2.8 (B), node 2 is in the transmission region accessible by two CHs. As a result, it forms a gateway for the two clusters. Node 5 and node 6 in Fig. 2.8 (B) can reach a CH in two hops, but they also immediately neighbour CHs. This forms a gateway that links to another gateway in a separate cluster, overall facilitating inter-cluster communication. (Purtoosi et al., 2004).

Ordinary nodes (cluster member):

Ordinary nodes are members of a cluster that form a network, and function independently of other nodes in neighbouring clusters.

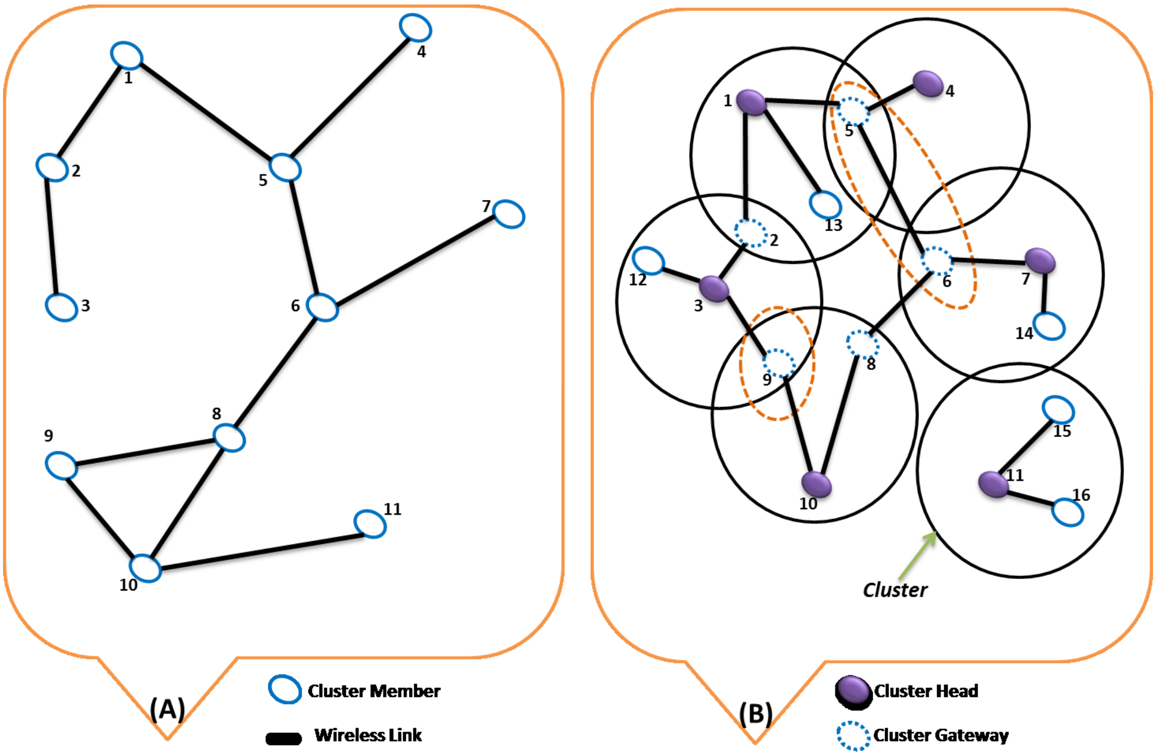

Fig. 2.8 Nodes in flat and cluster structure. (a) Flat structure. (b) Cluster structure

2.7.2 Clustering Objectives

The implementation of WSNs still requires further optimisation in order to overcome certain challenges involved in their application (Amis et al., 2000). For example, for WSNs to become the major network infrastructure, clustering algorithms are required to be accurately designed, such that they fulfil their role in allowing the communication of mobile devices to each other wirelessly, such that the connection is reliable and does not waste energy in the process (Baker and Epheremides, 1981).

Cost of Clustering

When WSN is applied in a large-scale, it's an indication that the clustering cost involved will have effects on the whole network cost. Increasing the number of nodes will increase the tasks involved for the cluster include processing and communication tasks between the nodes, which are. As a result, it is important to optimise the design of clustering networks such that they do not make network application large as resources are often expensive to run (Chen et al., 2013). It is also important to recall that as network size increases, communication requirements also increase and as a result, the cost increases. A compromise therefore needs to be maintained between scalability in clustering and the bandwidth available for data transmission.

Load Balancing

CHs routinely process data tasks as well as intra-cluster tasks. In order to maintain performance goals, it is required that the CHs have a balanced load, which is achieved when nodes are evenly distributed. Maintaining this balance is a challenge in WSNs, and maintaining clusters of equal size would save energy and therefore prolong the lifetime of the network. This is preferable over a system in which high rate CHs are employed, in which energy would diminish very quickly (Gayathri et al., 2007).

Clustering Formation

Although the ‘clustering formation’ initiative would be of a great advantage in the context of a WSN, its construction requires careful consideration. Clustering can be applied in several application types and as a result, there may be different requirements in the arrangements of nodes, including their sizes and variables used in configuration of the network (Yang & Zhang, 2007).

Real-Time Operation

In the design of WSNs, it is important to consider the optimal data lifetime. In certain cases, simply receiving the data is enough for its analysis and a delay in the receipt of this data is not a major problem (Mainwaring et al., 2002). However, in other cases it is essential that data is received in a real-time setting, for example in emergency services applications or military tracking (Chlamtac et al., 2003). Algorithms designed in these networks must therefore consider the context and application for which it will be used to ensure the QoS and that data are actually transmitted with the minimum delay.

Maximising Network Longevity

Applying WSNs in harsh environments requires nodes to have a reasonable lifetime. Compared to other nodes, CHs have a greater requirement for resources and as a result, it is imperative that the energy requirements in intra-cluster communication networks are reduced (Al-Karaki et al., 2004). This is often achieved when CHs are placed closer to the majority of nodes in its cluster or by maintain the load balance. Similarly, network longevity can often be attained through a combination of clustering and route setup (Younis & Fahmey, 2004).

Repair Mechanisms

WSNs are prone to failure in cases of node mobility, interference, and death of the nodes. It is therefore essential that clustering schemes consider mechanisms for recovery in these cases. In a case of the imminent death of a node or failure of the system, there must be a restoration mechanism or warning, such that reliable data communication can be recovered.

Synchronization

Slotted transmission schemes allow the schedule of sleep intervals of nodes, in order to reduce their energy usage. An example of these schemes is TDMA, which utilises mechanisms involving synchronisation to maintain the schedule. In clustering contexts, this synchronisation can greatly increase the lifetime of a network and as a result, its performance.

Data Aggregation

WSNs have the advantage of allowing data aggregation within the network. It is often found that dense networks contain several nodes that sense similar information. Thus it is an important to differentiate between useful sensed data which can be facilitated by data aggregation. In several sensor networks, in-network processing allows this mechanism and has become more commonly utilised. The ultimate advantage of this system is to minimise the total amount of data transferred. In the light of the above, when designing the cluster network methodology and the use of it, the capacity for data aggregation should therefore be considered (Intanagonwiwat et al., 2003).

Quality of Service (QoS)

It is important to consider the efficacy of WSN communication overhead requirements, to oversee QoS. Mobility of nodes often leads to changes in the network topology of hierarchical structures, for example in the cases of cluster additions or deletions. It has been found that a 'virtual backbone' (VB) is required in WSNs, which consist of nodes that are dynamically selected to control the transfer of messages. This causes a partition allowing virtual domains which each have their own VB. The creation of this divide allows a reduction in overall costs, as only the VB nodes and QoS information would be shared between nodes in one virtual domain (Chlamtac et al., 2003). The implementation of this, however, varies between applications in their metric requirements, which should be considered in its design.

2.7.3 Advantages of clustering

A key advantage in using cluster WSN architecture is the ability to connect many mobile terminals, in order to ensure efficiency of the system. Other advantages that build on this are described below:

Aggregation of Topology Information:

This system causes only one node to be required for storage of a small percentage of the network's routing information. This is because the total number of nodes in one cluster is fewer than the total number of nodes in the network. As a result, clustering may contribute to the efficiency of aggregating the topology information (Chinara & Rath 2009).

Efficiency and Stability:

Cluster structures tend to make WSNs appear smaller and therefore each mobile terminal appear more stable. Using the efficiency and stability system, mobile nodes that switch to neighbouring clusters allow information to be transferred, or data to be modified via mobile nodes that reside in corresponding clusters only (Mai et al., 2009, El-Bazzal et al., 2006).

Communication Coordination:

Communication coordination reduces excess data transfer between mobile nodes, through the limiting of clustering between inter-cluster communications.

Routing Efficiency:

Routing efficiency has been considered an alternative for improvement of MAC layer efficiency, as well as that of the routing process. This may be done by reducing the amount of message flooding which inevitably occurs in a MAC layer system (Sucec & Marsic, 2004).

Spatial Reuse of Resources:

To improve and increase the storage capacity of the system, information can be stored via the CH, allowing the re-use of resources. If they are not corresponding clusters, two clusters may distribute information over one frequency or code set. This is due to the non-overlapping nature of the multi-cluster structure. Similarly, specialised mobile nodes residing in a cluster may assist transmission from a CH, as a result of improved co-ordination. This system allows the preservation of resources used in retransmission (Tolba et al., 2007).

2.7.4 Classification of Clustering Algorithms

Algorithms used in WSN clustering can be differentiated in several ways. An early example of such classification is that between ‘static’ and ‘dynamic’ clustering. Formation of a cluster is considered to be dynamic if it drives CH reorganisation or cluster re-election when reacting to changes in network topology, or identifying optimal CH rotation between nodes. Dynamic architectures tend to present more efficient use of WSN sensors, which results in optimal management of energy consumption, thus contributing to the prolonging of the lifetime of the network.

WSN clustering algorithms are further divided into the following categories, which differentiate them based on their criteria for cluster formation as well as CH election parameters (Mamalis et.al, 2009):

- Probabilistic (random or hybrid) clustering algorithms
- Nonprobabilistic clustering algorithms

Probabilistic Clustering Approaches

Over the past decade, there has been a dramatic increase in the need for efficient WSN usage (Letswamotse et al., 2017). To overcome this, there has been an increase in the development of specific clustering protocols, including the improvement of scalability, energy consumption distributions, and network lifetimes. Some of these approaches were based around WSN clustering protocols, such as energy efficient heterogeneous clustered scheme (EEHC), hybrid, energy-efficient, distributed (HEED) and low-energy adaptive clustering hierarchy (LEACH). The main objective of these approaches aimed to prolong the lifetime of the network, and keep energy consumption to a minimum (Afsar and Tayarani, 2014).

The main approach used in probabilistic selection clustering algorithms the initial CH is determined through assigning *a priori* probability to each sensor node. These initial probabilities act as primary criteria such that individual nodes can then decide whether or not to be elected as CH (Heinzelman et al., 2002). This process is required to be as uniform, flexible and well distributed as possible. A secondary criterion can then be decided on during election of the CH, or during the formation of the cluster, with the goal of achieving optimal energy consumption and prolonging the lifetime of the network. Aside from obtaining a high-efficiency network, nodes in probabilistic techniques usually have the same capabilities, therefore CH election and formation process is the most appropriate technique. As a result, clustering algorithms tend to have fast execution convergence times, as well as reducing the number of messages exchanged between nodes. (Jin et al., 2008, Demirbas et al., 2004).

Nonprobabilistic Clustering Approaches

An alternative to probabilistic algorithms is the ‘deterministic’ approach, which uses different criteria to determine CH election, mainly taking into consideration the proximity between nodes and information transferred between nodes that are in close proximity to one another (Liu et al., 2005, Yi et al., 2007). Cluster formation in these WSNs considers the communication between either one or multi-hop neighbouring nodes. Generally, this approach is considered to be more complex than probabilistic clustering algorithms due to there being a higher intensity of message exchange and in some extent graph traversing. Because the cluster formation process is generally based on the nodes communication with their neighbours wither one or multiple hops neighbours.

However, these algorithms are often more accurate and reliable in extracting well-balanced clusters. Some nonprobablilistic algorithms take into account the transmission power, mobility and node proximity in the system in order to achieve multiple goals as opposed to protocols that follow a single criterion (Ding et al., 2005, Virrankoski et al., 2005).

CHAPTER 3:

LITERATURE REVIEW

3.1 POPULAR ROUTING PROTOCOLS

Wireless sensor networks (WSNs) are becoming increasingly important in many application areas. Several application scenarios require connectivity between WSNs and the Internet. Although WSN typically is not IP-enabled, connection to the IP network makes it easy to monitor sensors worldwide. The appropriate choice of a decent routing protocol in a WSN is an essential object due to the fact that WSN performance is significantly influenced by the type of the routing protocol used for the network. An overview of the main three traditional routing protocol used in WSN is presented in order to have a good understanding of the recent studies where most of the contribution build their work on the exiting routing protocol. DSDV, DSR, and AODV are some of the most popular routing protocols which are already built in most of the simulation software such as OPNET, NS3 and NS2.

DSDV ***"Destination-sequenced distance vector"*****:** is a proactive table driven protocol based on the Bellman Ford routing algorithm (Perkins and Bhagwat, 1994). Each node would have a routing table that contains information such as route oldness, shortest distance and path to every other node in the network, therefore routes between the nodes are always updated and well maintained. Neighbouring nodes in DSDV exchange the routing messages between themselves, which eventually updates their routing tables accordingly. Packets can be received until the sender node receives a route-replies message from the destination node, which gives the routing a feature of being able to catch the messages with unknown destinations. A memory with maximum buffer size is used to collect the packets which are waiting for routing information. These packets can be dropped off when they reach a size beyond the actual memory size. A sequence number is allocated by the destination for each packet, these numbers are usually an odd number unless presented with a link then it will be an even numbers (Perkins & Das, 1997).

DSR *"Dynamic Source Routing"*: is an on demand routing protocol which is used for a multi-hop networks based on the source routing concept. Networks on DSR can be totally self-configured and organised where the network does not require any existing network infrastructure. DSR works to reduce the network bandwidth overheads, as well as escaping from the large routing updates by not using the periodic routing message which is the case for AODV. In order for nodes in the network to discover and maintain source routes to random destinations DSR uses two mechanisms which is Discovery and Route Maintenance. Protocol in DSR can be exposed for a range of handy optimizations due to the unique mechanism that DSR uses when having the route as part of the packet itself (Johnson et al., 2007).

AODV *"Ad-hoc On-demand Distance Vector Routing"*: is one more of the traditional distance vector routing algorithms, which has inherited its main unique mechanism from both DSDV and DSR routing protocols. AODV share the on-demand feature with DSR, which is the route discovery process; however AODV still implements the classical routing tables which are an entry for each destination unlike the DSR where there are multiple route caches for each destination. AODV shares the loop free routes with DSDV, and does not involve in global periodic routing like DSDV. Furthermore, AODV does not involve in causing overhead to the packets when a route become available between sources to destination. AODV has the capability of providing the unicast, multicast as well as the broadcast communication, where a broadcast is used for a route discovery and unicast is used for a route reply massage (Das et al., 2003).

Current research emphasizes on the generation of algorithms that manage functionality and lifetime, as well as being robust and reliable. It has been well documented that convergence time variation is a more efficient system to control cluster formation (Mamalis et al., 2009), than constant time systems. Described below, are some common algorithms used in convergence time variation. These algorithms are divided into the categories of Variable time convergence (Probabilistic) (Abbasi and Younis, 2007), Constance convergence time (Non-probabilistic, Mamalis et.al, 2009), Location base, and weighting base clustering algorithms depends on their criteria for cluster formation as well as CH election parameters (Abbasi et al., 2007)

3.2 VARIABLE CONVERGENCE TIME ALGORITHMS

Low Energy Adaptive Clustering Hierarchy (LEACH)

LEACH is a dynamic clustering protocol used in WSNs, applying a hierarchical, distributed, probabilistic and one-hop concept. It functions to improve the lifetime of the network by reducing energy consumption in individual nodes. This is through a reduction in communication messages required, by carrying out data aggregation. Through the LEACH protocol, nodes do not require a centralized control to make autonomous decisions. As a result, every node is capable of becoming the CH to maintain a balance in energy distribution over the nodes as a network. This balancing of energy distribution over the network makes LEACH an efficient protocol, as it provides higher scalability in formation of clusters, as well as enhancing the network lifetime. A drawback, however, of the LEACH protocol is that due to the CH election being probabilistic, low energy nodes may be selected as CHs, and as a result, some nodes will not have a CH in their range. In addition to this, LEACH functions to form intra- and inter-cluster topologies, which limits its use over large regions (Heinzelman Ding et al., 2002).

Energy Efficient Hierarchical Clustering (EEHC)

(Bandyopadhyay and Coyle, 2003) described an algorithm based on randomised clustering for WSNs. In order to maximise the network lifetime, EEHC ensures connection of the CH to the base station, and works in two stages; initial and extended. During the initial stage, known as ‘single level’ clustering, a CH node announces itself as a CH by sending a message named as ‘volunteer CH’ to all nodes in the range. Non-CH nodes that receive this message (either directly or by forwarding) become members of the nearest group. Nodes are announced to be CH by the nodes that are outside the area are called ‘forced CH’. The second stage of EEHC is to allow multi-level clustering. This is the extended stage, which is based on hierarchical clustering, where the CH in the lowest level transmits the data to a CH in the next level, until it reaches the top level, where it transmits the data to the base station. EEHC has time complexity, which is an advantage over other algorithms such as LCA, making this algorithm suitable for networks with a large number of nodes. The EEHC algorithm results in minimization of energy spent in communicating the information to the information processing centre thus reduce the energy consumption (Bandyopadhyay & Coyle, 2003).

Linked cluster algorithm (LCA)

LCA was one of the very first clustering algorithms developed by Baker and Ephremides. This algorithm focussed on connecting the CH directly with all nodes in the cluster and designed a network that can handle the mobility of the nodes. Nodes in such algorithm are usually allocated with a unique ID number. There are two ways for the node to become the CH; either by having the highest ID in the set including all neighbour nodes and the node itself, or when none of the neighbours are CH (Baker and Ephremides, 1981).

Hyper Energy-Efficient Distributed Clustering (HEED)

HEED is also a distributed, hierarchical clustering scheme protocol through which single-hop communication is transferred within a cluster. Nodes are selected as CH based on the following parameters; intracluster communication cost and residual energy. The intracluster communication is used to define the proximity of one node to another, and to define whether or not a node should join a particular cluster. Residual energy probabilistically decides the election of initial CHs. In contrast to LEACH, nodes are not randomly selected, but are chosen based on their higher residual energy to become CHs. As a result, the CH nodes are distributed evenly over the network. In addition to this, clusters that are chosen communicate with CHs yielding low costs of intracluster communication. Although HEED outperforms LEACH in the efficiency of prolonging the network lifetime, it requires synchronisation and as a result, in larger-scale networks, large amounts of energy are consumed for data transmission between CHs that are far from each other (Younis and Fahmy, 2004).

Adaptive clustering

This algorithm was developed by Lin and Gerla to support the multimedia in a multi-hob mobile *ad-hoc* network based on using CDMA. Nodes are also assigned with ID numbers. This algorithm takes the advantage of balancing the interest in the separation and spatial reuse of channels to control the cluster size. The way to increase this is to group the nodes into small clusters and data delivery delay, i.e. this algorithm avoids larger clusters. Adaptive clustering is similar to the LCA and TDMA (Time-Division Multiple Access) in the method used for intra-cluster communication. Furthermore, the adaptive clustering is similar to the TDMA but with less implementation complexity. However, each cluster in this algorithm requires a different code resulting in simplified implementation (Lin and Gerla, 1997).

Scalable self-configuration and self-healing (GS3)

It has been previously described that in a large network, the geographical location of clusters in a network should not be ignored. Zhang and Arora designed the GS3 algorithm, for self-configuring the wireless network into a virtual hexagonal structure. It is used as a guide for the grouping and distribution of the nodes. This algorithm uses the radius as the cell measurement size. Where the size increases, the energy consumption is said to increase, as well as the reliability. There are two kinds of nodes in this algorithm; big and small nodes. The big nodes initiate cluster formation and interface the small nodes to other cells in the network. The big nodes will select the heads of its neighbour cells and the remaining nodes will become cell members. GS3 offers one-way communication between nodes, and guarantees a certain number of CHs in the system, unlike other algorithms where they use more than one way (message) of communicating. Furthermore, GS3 is a static network suitable for mobility, because it is a self-healing system (Zhang and Arora, 2003).

WSNs clustering based on semantic neighbourhood relationships

Rocha has proposed a new way of clustering based on a fuzzy interface system which groups the sensor nodes that are semantically related into a cluster depending on the semantic of collected data. The semantic neighbourhood cluster has an efficient effect on the network data aggregation process. This helps in reducing transmissions, where most of the semantic relations between sensor nodes are used, by aggregating more similar data and operating nodes that are semantically unrelated, in a low duty cycle. This cluster algorithm overcomes the usual algorithm issues where neighbouring nodes sometimes do not cover relevant areas or provide measurements that are not semantically related. Consequently, this results in a reduction in the communication resource usage, reducing the energy consumption and improving the data accuracy. The algorithm proposed here obtained about 70% of energy savings compared to LEACH algorithms and 47% compared to deterministic clustering, thus proved itself as an efficient clustering method for the WSNs (Rocha et al., 2012).

Time Controlled Cluster Algorithm (TCCA)

This algorithm is similar to LEACH, where the selection of a CH is based on the residual energy parameter, as well as applying a probabilistic approach to distribute the load among sensor nodes. Based on the residual energy and present probability, each node can decide whether or not to elect itself as a CH. TCCA has the advantage of applying both HEED and LEACH approaches, where the HEED energy fraction contributes to TCCA, along with the CH election technique used in the LEACH algorithm. Similar to HEED, a node can announce itself as a CH to all other nodes, by simply sending a message which includes the ID of the CH node, residual energy as well as 'time left to live' (which is based on the energy) in order to control cluster size (Selvakennedy and Sinnappan, 2007).

Distributed Multi-Hop Overlapping Clustering Algorithm (MOCA)

MOCA is a distributed, probabilistic algorithm that was designed to organise sensors into clusters that overlap. This technique was introduced to improve network issues such as node localisation, topology discovery, intercluster routing and recovery from CH failure. This algorithm specifies the distance within the cluster range, assigning it the number of hops as 'k'. This algorithm also assumes that a CH is formed from a node with the probability 'p', and all CHs then advertise themselves to nodes that are within their range. Nodes that wish to join a cluster will send a message to all CHs. The CH will then respond and the node will send the ID of all CHs that responded. This notifies the nodes that the IDs given were those of boundary nodes. This was supported by evaluation work, where the 'p' value was used to control the number and overlapping of nodes within a cluster (Youssef et al., 2006).

Energy Efficient Clustering Scheme (EECS)

Using this scheme, a certain number of CHs are chosen, and their election is therefore based on two parameters; a localized competition process, and the residual energy. In the first scenario, high energy nodes compete for the position of CH, by announcing their energy levels amongst other nodes. Therefore, the node with the highest energy level will become the CH. Following this, cluster formation can take place based on the proximity between the BS and all other clusters. This overcomes the issue that the further the BS from the cluster, the higher the energy transmission required, which can in turn prolong the network lifetime and save energy consumption (Ye et al., 2005).

Random competition based clustering (RCC)

RCC first was designed to serve the mobile *ad-hoc* network then developed to be used with the WSNs. In this algorithm, the node to become a CH must be the first node to broadcast a claim to become CH in the same radio coverage. Once all the neighbouring nodes know about this claim, each node will join the cluster as a member and that means that they give up their rights to be CH. In RCC, the current CH only gives up its CH position when another CH moves near to it, unlike other clusters where the CH gives up the CH position once a node with a lower ID is detected. Furthermore, there is evidence that RCC is more stable than others such as adaptive clustering (Xu and Gerla, 2002).

CLUBS

The CLUBS algorithm was proposed by R. Nagpal and D. Coore. The principle of CLUBS by which a node can become a CH is to choose a random number from a fixed range of numbers. If no neighbouring nodes interrupt, the node counts down from the random number until it reaches zero. Once it reaches zero, it broadcasts a message; 'recruit' to all neighbours, announcing itself as a CH. The node that joins this cluster is called a 'follower' which automatically gives up the CH position after joining. In CLUBS, the node can be a follower of more than one CH (groups) i.e. it allows overlap. In order to terminate, the CLUBS algorithm requires all nodes to be connected to a cluster, either as CH or as a follower (Nagpal and Coore, 1998).

3.3 CONSTANCE TIME CONVERGENCE ALGORITHMS

Reconfiguration of cluster heads for load balancing in WSN

Recent studies have introduced different reconfiguration algorithms, and have subsequently been categorised into two main groups; self and forced reconfiguration, depending on which node was to be allocated as the CH. In a self-reconfiguration, a CH can be selected in two ways, either by having a table for the neighbour nodes or by using probability functions. In the forced reconfiguration, the current CH can determine a new CH from the general nodes (GN) based on certain conditions such as a node with the most remaining energy. N. Kim developed an algorithm based on forced reconfiguration, in order to prolong the network lifetime with an optimum distribution of CHs in the range. The cluster area can be divided into many sub regions depending on the number of nodes in the cluster, where the CH can choose the next CH from different sub regions. Hence the CHs can be reasonably distributed which will result a balance of the GN number in the clusters and this lead to low energy consumption and increase in the network lifetime. This work was compared with LEACH and ET-driven, (Energy-Threshold-Driven Based Reconfiguration), algorithms and proved to perform better, as well as remain simple and effective tool for data aggregation in the WSNs (Kim et al., 2008).

Ring-structured Energy-Efficient Clustering Algorithm (RECA)

During the early stages of cluster formation, the RECA algorithm groups sensors into clusters that are bridgeable by one-hop. At this stage, the expected node number is estimated based on the following parameters; overall number of nodes in the whole network, the geographical range covered by the network and finally, the lowest range of transmission. The node has the option to be either a CH or a member of the cluster. Any node that has not declared itself as a CH nor a member of the cluster, assigns itself a number between 0 and 1, and this number is compared to a threshold value that is based on the location number of the node. If the assigned value is lower than the threshold number, then the node becomes a CH and declares this to the remaining nodes in its proximity. If the assigned value is higher than the threshold number, the node will wait for announcements from other CHs, and joins the cluster that possesses the best signal-to-noise ratio. This continues until all nodes have been assigned a role of either CH or cluster member. This algorithm gives each node the option to be a CH based on its geographical position. This work was supported by simulations and compared to both LEACH and HEED, and it was shown to be more energy efficient and capable of prolonging the network lifetime (Li and Znati ,2007).

A self-Organized Communication Efficient Clustering Algorithm in WSN

One of the essential strategies in WSN is to organise large multi-hop wireless networks into clusters in order to obtain scalable architectural and management strategies (i.e. CH). The Self-Organised Communication Efficient Clustering Algorithm was design by M, Moghadam to tolerate failure and mobility in the network and it is a self-organised algorithm, where sensor nodes do not need to have any information about the network topology. All sensor nodes can become a CH and move, replicate or reposition themselves in a self-organised manner. Data transaction of the node is used to decide the need for generating, migrating or dissolving a cluster. The mathematical algorithm behind this protocol is based on the idle rule, migration rule, replication rule, CH advertisement and forwarding rule, cost efficiency and stability rule. On the whole, locating the optimal number and places of the CHs will reduce the communication cost of the network and data aggregating between nodes. Consequently, the network lifetime and scalability is guaranteed. This algorithm was compared with LEACH and ACE (employs and emergent) algorithms and showed better performance during the steady state phase (Moghadam et al., 2011).

Multiple Parameter Based Clustering (MPC)

(MPC) Multiple Parameter based Clustering was proposed by Khan Asif as a new method for wireless sensor network clustering; MPC was embedded with the traditional k-means clustering algorithm. Furthermore, MPC was anticipated to take variation of parameter into account to form clusters such as node energy level, distance from base station, received signal strength indication (RSSI), and latency. Having different parameters gives the cluster more power to avoid poor node distribution cluster or highly dense clusters, as well as minimum and maximum the range of nodes in every cluster. These parameters can also cause the network consume less energy by adding the right parameter which supports low energy consumption (Khan et al., 2012).

A New Clustering Protocol for Wireless Sensor Networks Using Genetic Algorithm Approach

Recent cluster researches are focusing on presenting algorithms with the least cost possible, which also includes the receiving and transmission between all CHs and the base station. Genetic Algorithm (GA) is considered as a global search heuristic algorithm and is categorized as a dynamic technique algorithm in order to find the optimum states which can be estimated by generating different parameters. The fundamental principle behind the GA mathematical formula is that an increase in coverage is benchmarked in opposition to the network lifetime. Norouzi adapted a new cluster using the GA based on taking the average used energy amount of each node, using the ratio of total energy consumptions to the total distances of all the nodes, thus, the lifetime of network will be extended. Also, TDMA schedule was used in every cluster in order to ensure that all sensors radios are off most of the time and only activate when there is a need to transmit data. All activations are logged in order to keep a track of the energy level, number of nodes still alive and number of transmissions. This algorithm was demonstrated using simulation in comparison with LEACH protocol and shows efficient performance in energy consumption and network lifetime (Norouzi et al., 2011).

Efficient Clustering for Improving Network Performance

Recent studies have put effort and consideration to the performance of multi-hop networks, providing reliable communication and a reduction in energy consumption. T. Anker has addressed this issue and presents an efficient clustering for improving network performance using the Belief Probabilities algorithm (BP) where the power balance of the nodes was well thought out, as well as the total transmission power aggregated in the multi-hop routing. BP characteristics are fast convergence, accurate results, and good performance in an asynchronous environment, making it an efficient and popular technique for distributed inference. This algorithm chooses the CH founded on a unique set of global and local parameters under the energy boundaries and network performance. The network lifetime is measured by the period of time that the network is available and working properly to provide services, where some others only measured the network lifetime by the time that the first or the last node dies. Using simulation this algorithm has showed an increase of 40% in compare with HEED scheme in terms of data collection quality, network reliability and transmission costs (Anker et al., 2008).

Power Aware Routing Protocol (PARP)

In the last few years, there has been a significant amount of research studies in the field of power aware algorithms. The power aware routing protocol (PARP) was proposed by Rangaswamy to achieve optimum QoS i.e. delay, packet delivery ratio, and reduction of the power consumption of the WSN. PARP uses link quality estimation along with power aware, and the mathematical algorithm based on the concept that a node can decide on the next hop depending on its own position, neighbours and perhaps on the destination node. The PARP protocol allows the network to choose the optimum node of the grid, in the case that the current optimum node is no longer available. As a result, PARP reduces power consumption and delay as proved in the simulation with comparison with the AODV protocol. However, despite this, there was an observed increase in the packet delivery ratio (Rangaswamy and Rangaswamy, 2012).

3.4 LOCATION BASED ALGORITHMS

An Average Distance Based Self-Relocation and Self-Healing Algorithm

Relocating mobile sensor nodes is an efficient energy technique that can lead to optimum sensing coverage as it is an important key to measure the QoSs in the WSN. Mobile sensor nodes require geo-location information or hardware (e.g. GPS) for each time they run. But this will consume more energy as well as the cost for the hardware where GPS cost is about £30 each with approximately 3 metres accuracy. Reducing the sensing range and adjustable sensing range are some of the main efficient keys to reduce the energy consumption as well as extending the network lifetime. Qu has introduced a self-relocation and self-healing algorithm based on the average distance between the nodes (usually 18-25 m) where geo-location or GPS is not required. The received signal strength allows sensors to obtain the appropriate distance between each other and thus to spread, relocate and move themselves to optimal locations. This algorithm was approved by using simulation where coverage of 94% was achieved as a result. However, this algorithm is only suitable for a small distance (i.e. emergency surveillance) due to the random movement direction (Qu and Georgakopoulos, 2012).

Ko Young propose LAR *"Location-Aided Routing"* (Ko et al., 1998), the benefits of this technique is to limits the explore for a new route to an expected "demand zone". One of the main purposes of the geographic location information is to limit the route order flooding to a smaller zone. The order region is expected depends on the previous destination's location and its mobility form. However, if the mobility information is not precise, that could effect on the request zone and may extend to the entire network.

LBRP *"Location-Based Routing Protocols":* In such type of routing protocols, sensor nodes use the position information in the grid to be allocated. Among neighbouring nodes, there is a distance that could be expected on the base of arriving signal strengths. Exchanging such data between relative neighbouring nodes can allow the nodes to coordinate with each other neighbouring nodes (Savvides et al., 2001). Otherwise, the node can be connecting with a satellite using GPS techniques to provide node location. However, for such things to be used nodes need to be supplied with a GPS receiver that uses a low power (Xu et al., 2001).

GAF *"Geographic Adaptive Fidelity"* (Xu et al., 2001) is a routing algorithm based on the location and energy-aware which was initially created for MANET, but at the same time suitable for sensor networks. There are two types of network zone; virtual grid and static zones. For each zone, the nodes cooperate to each other to handle some of the activities. For instance, nodes among themselves would select a single node to be awake for a specific time, and then others would go to the sleep mode. The duties for the selected node (active) is to represent the other nodes in the same zone while in sleeping mode by reporting and monitoring data to the BS. Therefore, in this case, GAF preserves energy by switch off needless nodes in the network, while keeping the routing path at the same reliability level. GPS is to be used by the nodes to indicate their location, which then helps the nodes to appoint themselves in the virtual grid. In terms of the cost of packet routing in GAF, the nodes that are connected to the same point on the grid are considered to be equal.

Inside GAF There are three states. First is active; this is used to inverting participation in routing. Second is discovery; this is used to for deciding the neighbours in the grid. Finally is the sleep; which is used when the radio is switched off. In order to handle the mobility, each node has to rate the time of leaving the grid and send it to its neighbours. In some cases, for instance sleeping neighbours detect their sleeping time to preserve routing fidelity. For the active node, there is leaving time, and before it expires only one of the sleeping nodes would be active. There are two kinds of GAF mobility (GAF-mobility adaptation), non-mobility (GAF-basic).

(Joa-Ng et al., 1999) presented an area-based on two-level link state routing protocol. One of the important aspects about this protocol is dividing the physical region into smaller areas which then called zones. There are two ways to perform these zones either inter-zone or intra-zone. In link states the zone connectivity data among neighbouring zones and the host connectivity data inside each zone need to be well-maintained. For each zone, each host periodically broadcasts its link connectivity data to its neighbours. As for the exchanged data, each node will create its own intra-zone routing table, which contains all the routing information for nodes in the same zone (Liao et al., 2001).

Hass and Pearlman (Hass et al., 1998) have tried to merge reactive and proactive protocols in ZRP by developing a route exploring phase on demand. Latest papers offer comparative performance assessment for the multi-routing protocols. However the routing algorithms do not support the physical location. Hass and Pearlman offer two kinds of algorithms to minimize route exploring overhead based on location information named *"Location-Aided Routing*" (LAR).The new approach may be supported by the GPS "G*lobal Positioning System"* (Parkinson et al., 1983), GPS helps a mobile host to get the physical location. In fact, some of the information's that provided by GPS contain an error, for instance, position information, because the differential in GPS offers a few metres but others offer about 50-100 metres such NAVSTAR Global Positioning System. In the new algorithm, they supposed each host to get its current location precisely, which is available for each host, for instance, there is no error. (Young-Bae et al., 2000).

AZR-LEACH: An Energy Efficient Routing Protocol for Wireless Sensor Networks

Energy consumption remains one of the biggest issues in WSNs and many researchers are working to maximise the sensor life. Cluster-based routing protocols work on selecting cluster heads where CH aggregates both data received and sent between the nodes in the sensor field. Researchers have found this to be efficient in terms of energy saving, for instance Khan and Sampalli (2012) who proposed an energy efficient protocol for WSNs known as AZR-LEACH. AZR-LEACH was proposed for static clustering and works for both large and small networks. In order to achieve an optimal number of cluster head selection, the network field was divided into four rectangular clusters. Each rectangular cluster was then segmented into zones to ensure effective interaction between the nodes, cluster head and base station. Khan and Sampalli asserted that the layout of dividing the network field into rectangular zones improved the WSN communication between nodes and CH, as well as improving the network stability and energy consumption, which lead to a prolonged network lifetime (Khan and Sampalli, 2012).

Similarly, Manzoor proposed the Q-LEACH as a new routing protocol for WSNs. The main focuses were to optimise clustering process, network stability and network life-time. In this protocol, the nodes were randomly distributed in a sensor field. In order to obtain the optimum clustering, the sensor field was then divided into four equal quadrants based on location information. This work sought to achieve optimal load balancing among the nodes and therefore decrease energy consumption. Manzoor et al successfully demonstrated that Q-LEACH improved load balance and energy consumption over the network as a whole. They therefore concluded that dividing the network into quadrants would be more efficient and appealing for clustering in WSNs (Manzoor et al., 2013).

K-means clustering is an algorithm

K-means clustering is an algorithm which divides the nodes in one field into a specific number of clusters K which result in a K centroid for each cluster. Due to the different location of the nodes k-means suggests placing the centroid the furthest distance from other centroids in order to ensure optimum results.

The first stage is assigning these centroids points in the fields and then calculate or assign the node to the centroid point given. The next step is recalculating the given centroids obtained previously as the current centroid of the clusters and binding each of these values together.

This process is carried out via a loop where the user can select the number of cycles for the k-means to ensure that the centroids become static. The main purpose of K -means algorithm is to reduce an objective function for example a squared error function:

$$\boldsymbol{J} = \sum_{j=1}^{n} \sum_{i=1}^{n} |\boldsymbol{x}_{i}^{(j)} - \boldsymbol{c}_{j}|^{2} \qquad \text{Eq. 3.1}$$

Where $|x_i^{(j)} - c_j|^2$ is a chosen distance measure between a data point $x_i^{(j)}$ and the centre of each cluster c_j where n shows the distance from the data point to their particular cluster centre

The K-means algorithm consists of the following steps:

a) Position the K points into the object nodes in the sensor field which are the primary cluster centroids.
b) Assign each node to the nearest centroid
c) After steps 2 completed recalculate the position of the K centroids
d) Steps 2 and 3 are repeated until centroids stop moving. This divides the nodes into clusters so their distance will be easier to calculate.

The k-means algorithm ensures that the process of determining the centroid's value will come to an end however; this method does not provide the best formation of nodes for optimal partition. Furthermore, k-means has multiple cycles to get to the final centroid value and as a consequence consumes more time and energy than alternative clustering methods (Kanungo et al., 2002).

3.5 SLEEPING MODE ALGORITHMS

The wake-up scattering algorithm

Fontanelli proposed a distributed algorithm, which presents convergence results based on a periodic schedule for the wake-up times. The principle of this algorithm is to scatter the implementation of the neighbouring nodes, assuming that two nodes are directly connected with each other in a large area. This requires them to function at specific times. This system uses a scattering algorithm, and is designed to generate low communication overheads. Nodes are synchronised with each other, where each node needs to know the 'wake-up' time of a neighbouring node. Nodes exchange this information before calibration. The node will notice the wake-up time for the node in closest proximity to itself. Based on this information, the node will select a new wake-up time. As a result, this algorithm provides a local stability for general topologies, as well as providing global stability for specific topologies of the WSN, hence maximizing the lifetime of a WSN (Fontanelli et al., 2009).

Threshold based Routing Protocol for WSN with Sleep/Awake Scheduling

In WSN, a user should be able to assign some sensor nodes to monitor specific events, and know when interested events happen in the interested field (Guo and Yang, 2007). Thus, the sensor network builds a bridge between the real world and computation world. Each node typically consists of the five components: sensor unit, Analog Digital Convertor (ADC), Central Processing Unit (CPU), power unit, and communication unit (Yick et al., 2008). The sensor unit is responsible for collecting information as the ADC requests, and returning the analogue data it sensed. ADC is a translator that tells the CPU what the sensor unit has sensed, and also informs the sensor unit what to do. Communication unit is tasked to receive command or query from, and transmit the data from CPU to the outside world. CPU is the most complex unit that interprets the command or query to ADC, monitors and controls power unit if necessary, processes received data, computes the next hop to the sink etc. (Sharma and Singla, 2016)

Energy efficiency is a central challenge in sensor networks, and the radio is a major contributor to overall energy node consumption. Current energy-efficient MAC protocols for sensor networks use a fixed low-power radio mode for putting the radio to sleep. Fixed low-power modes involve an inherent trade-off: deep sleep modes have low current draw and high energy cost and latency for switching the radio to active mode, while light sleep modes have quick and inexpensive switching to active mode with a higher current draw. Jurdak proposes adaptive radio low-power sleep modes based on current traffic conditions in the network. It first introduces a comprehensive node energy model, which includes energy components for radio switching, transmission, reception, listening, and sleeping, as well as the often disregarded microcontroller energy component for determining the optimal sleep mode and MAC protocol to use for given traffic scenarios.

The deepest sleep mode, which turns off the oscillator and voltage regulator, provides the lowest current draw of all low-power modes. However, it also involves the highest energy cost and the longest latency for switching the radio back to active mode. In contrast, the lightest sleep mode provides a transition to active mode that is quick and energy inexpensive, but it has a higher current draw. In a low traffic scenario, it is better to use the deep sleep mode as nodes spend more time sleeping than switching back and forth between sleep mode and active mode. In a high traffic scenario, a lighter sleep mode is more suitable as the cost of switching the radio frequently into deep sleep mode would exceed the energy saving of the deep sleep mode's low current draw (Jurdak et al., 2010).

Kaebeh proposed *Dynamic and Real-time Sleep Schedule Protocols for Energy Efficiency in WSNs* in order to conserve energy. The network must provide quality of service sleep schedule, and use a mechanism to turn off the radio receiver periodically in coordinating method. Generally, one of the techniques that can fulfil WSN energy efficiency is reducing task-cycle of sensors' operations. The task-cycle can be reduced in the sleep schedule of nodes. The base-rate of nodes' energy consumption can be determined by the nodes' sleep schedule task-cycle. The energy consumption of sensors also reduces when neither transmits nor receives any data packet. Sensing Energy Each sensor node can include several physical sensors, ranging from simple temperature sensors to complex video sensors. Each of these sensors will typically have its own energy consumption characteristics, and in some cases, its own sampling frequency (Kaebeh et al., 2016).

(Trigoni et al., 2004) propose a methodology for trading energy vs. latency in the context of sensor database systems. More specifically, they propose a new protocol which schedules message transmissions very carefully and in such a way so as to avoid collisions at the MAC layer. The sensor nodes' radios can be turned off most of the time and they wake up only during very well defined time intervals. It was shown how routing protocols can be optimised to interact in a symbiotic way with the scheduling decisions, resulting in significant energy savings, however, at the cost of higher latency.

(Arumugam et al., 2005) and (Kulkarni et al., 2006) argue that TDMA is desirable in WSNs for saving energy because TDMA allows a sensor node to minimize idle listening.

Moreover, TDMA has proved to be applicable in converting existing distributed algorithms into a model, which is consistent with WSNs. They proposed a self-stabilizing, deterministic algorithm for TDMA in WSNs where a sensor node is aware only of its neighbours. They also discuss the optimizations in order to improve bandwidth utilization and recovery from corrupted slots. The main focus was on the problem of energy-efficient converge cast (source-to-sink communication) in sensor networks. This problem identifies the energy-latency trade-offs during converge cast. They show how TDMA can be effectively used to provide energy-efficient converge cast. Their solution allows the sensor nodes to save energy when the network is idle and to switch to active mode whenever the network detects an event.

Cui proposed a simple link scheduling algorithm to find the minimum-delay schedule that provides the slot lengths for all the links. Their next step is to combine the obtained results with their previous work concerning an energy-optimal cross-layer design in order to reduce to the minimum the delay of transferring a fixed number of bits from the source nodes to the sink in an energy-limited manner. Moreover, they study the trade-off between the total energy consumption and delay (Cui et al., 2005).

(Keh et al., 2011) proposed *the Optimal Sleep Control for Wireless Sensor Networks* as an effective sleep mechanism to save the energy of sensor nodes. The nodes were randomly deployed and the nodes distance to the sink was used to determine the sleeping probability. Also the nodes remaining energy was used as a parameter to set the active and sleep time. As a result frequency of the transmission for the nodes near the sink has been effectively reduced thus network loading balance was achieved.

3.6 WEIGHTING BASED ALGORITHMS

A massive quantity of research work has been carried out on the characteristic of the weight based clustering algorithms. This work has started by Chatterjee when he first introduced the WCA (Weighted Clustering Algorithm) in 2002 (Chatterjee et.al, 2002). WCA is an on demand weighted clustering algorithms which was initially designed to work for mobile *Ad-hoc* networks. WCA takes into consideration many different parameters as the main attributes in order to select the CH. These parameters are the battery level power, the transmission power, the mobility of the node and the total number of nodes to be supported. Each node works out the values of these parameters and result in one final figure which called the node weight, then node with highest weight value is to be chosen as the CH. Once the cluster is formed then

WCA involve less communication and messages between all nodes in the network which result in using less energy. WCA had performed better than other algorithms which use one parameter at a time; however it still lacks a drawback before the clustering process start due to the unknown of the nodes weights and the complex computing that each node has to go through. Recent studies has taking the WCA weighting clustering base as a new route to enhance the WCA and come up with new approaches. The following

shows the essential points of the further WCA researches and approaches.

Table 3-1 Recent WCA base algorithms literature review

Year	Authors	Paper Title	Issue	Proposed work
2010	Muthuramalingam, S., RajaRam, R., Pethaperumal, K. and Devi, V.K.	A Dynamic Clustering algorithm for MANETs by modifying Weighted Clustering Algorithm with Mobility Prediction	signals and power are wasted to update the positional information of the nodes	WCA + Mobility Prediction
2010	Muthuramalingam, S., Viveka, R., Diana, B.S. and Rajaram, R.	A Modified WCA for stable Clustering using Mobility prediction Scheme	high mobility of nodes = high frequency of re-affiliation = increase the network overhead	A time-based WCA (duration of nodes that are alive is considered) + Link Expiration Time
2014	Sowmya, K.S. and Cauvery, N.K.	Weighted Clustering Algorithm with Ant Colony Optimization to Provide Better Quality of Service	Rapidly changing topology of (MANETs) = Quality of Service (QoS)	WCA + Ant Colony Optimization
2015	Hudedagaddi, D., Ravishankar, P., Rakesh, T.M. and Dengi, S.	Comparison of WCA with AODV and WCA with ACO using clustering algorithm	MANETs routing challenge = optimal QoS	WCA + Ant Colony Optimization

2014	Kothari, D., Rao, D.S. and Singh, A.	An Efficient Cluster head Election Algorithm for Mobile Ad-hoc Network	scalability, stability and efficiency of the network	WCA by leveraging most connected, least utilized node for cluster head election
2014	Karimi, A., Afsharfarnia, A., Zarafshan, F. and Al-Haddad, S.A.R.	A Novel Clustering Algorithm for Mobile Ad Hoc Networks Based on Determination of Virtual Links' Weight to Increase Network Stability	Stability of clusters = rapid failure of clusters, ,high energy consumption and decrease in the overall network stability	WCA that considers the direct effect of feature of adjacent nodes. It determines the weight of virtual links between nodes
2014	Jarchlo, E.A. and Bazlamaçci, C.F	Life Time Sensitive Weighted Clustering on Wireless Sensor Networks	Energy efficiency in clustering WSN, and Prolong the network life	Modifying as well as enhancing the (WCA) to be used in WSN
2011	Nassuora, A.B. and Hussein, A.R.H.	CBPMD: A New Weighted Distributed Clustering Algorithm for Mobile Ad hoc Networks (MANETs)	The number of clusters formed in the network and the overhead.	assumed a predefined limit for the number of nodes to be held by a cluster-head

2008	Zainalie, S. and Yaghmaee, M.H.	CFL: Clustering Algorithm for Localization in WSNs	Enhance the WCA by have a maximum number of nodes and minimum number of clusters	Broadcasting hello message through the network to build a neighbor table then have its own way to calculate the weight.
2005	Ding, P., Holliday, J. and Celik, A.	Distributed Energy-Efficient Hierarchical Clustering for Wireless Sensor Networks	network lifetime and improving scalability	node locates its neighbors, then calculate weight based on residual energy and distance

CHAPTER 4:

METHODOLOGY

4.1 SIMULATION OVERVIEW

Simulation is a powerful tool that has been successfully used to better understand the behaviour of a system. It is used for the modelling of several processes, to mimic real-life systems in a fast, high performance, and low cost way. The definition of simulations has varied over time, Johnson and David (1996) describe simulation as "*A computer program that defines the variables of a system, the range of values those variables may take on, and their interrelations in enough detail for the system to be set in motion to generate some output. The main function of a computer simulation is to explore the properties and implications of a system that is too complex for logical or mathematical analysis*". (Johnson and David, 1996)

Simulation is one of the only methods through which information about real-world systems can be extracted, circumventing the issues such as system complexities, the large scales of real-life systems, and importantly, the very high costs involved. Using simulations, this information can be extracted and analysed effectively.

As described previously, wireless sensor networks (WSN) are modelled using simulation tools for several reasons;

- High versatility leads to the use of these simulations in a wide range of environments and locations.
- Simulations carried out can be repeatable and can obtain a large volume of the information required.
- Simulations circumvent require simple computational modelled with flexibility in order to solve the problem and don’t disrupt the progress of analysis.

Considering all of the above, simulation is understandably the common tool of choice that is widely used in system analysis. Drawbacks of using simulations are small, and are therefore often overlooked; for example, the complexity of the simulation often increases with the complexity of a system, and as a result simulator construction can become complicated, as well as there being an increase in the time required to run the simulation. However, these complications can be removed to further advance and optimise the protocol. Similarly, software packages can often be used for network modelling which can also help in overcoming the drawbacks of using simulation.

4.2 ULTIMATE GOAL: WHAT SHOULD BE EXPECTED FROM A GOOD WSN SIMULATOR?

The most important outcome of intensive simulation designs is that they are available, reusable, scalable and most importantly, are of high quality performance. They are also required to support different scripting languages for experiment definition and for the processing of results. Finally, they are required to have graphical, trace and debug support. Some of these qualities are described below in further detail, when considering these factors in WSNs (Egea-López et al., 2005).

4.2.1 Reusability and availability:

One of the main uses of simulation software is in the testing of new techniques in various scenarios, which could be both controlled and realistic. Using simulations, researchers can compare and contrast the effectiveness of existing techniques, against novel ones. The models therefore need to be both easy to modify and integrate into existing ones. To optimise the ability to integrate and modify models, information is required regarding the length of time that a framework has been in use, and similarly, the frequency at which it has been used.

In addition to this, the package needs to be designed such that it has high modularity and clean interfaces for an easy change in functionality. Specific WSN detains can then be modelled using 'ready-to-use' models. These general-purpose packages comprise a TCP/IP suite. There is also a requirement for wireless MAC protocols and *Ad-hoc* routing support (Perkins, 2008).

4.2.2 Performance and scalability:

WSN simulation has a direct requirement for efficient performance and scalability. The performance of a WSN simulation is dependent on the effectiveness of the programming language used, whereas the scalability is dependent on the processor, logs storage size requirements, and finally, memory usage. Similarly, the simulation design is limited by their real-time operation, which as a result means that they cannot exceed a certain time frame. The increased interaction of WSN simulations with the environment, power consumption and radio propagation, has resulted in an increase in the complexity of scalability and performance issues, and has remained a challenge in the design of simulation tools (Rappaport, 2002).

4.2.3 Support for rich-semantics scripting languages to define experiments and process results:

There are several points that require consideration in the configuration of a WSN trial. These include the following as listed:

- Number of nodes in the test
- Placement of each node
- Movement of nodes
- Number of nodes that move

- Method of node movement
- Energy model used
- Number of physical environments
- Method by which events are generated
- Statistics measured in experiment
- Parameters of radio model

This large number of variables demands that 'input scripting languages' be used, which have high-level semantics. Similarly, due to the many replicas that are generated in experiments, it must be remembered that large volumes of data are likely to be obtained. To overcome this, the 'output scripting language' that is used must be carefully chosen in order to obtain data as quickly and precisely as possible.

4.2.4 Graphical, debug and trace support:

Graphical support is used in simulations as a debugging aid. The graphical support should be capable of inspecting modules, event queues and variables in real time, as well as having the option for 'run-until' and 'step-by-step' execution, which would make the graphical interface extremely useful in debugging the system. Graphical support is also used as a composition and visual modelling tool, which allows the composition of basic modules and aids in the design of small experiments. This is more suitable for small-scale simulations. A final use of graphical support is as a results plotter, allowing the fast and efficient visualisation of data results, without the need for an application for post-processing (Egea-López et al., 2005).

4.3 WSN SIMULATION TOOLS

The following describe some of the most commonly used simulation tools, in the context of wireless sensor networks. There are comparative advantages and disadvantages for each, but often they are chosen depending on their availability (free-source), ease of use, cost and compatibility with layer applications.

4.3.1 GloMoSim

GloMoSim (Global Mobile Information system Simulate) (GloMoSim, 2007) is a simulation tool that is written in Parsec and C. It consists of a library of simulation modules; each specific to a wireless communication protocol. It is scalable and can carry out discrete-event simulations and is often used in mobile wireless and *Ad-hoc* network simulations.

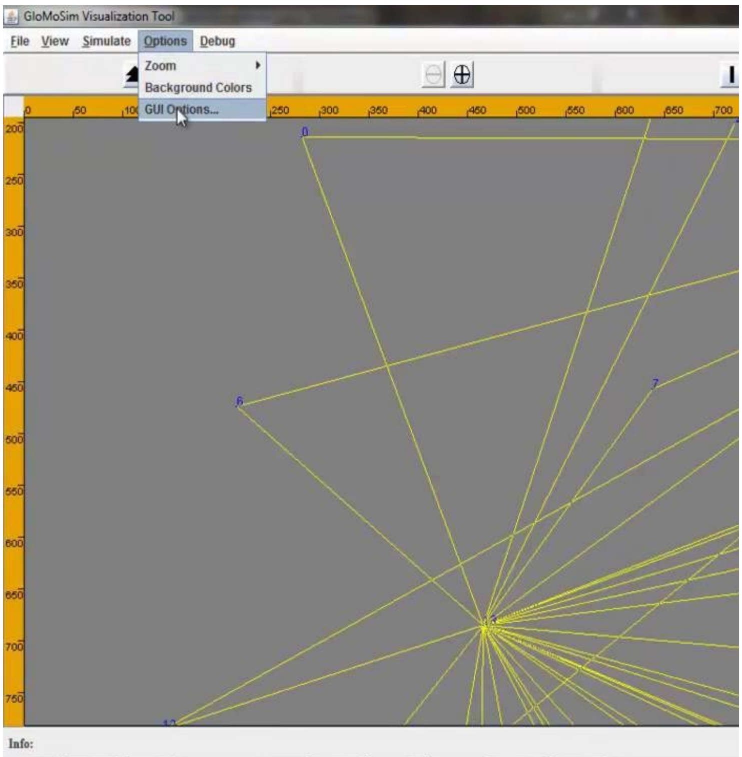

Fig. 4.1 GloMoSim Simulator Tool

4.3.2 OPNET

OPNET (Optimal Network Simulator) is a simulation tool with a hierarchical model structure. It is object-oriented, and was built based on both hybrid and discrete analytic modes of simulation. The UMTS model suite of the OPNET simulation tool allows the modelling of UMTS networks as a way of evaluating data throughput, end-to-end QoS, end-to-end delay, drop rates and delay jitter (OPNET, 2007). OPNET is not an open source, which causes difficulty in modifying and expanding the simulation tool if required.

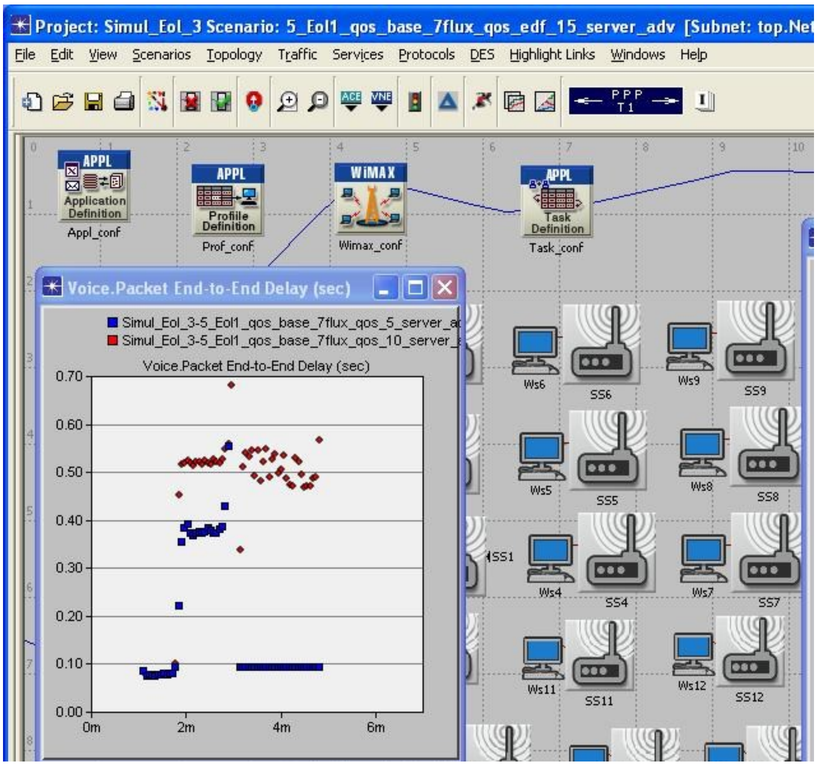

Fig. 4.2 OPNET Simulator Tool

4.3.3 OMNeT++

OMNeT++ is also an event simulator, as well as being a C++ based tool. It is used in the simulation of parallel or distributed communication networks. This tool has a description topology regarding the network language, as well as a graphical network editor and GUI and command line execution environments for its simulation class libraries. Another feature of this simulation tool is its capability to programme models in C++, then assemble them into larger models (OMNeT++, 2006), while providing the architecture for these models.

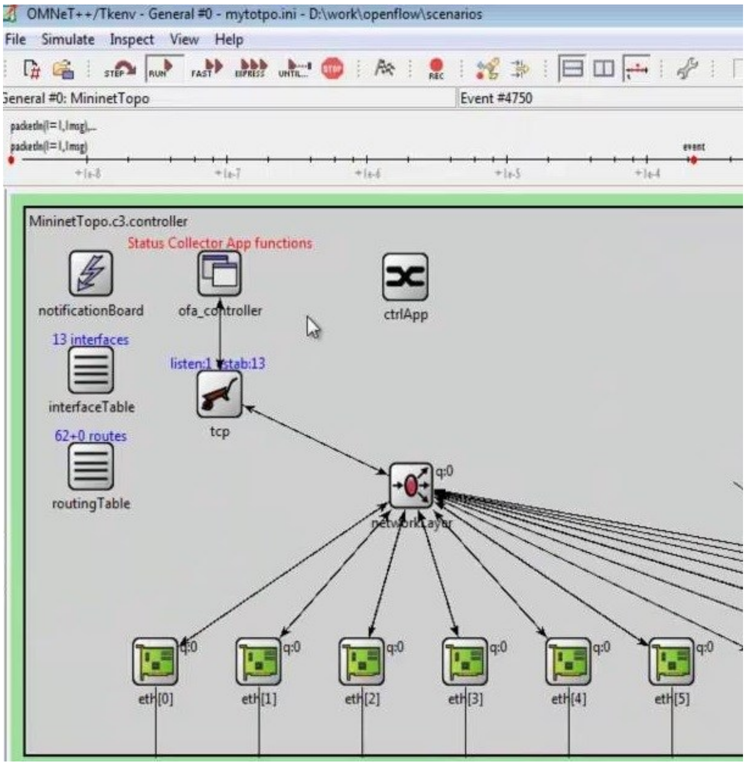

Fig. 4.3 OMNeT++ Simulator Tool

4.3.4 J-Sim

J-Sim (JavaSim) is written in Java. It is a compositional simulation tool that is component based and object oriented. Using this tool, it is easy to add or delete modules, as well as being used in network simulation through the incorporation of real sensor devices. This simulation tool is particularly used due to its provision of power models, energy models, physical media, sensor and base station nodes and finally, sensor and wireless communication channels (J-Sim, 2007).

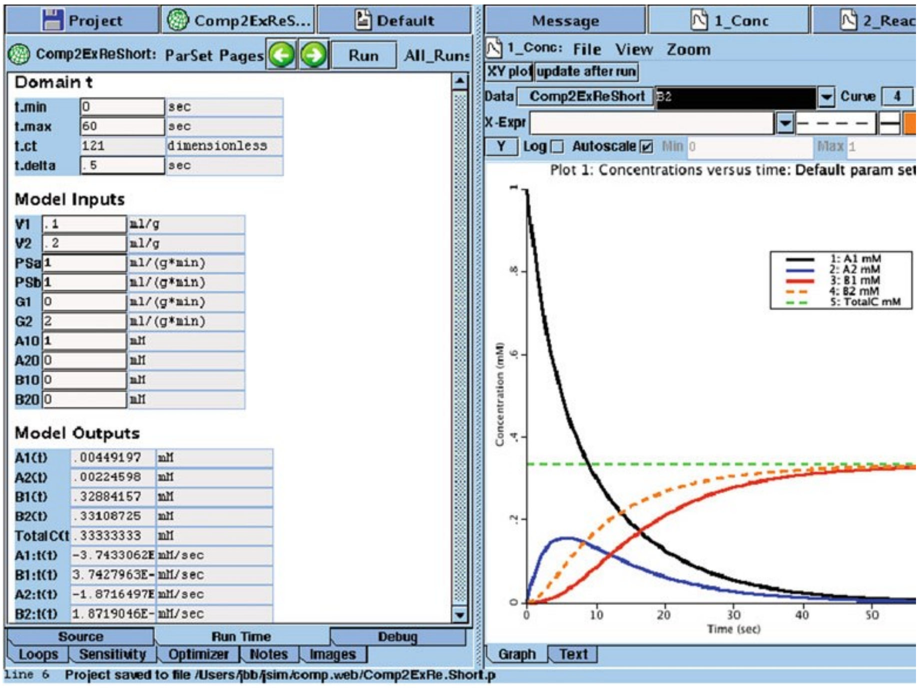

Fig. 4.4 J-Sim Simulator Tool

4.3.5 QualNet

QualNet is an event simulator, used in sensor networks, WiMAX satellite networks, MANETs and others. This simulation tool is used in the simulation of communication networks that are either wired or wireless. It is also capable of using existing network protocols as models for their future organisation around the OSI stack (QualNet, 2006).

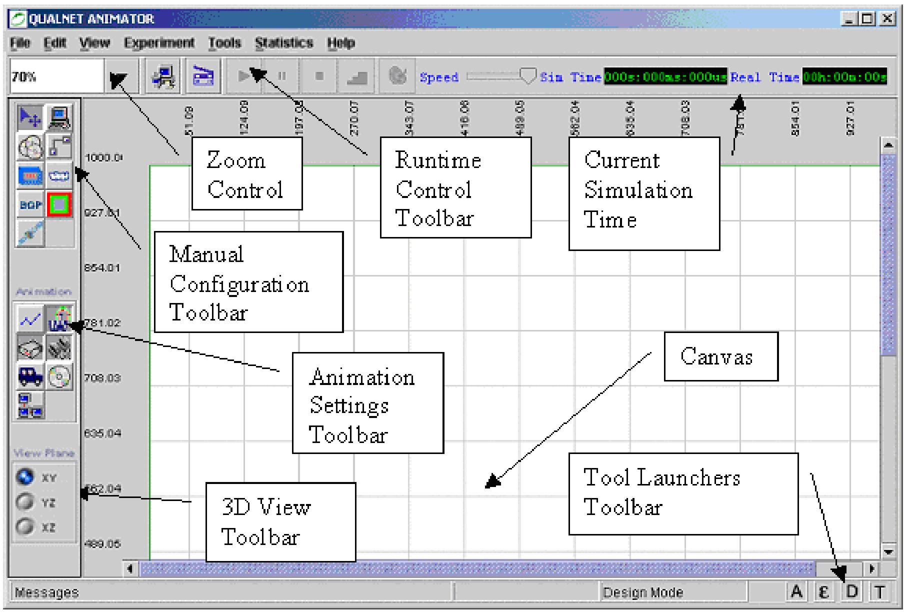

Fig. 4.5 QualNet Simulator Tool

4.3.6 TRMSim-WNS

TrimSim- WSN "Trust and Reputation Models Simulator for Wireless Sensor Networks" is a simulator which was designed for WNS using Java. TrimSim allow users to easily add their model to be integrated as a new model within the simulator, then allow the user to choose the network scenario, for instance static or dynamic network, number of nodes, types of node (client or server), etc.

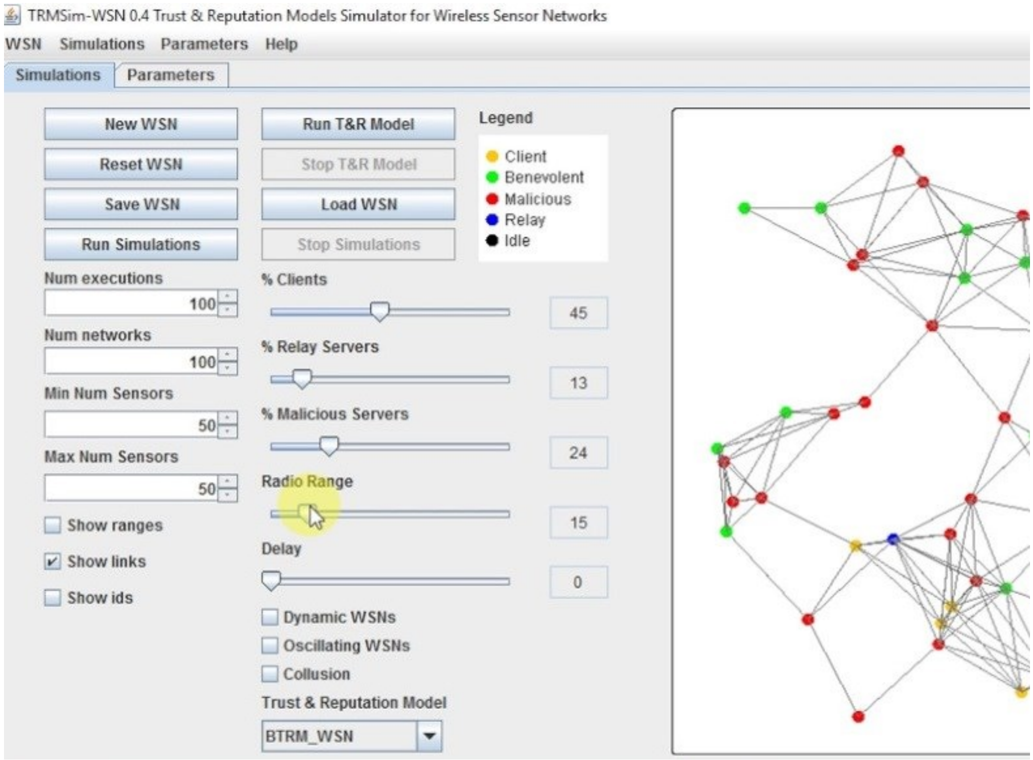

Fig. 4.6 TrimSim- WSN Simulator Tool

4.3.7 NS2

NS2 was developed at Berkeley, at the University of California. It is a free, open source tool which can run on several platforms, including Windows and Linux. This simulation tool is written in Object Tool Command Language (OTcl) and C++, and is an object oriented discrete event simulator. By making modifications to either of the command languages mentioned, NS2 can easily be extended. NS2 has also been validated and used by many members of the research community, which explains its wide range of use and acceptance as a reliable simulation tool (NS2, 2007).

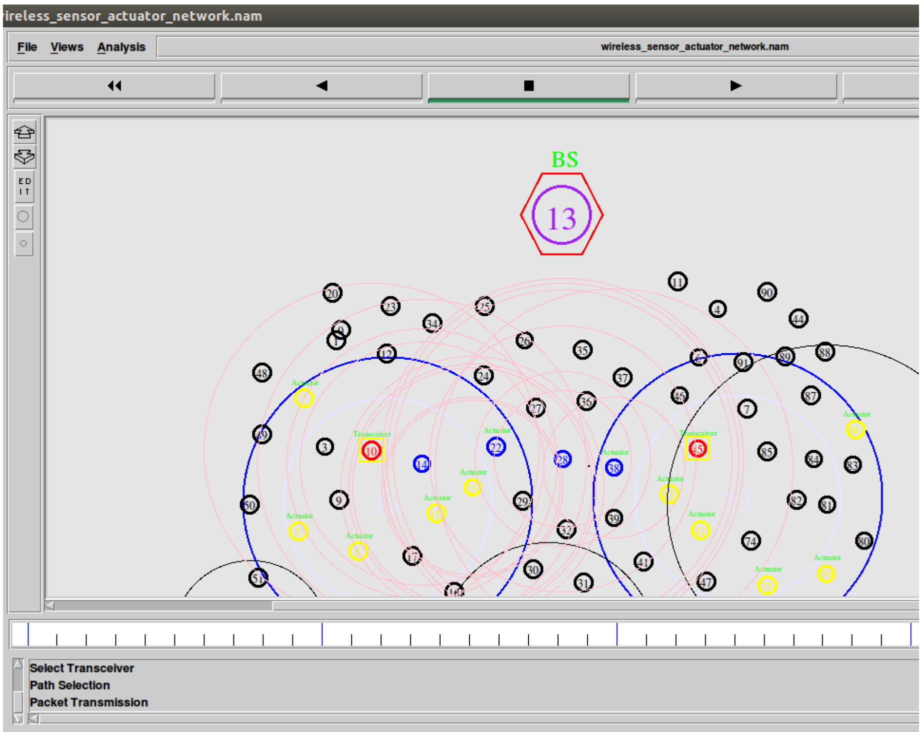

Fig. 4.7 NS2 Simulator Tool

4.3.8 NS3

NS3 is the new version of NS2 which aim to enhance an ideal and open source simulator for the network research. The development focuses of the NS3 were towards the improvement upon the fundamental structural design, models, software incorporation, and educational components of NS2. Furthermore, NS3 development dedicated on building a well-documented solid simulator that is easy to debug and use. NS3 uses the C++ language for programming and in order to write the scripts then python is used where in NS2 TCL is used to write the scripts, thus why the scripts used in NS2 cannot be used in NS3.

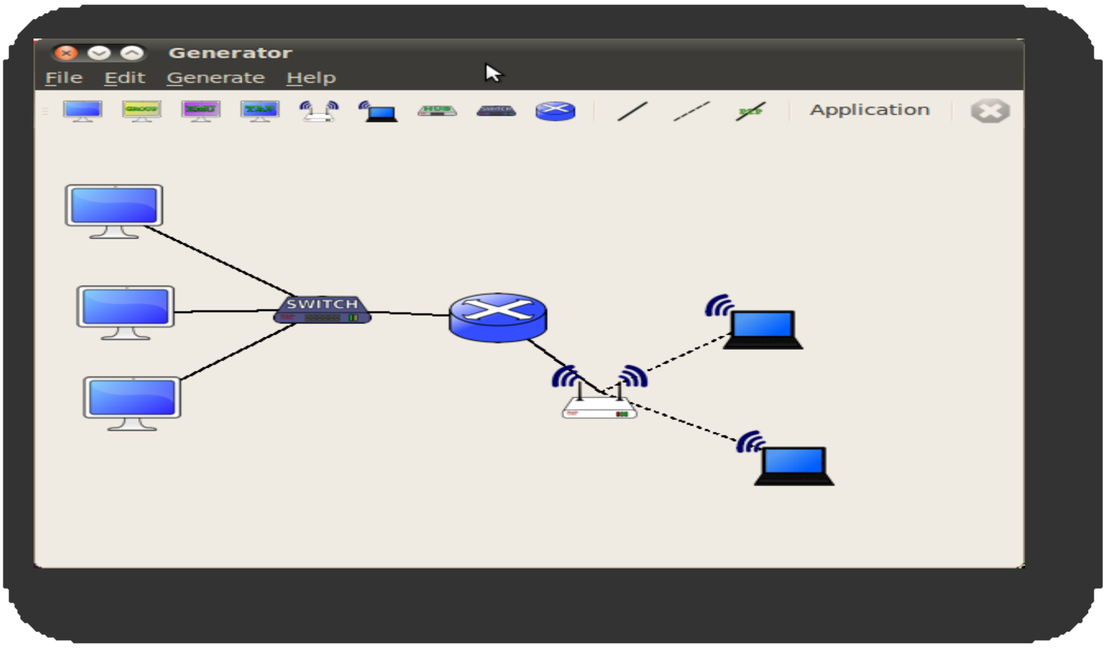

Fig. 4.8 NS3 Simulator Tool

4.3.9 MATLAB

MATLAB is used by scientists and engineers to express computational mathematics as it allow deploying algorithms and applications, designing the systems and analysing larger data sets. MATLAB include a set of running algorithms which also can be enhance, matrix calculations. Furthermore, MATLAB is built-in graphics to support user interfaces (UI), thus data visualisation is easy and more data visions can be gained. MATLAB has been used in many different fields, for instance image and signal processing, finance, smart grid design, and communications (The Mathworks Inc., Natick MA).

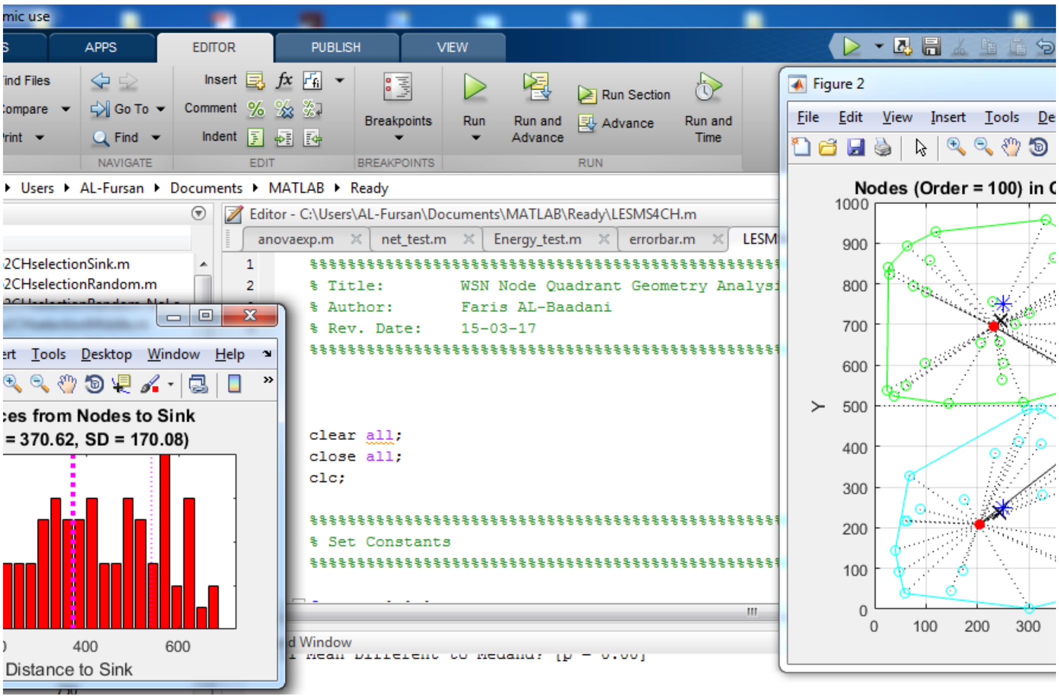

Fig. 4.9 MATLAB simulation tool

4.4 SIMULATORS USED FOR THIS PROJECT

a) MATLAB

b) NS2

MATLAB is used for analysis of layer 1 and TCP/IP protocol in layer 2. While Network simulators, NS2, are used for analysis of layer 2 and 3 of TCP/IP model

4.4.1 NS2/NS3 Similarities and Differences

Although NS3 is the latest version it is a completely new platform. In 2011 the disadvantages of NS3 is that it was still in the early stages of development and it did not have a lot of input from programmers and researchers. This meant that those who maintained the software were slow in answering queries and bug reports that would have helped researchers in testing and validating the system. This means that not many researchers are is using it needs to be tested and developed more to gain credibility. It was suggested (2013) for NS3 to gain credibility NS3 code and scripts needed to be included in research with the results published so that others could get the support from this when carrying out their own research. Also guidance and information on how to use NS3 in different models needed to be available as well as a larger number of simulation models comparable to the number available to NS2. An advantage of NS2 is that it has been around a long time it has benefited from model contributions and components based on the framework of NS2.

It has had software designed for it that allows users to analysis data and create scenarios based on visualising rather than coding. Since 2011 there has been a lot of updates and support made available to NS3 that was not in existence then, such as online tutorials and how to run simulations and updates. This means that in the near future NS3 will become as popular as NS2.

One of the shortcomings of NS2 is that the user needs to have a good knowledge of two languages, C++ and OTcl which is known as dual language architecture. This can cause difficulties for a user in that that it would take more time for him to learn both languages and need to know which one to use to code components. Also as there are two languages NS2 requires more memory management functions. Another disadvantage of NS2 compared to NS3 is how the packets and headers have been designed. In Ns2 a packet is made up of two separate sections one for the header and the other for the payload data. Whereas NS3 a packet consists of a single buffer which corresponds exactly to the steam of bits that would be sent over a real network.

4.4.2 Introduction to NS2

The NS2 (Network Simulator version two) is a common event driven network simulator, which is publicly available. It supports simulations of vast number of protocols. It also offers a rich environment for developing new protocols. Furthermore, large-scale protocol interaction can be studied within its controlled environment.

"The NS2 was developed as part of the VINT project (Virtual Internet Testbed). This was a collaboration of many institutes including UC Berkeley. Version 1 of NS was developed in 1995 and version 2 was released in 1996. Version 2 included a scripting language called Tcl (OTcl). NS2 was first developed as the REAL network simulator. Now, NS2 is supported by Defence Advanced Research Projects Agency and National Science Foundation"

The NS2 is built in Object- Oriented extension of TCL and C++. It can run on Linux or Cygwin (Windows based command line interface) operating systems. NS2 can be used to simulate both wired and wireless networks. The simulator is available as an open source and it provides online documents. Using different layers' protocols, developers, using NS2, have the ability to create computer networks, consisting of nodes, routers and links, and send data packets via the network connections.

NS2 has many and expanding uses including:

- To evaluate the performance of existing network protocols.
- To evaluate new network protocols before use.
- To run large scale experiments not possible in real experiments.
- To simulate a variety of IP networks

OTcl is used to execute users command scripts, which can be used to define:

1. The network topologies.
2. The modules and their relationships.
3. The protocol to be implemented.
4. The application to be simulated.
5. The form of output that is expected to be obtained,
6. etc.

The data path and the control path implementations are separated by NS2 for efficiency reason. Using C++, the event scheduler and the basic network component objects in the data path are written and compiled, in order to reduce packet and event processing time. An OTcl linkage provides these compiled objects to the OTcl interpreter. It thus creates a matching OTcl object for each of the C++ objects. The controls of the C++ objects are given to OTcl, this is achieved by the OTcl linkage making the control functions and the configurable variables, specified by the C++ object, act as member functions and member variables of the corresponding OTcl object.

4.4.3 Characteristics of NS2

The fundamental concept of the NS2 software architecture provides programmable composition and promotes extension by users. This model expresses simulation configuration as a program rather than as a static configuration.

NS2 implements the following features:

- Router queue Management Techniques DropTail, RED, CBQ.
- Multicasting.
- Simulation of wireless networks.
- Terrestrial (cellular, adhoc, GPRS, WLAN, BLUETOOTH), satellite
- IEEE 802.11 can be simulated, Mobile-IP, and adhoc protocols such as:
    - DSR, TORA, DSDV and AODV.
- Traffic Source Behaviour- www, CBR, VBR
- Transport Agents- UDP/TCP
- Routing
- Packet flow
- Network Topology
- Applications- Telnet, FTP, Ping
- Tracing Packets on all links/specific links

4.4.4 Output Files of NS2

When a simulation is finished, NS2 produces one or more text based output files that contain detailed simulation data, if specified to do so in the input Tcl (or more specifically, OTcl) script.

The output form of the simulation can be text or animation format, which could be interpreted interactively using additional tools such as NAM (Network AniMator developed as part of VINT project), and XGraph.

The simulation results are stored as trace files, which can be loaded for analysis by an external application:

A. Text trace files (out.tr) for use with trace file analyser software as XGraph or TraceGraph.
B. NAM trace file (out.nam) for use with the Network Animator Tool

NAM has a graphical user interface with a display speed controller and play functions (such as play, fast forward, rewind, pause, etc…). Information such as throughput and number of packet drops at each link, can be graphically presented, however, the graphical information cannot be used for accurate simulation analysis.

4.4.5 Packet Tracing

There two main packet-flow detail recording activities in packet tracing can be classified as text-based and NAM packet tracing.

Text-Based Packet Tracing

The Two most popular languages for text based tracing are AWK and pearl. The text-based packet tracing records the details of packet flowing through the networks check points (e.g., nodes and queues). The format of each line, in a normal trace file, is shown in Fig. 4.10. In that file, each 12-columns form a line.

event	time	from node	to node	pkt type	pkt size	flags	fid	src addr	dst addr	seq num	pkt id

```
r : receive  (at to_node)
+ : enqueue  (at queue)          src_addr : node.port (3.0)
- : dequeue  (at queue)          dst_addr : node.port (0.0)
d : drop     (at queue)
```

```
r 1.3556 3 2 ack 40 ------- 1 3.0 0.0 15 201
+ 1.3556 2 0 ack 40 ------- 1 3.0 0.0 15 201
- 1.3556 2 0 ack 40 ------- 1 3.0 0.0 15 201
r 1.35576 0 2 tcp 1000 ------- 1 0.0 3.0 29 199
+ 1.35576 2 3 tcp 1000 ------- 1 0.0 3.0 29 199
d 1.35576 2 3 tcp 1000 ------- 1 0.0 3.0 29 199
+ 1.356 1 2 cbr 1000 ------- 2 1.0 3.1 157 207
- 1.356 1 2 cbr 1000 ------- 2 1.0 3.1 157 207
```

Fig. 4.10 Text-Based Packet Tracing file showing the format of each line in a trace file

A trace file by itself is not adequate unless meaningful data is extracted from it. A user can extract data of interest, such as average throughput and packet-time taken to reach destination, from the trace file for further analysis in post analysis phase.

AWK Language

Awk, stands for the names of its authors “Aho, Weinberger, and Kernighan”, is a programming language that is used for pattern scanning and processing and it allows easy manipulation of structured data and the generation of formatted reports. It looks through one or more files for specified criteria and then executes associated actions. Patterns in AWK control the execution of rules: a rule is executed when its pattern matches the current input record. Any AWK expression is valid as an AWK pattern. Then the pattern matches if the expression's value is non-zero (if a number) or non-null (if a string). The expression is re-evaluated each time the rule is tested against a new input record.

Some of AWK key features are:

a) Awk views a text file as records and fields.

b) Awk has variables, conditionals and loops

c) Awk has arithmetic and string operators.

d) Awk can generate formatted reports

e) Awk only works with text files.

f) Awk reads the input file in a standard way and generate a standard outputs file.

AWK can process rows and columns with an ease of use. It understands the same arithmetic operators as C, thus can be considered a pseudo-C interpreter. The AWK can search for particular strings and modify the output using string manipulation functions. It has associative arrays, which can simplify complex problems.

4.4.6 AniMation Trace (NAM)

The NAM is an animated based trace; it records the simulation details in a text file and uses such file to playback the simulation in animated form providing visual interpretation of the network topology. NAM has the following features:

- Provides a visual interpretation of the network created
- Provides a drag and drop interface for creating topologies.
- Can be executed directly from a Tcl script
- Presents information such as throughput, number packets on each link.

Controls include play, stop ff, rw, pause, a display speed controller and a packet monitor facility.

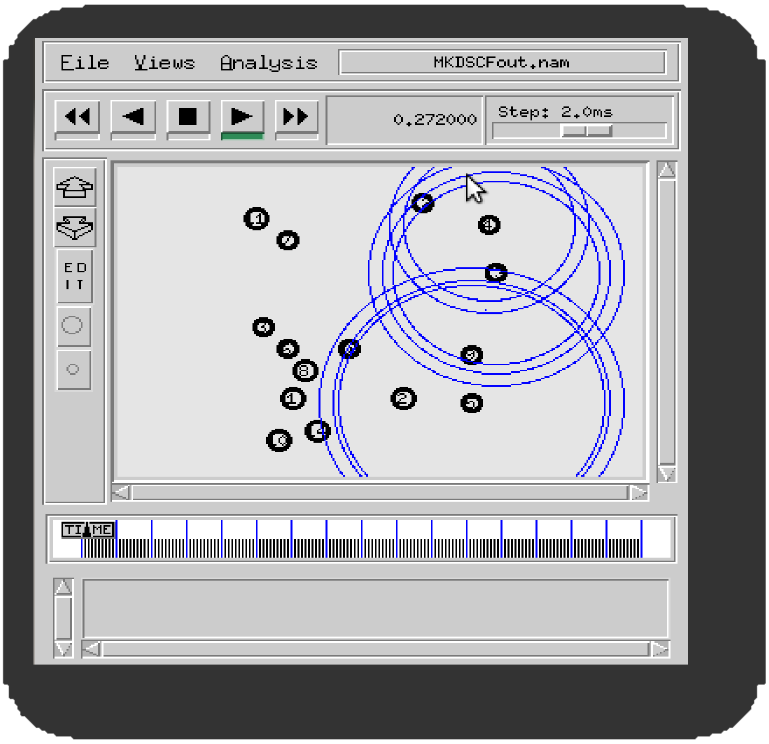

Fig. 4.11 Nam visual trace tools

4.5 VALIDATION METHODS

Before conducting experiments, each model needs to be verified and validated throughout the modifications and the development of the simulator. Verification deals with building the model correctly and is performed using methods of code inspection and model checking, while validation is a comparison of simulation results with that of a known baseline and is performed by comparing measurements and results of real networks to the results of the developed model. Due to high hardware cost and complex tests procedures the results are often compared to analytical approximations rather than to test simulation model against real system. By using the NS2 simulator alongside the AODV protocol, which has been verified and validated by researchers, this has created a baseline for the experiments of this research.

Due to the methodology of this research there were no ethical issues raised in relation to human or animal interaction and the environment.

4.6 HARDWARE EXPERIMENTS DEMONSTRATION

The aim of these experiments is to apply the basic architecture of the WSN by implementing it on hardware.

4.6.1 Overview

Wireless sensor network consists of mainly 3 components. They are as follows:

- Sensors: these have the capabilities to detect the physical or environmental conditions around them.
- Nodes: are electronic devices with smaller processing power which has the capability to communicate with the sensors and forward the data to sink.

- Sink: is the part of the sensor network but it has ample amount of processing capability. These are generally powered and can communicate with the external world as well.

WSN can be implemented in mainly two ways.

- Firstly, single node architecture which is an independent module. It is kind of easy fix module which can sense, analyse and transmit information.
- Secondly, multi node architecture which is used for an application that has many number of sensor distributed in a region

4.6.2 Hardware Specification

- Raspberry pi3; It is a mini computer featuring the following:

a) SoC: Broadcom BCM2837 (roughly 50% faster than the Pi 2);

b) CPU: 1.2 GHZ quad-core ARM Cortex A53 (ARMv8 Instruction Set);

c) GPU: Broadcom VideoCore IV @ 400 MHz;

d) Memory: 1 GB LPDDR2-900 SDRAM;

e) USB ports: 4;

f) Network: 10/100 MBPS Ethernet, 802.11n Wireless LAN, Bluetooth 4.0.

- Arduino UNO R3; It is a device with following features:

a) ATmega328 microcontroller;

b) Input voltage – 7-12V;

c) 14 Digital I/O Pins (6 PWM outputs);

d) 6 Analog Inputs;

e) 32k Flash Memory;

f) 16Mhz Clock Speed.

- ESP8266 WiFi Module device with following features: 802.11 b/g/n, Wi-Fi Direct (P2P), Integrated TCP/IP protocol stack, Integrated TR switch, Integrated PLLs, regulators, DCXO and power management units, +19.5dBm output power in 802.11b mode, Power down leakage current of lt;10uA, Integrated low power 32-bit CPU could be used as application processor, SDIO 1.1/2.0, SPI, UART, Wake up and transmit packets in lt; 2ms, Standby power consumption of lt; 1.0mW (DTIM3).
- Power bank
- GSM Module Sim900
- Battery for sensor

4.6.3 Single node Architecture

Functionality:

The single node sensor architecture contains of:

- Power unit: this is used to provide the operating current to the computing node as well as to the sensor. Power unit can be battery or directly a stationary power supply.
- Sensor: are the electro mechanical devices which have capabilities to detect physical or environmental changes happing nearby them. In this study a flam sensor was used.
- Buzzer: is used to notify the critical conditions to the uses around
- Communication device: are used to communicate the event detected by the node to the remote user or appropriate authority.

Procedure:

a) Ignite flame near sensor;

b) Sensor will detect that flame ;

c) After flame detection sensor sends signal to processor;

d) Processor will process that signal and forward it to GSM module and buzzer;

e) Buzzer will buzz and SMS will be sent through GSM module as per Fig. 4.12 and Fig. 4.13.

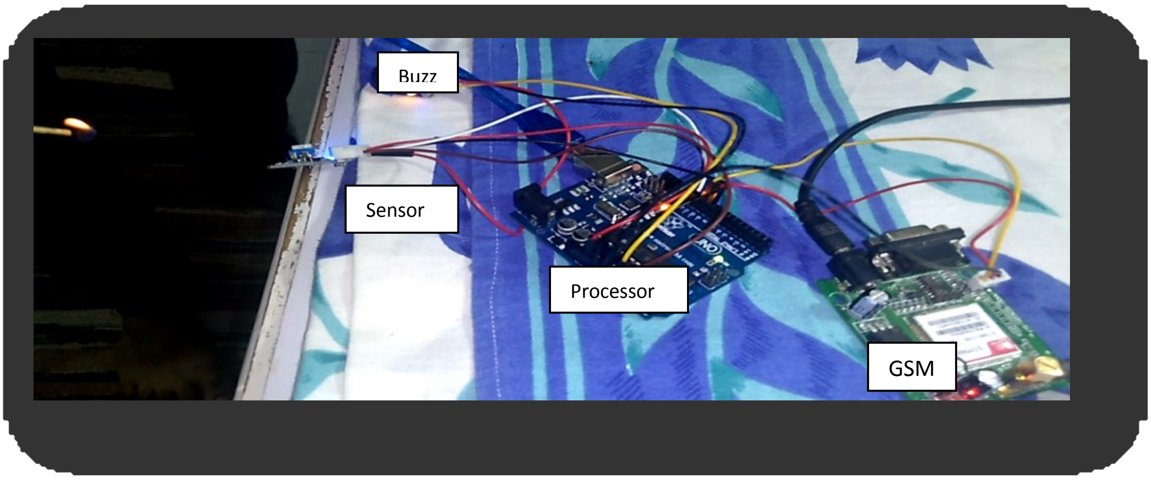

Fig. 4.12 Single node Architecture, including Buzz alarm, sensor, processor and GSM modular

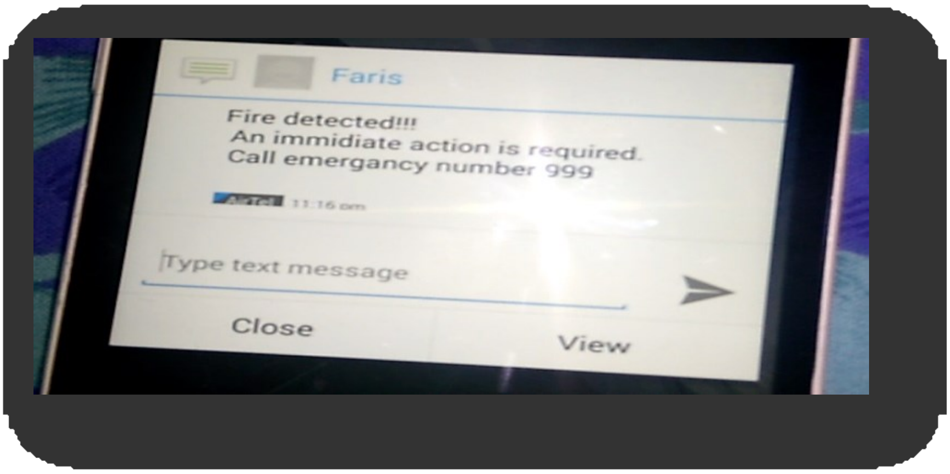

Fig. 4.13 Mobile message received from the single node

Fig. 4.13 shows the message received from the sensor node to the mobile as a result of detecting the flam.

4.6.4 Multi Node Architecture

Configuration details:

The following software and hardware are required for doing practical session

a) Processor
b) Mini-computer device as a sink
c) Different sensing chips as sensor
d) Communicating device having capability to send and receive
e) Router if information needs to be sent to outside of sensor network
f) Compatible battery for sensor and processor
g) Serial port monitor
h) Code developer and burner software
i) Resistors and capacitors to manage current and voltage

Procedure:

a) There are two nodes having battery, processor and communication device.

b) Trigger event from sensor nodes, sensor node will send information through communication device

c) Sink containing high processing power will receive the information from nodes.

d) Sink is connected with two networks, firstly with sensor network through which it receives information and secondly to the external network through a router.

e) Node can be monitor through external network.

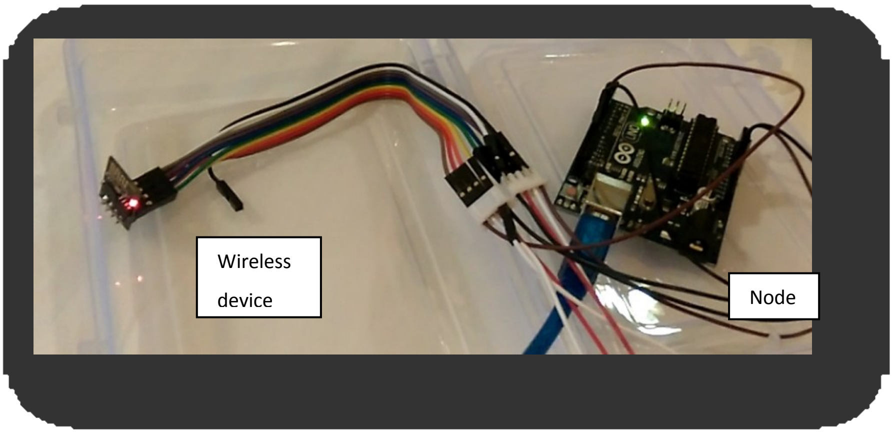

Fig. 4.14 Sensor node connected wirelessly to the sink

Node connected to wireless device in order to be able to communicate with the sink as shown in Fig. 4.14.

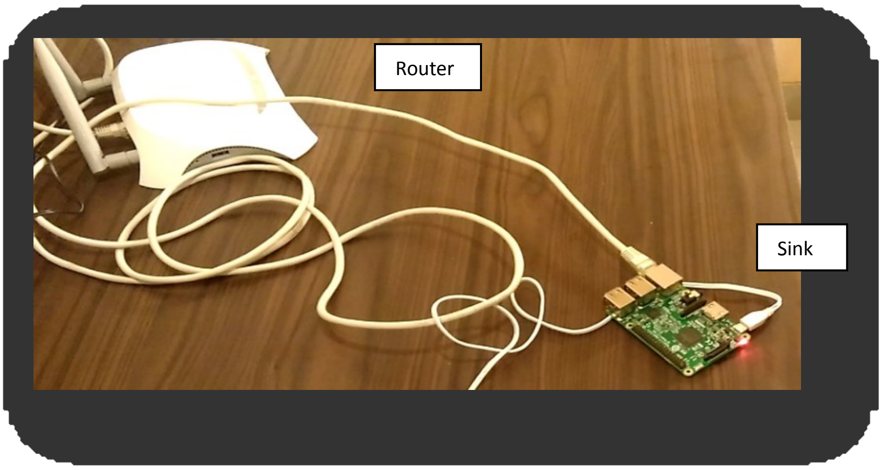

Fig. 4.15 the sink connected to the router for further communications

The sink node is connected to an external network through a router as shown in Fig. 4.15.

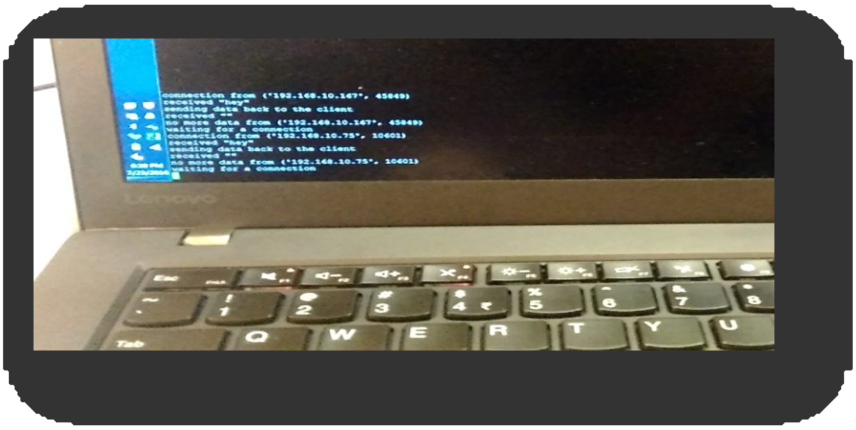

Fig. 4.16 System message for nodes connected successfully

Fig. 4.16 shows the communication data received by the sink from the sensor nodes. This Laptop is connected wirelessly with the router (the external network of the sink).

CHAPTER 5:

LOCATION-AWARE PROTOCOLS

5.1 INTRODUCTION

Wireless Sensor Networks (WSNs) consist of member nodes, cluster head (CH) nodes, a base station (BS), and wireless connections. In this chapter, a simulation is presented in which CH selection approaches are compared. Nodes were deployed randomly in 2D space divided into four or eight regions, such that each region has its own member nodes and local CH. Member nodes were directly connected to their local CH, and each CH was connected to a single BS. The BS was situated at the centre of the WSN, such that it could communicate with each CH. The BS was assumed to have a power supply rather than being powered by battery.

Previous protocols (Liao et al., 2001) and literature (Khan and Sampalli, 2012) suggest that the optimal position for a CH minimises its distance from the nodes for which it is responsible. Five approaches for CH selection to realise this objective are compared: (i) random selection; (ii) selecting the node closest to the arithmetic mean node coordinate; (iii) selecting the node at the medoid coordinate; (iv) selecting the node nearest to the region centre; (v) selecting node nearest the BS. In each case, the dependent variable was the mean distance of all nodes to their local CH, such that a lower mean distance is taken to be superior. In addition, three node grouping schemes are compared: (i) all nodes in the same group; (ii) divide the sensor field into four rectangular quadrants and allocate nodes accordingly; (iii) divide the sensor field into eight sectors and allocate nodes accordingly. The schemes proposed are also compared to the *k*-means clustering approach (Kanungo et al., 2002).

5.2 METHOD

In order to decide which node is the optimum to be selected as the CH, several parameters needed to be taken into account as the basis for the proposed scheme. The following features have been deliberated in the proposed scheme:

- The selection of CH process is not periodic and is invoked as rarely as possible. This results in diminishing the system updates and consequentially reducing the node computation and communication cost;
- Nodes within certain transmission range can result in an efficient power consumption. In other words, nodes communications within close distance from each other will consume less power than nodes at a further distance from each other. CH has more duties to carry out toward its members, thus consumes more power than other ordinary nodes;
- The signal attenuation becomes higher from increasing the distance, thus when nodes move away or placed at a further distance from their CH, the communication becomes more challenging. CH can enhance its performance when at closer proximity from its members within the transmission range.

5.2.1 Assumptions

The following nine assumptions were made:

- The number of nodes is known in advance;
- The nodes are randomly deployed;
- The nodes are static;
- The dimension of the sensor field is known in advance;
- One node per quadrant is expected to act as a CH;
- The sensor node field is located in a flat environment (2D);
- All the nodes are connected together in a single-hop base;
- The coordinate of the BS is known;
- The BS is capable of receiving, aggregating, and then forwarding the data between the CH and the chosen endpoints.

5.2.2 Apparatus

All simulations presented in this chapter were created in MATLAB (The Mathworks Inc., Natick MA). Statistical analyses were performed using SPSS (IBM Corp., Armonk NY).

5.2.3 Procedure

The BS was placed at the centre of a 2D rectangular sensor field, which was set to a size of $1000 \times 1000\mathrm{m}$. A total of 100 nodes were deployed at 2D random coordinates. The following steps were performed:

a) Define the WSN sensor field size (1000×1000) and create 100 nodes at random coordinates within the sensor field ($n = 100$);
b) Set the position of the BS to be the WSN sensor field centre (500×500);
c) If sensor field division to be done, divide the WSN into four (quadrants) or eight (sectors) regions, and allocate nodes to those regions;
d) Identify the CH node by: (i) random selection; (ii) selecting the node closest to the arithmetic mean node coordinate; (iii) selecting the node at the medoid coordinate; (iv) selecting the node nearest to the region centre; (v) selecting node nearest the BS.
e) Calculate distance of each node to local CH;
f) Determine Euclidean, d, between each node and the selected CH using Eq. 5.1, wherein (n_x, n_y) is the 2D coordinate of a node and (b_x, b_y) is the 2D coordinate of the CH.

$$d = \sqrt{(n_x - b_x)^2 + (n_y - b_y)^2} \qquad \text{Eq. 5.1}$$

a) Repeat each simulation 100, 500 and 1000 times to obtain statistically representative behaviour for analysis.

5.2.4 Cluster Head Selection Schemes

Scheme (i): Random Selection

One random node was selected to serve as CH. Where the sensor field was subdivided into four quadrants or eight sectors, one node in each region was randomly selected.

Scheme (ii): Selecting the Node Closest to Arithmetic Mean Node Coordinate

The arithmetic mean 2D coordinate of the nodes was calculated (Eq. 5.2), and the node closest to this coordinate (i.e., shortest Euclidean distance, given by Eq. 5.1) was selected to serve as the CH.

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i, \qquad \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$$ Eq. 5.2

Scheme (iii): Selecting Node at the medoid Coordinate

The medoid has similar concept to the mean and the centroid, but the medoid is a member node of the cluster. The medoid can be calculated by minimizing the absolute distance between the points rather than the square distance. Random nodes selected to be medoids through an iterative process, and their distance to the nodes is calculated using Eq. 5.1 the shortest distance to the nodes is the medoid.

Scheme (iv): Selecting Node Nearest to Region Centre

The sensor field had a size of $1000 \times 1000\text{m}$, which was in some experimental conditions divided into four quadrants or eight sectors. In each case, the node closest to the centre coordinate of the region (quadrant or sector) was selected to serve as the CH (i.e., the shortest Euclidean distance, given by Eq. 5.1). For instance, where subdivision of the sensor field into four quadrants was done, the centre of the four quadrants were $(250,250)$, $(250,750)$, $(750,250)$, and $(750,750)$.

Scheme (v): Selecting Node Nearest BS

In all analyses, the BS was placed at the centre of the $1000 \times 1000\text{m}$ sensor field, at coordinate $(500,500)$. For each node group (four quadrants or eight sectors), the node closest to the BS was selected to serve as the CH, again calculated using Eq. 5.1.

5.2.5 Sensor Field Division Schemes

Scheme (i): Four Quadrants

Placing the BS at the centre of the sensor field creates four quadrants, to be labelled I-IV (Fig. 5.1).

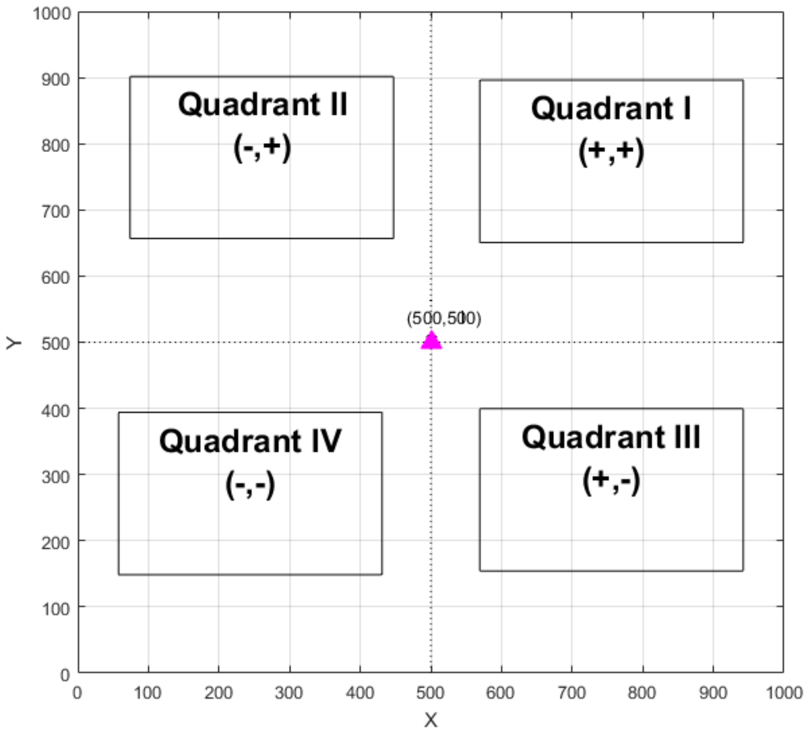

Fig. 5.1: Subdivision of WSN into Four Quadrants (I..IV).

Scheme (ii): Eight Sectors

In this scheme, each quadrant (Fig. 5.1) was further divided into two triangles, creating eight sectors. The centre of these triangles (C) was then calculated as mean of x and y coordinates of the three vertices (Eq. 5.3).

$$C_{(x,y)} = \left(\frac{x_1 + x_2 + x_3}{3}, \frac{y_1 + y_2 + y_3}{3}\right) \quad \text{Eq. 5.3}$$

5.2.6 Statistical Design

The five CH allocation schemes were compared using ANOVA and a series of post-hoc t-tests for each of the three WSN subdivision schemes to determine if there was any significant difference in the dependent variable (the arithmetic mean distance of nodes to their local CH). The null hypothesis is that the CH allocation schemes are not significantly different from one another, with the alternative hypothesis that one or more scheme will yield a significantly lower arithmetic mean distance to the local CH. A *p*-value of 0.05 was used to declare significance in each test.

Finally, a comparison of the quadrant (four) and sector (eight) grouping schemes is compares to k-means clustering for four or eight clusters, again with the dependent variable of mean distance of each node to its local CH using the five CH selection schemes described above (random, arithmetic mean, medoid, nearest to centre, nearest to BS).

5.3 RESULTS AND ANALYSIS

5.3.1 Single Group (No Sensor Field Subdivision)

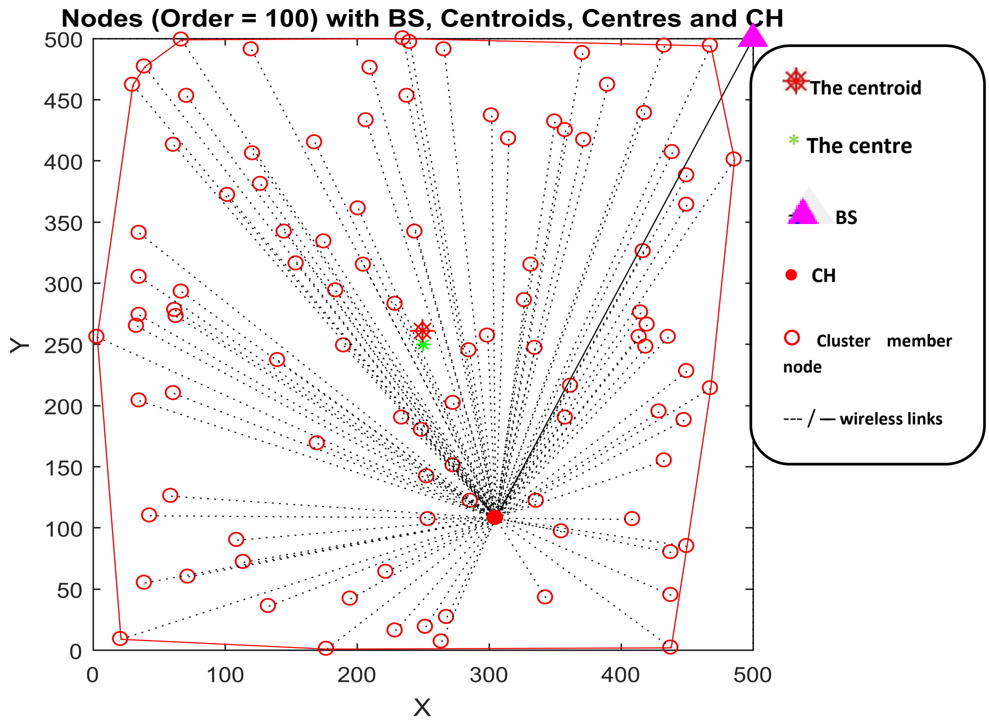

Fig. 5.2: Network base structure used for the experiment of this research.

Fig. 5.2 shows that the BS is placed at the top-end of the sensor's field. The centroid is calculated and is represented by (✱) as well as the centre that is represented by (*). A random node is selected as CH; other member nodes are connected to their CH, which then connects to BS.

Fig. 5.4 shows the mean and the SD Euclidean distance of each node to (a) the random CH (b) the arithmetic mean, (c) the medoid, (d) the centre and (e) the BS

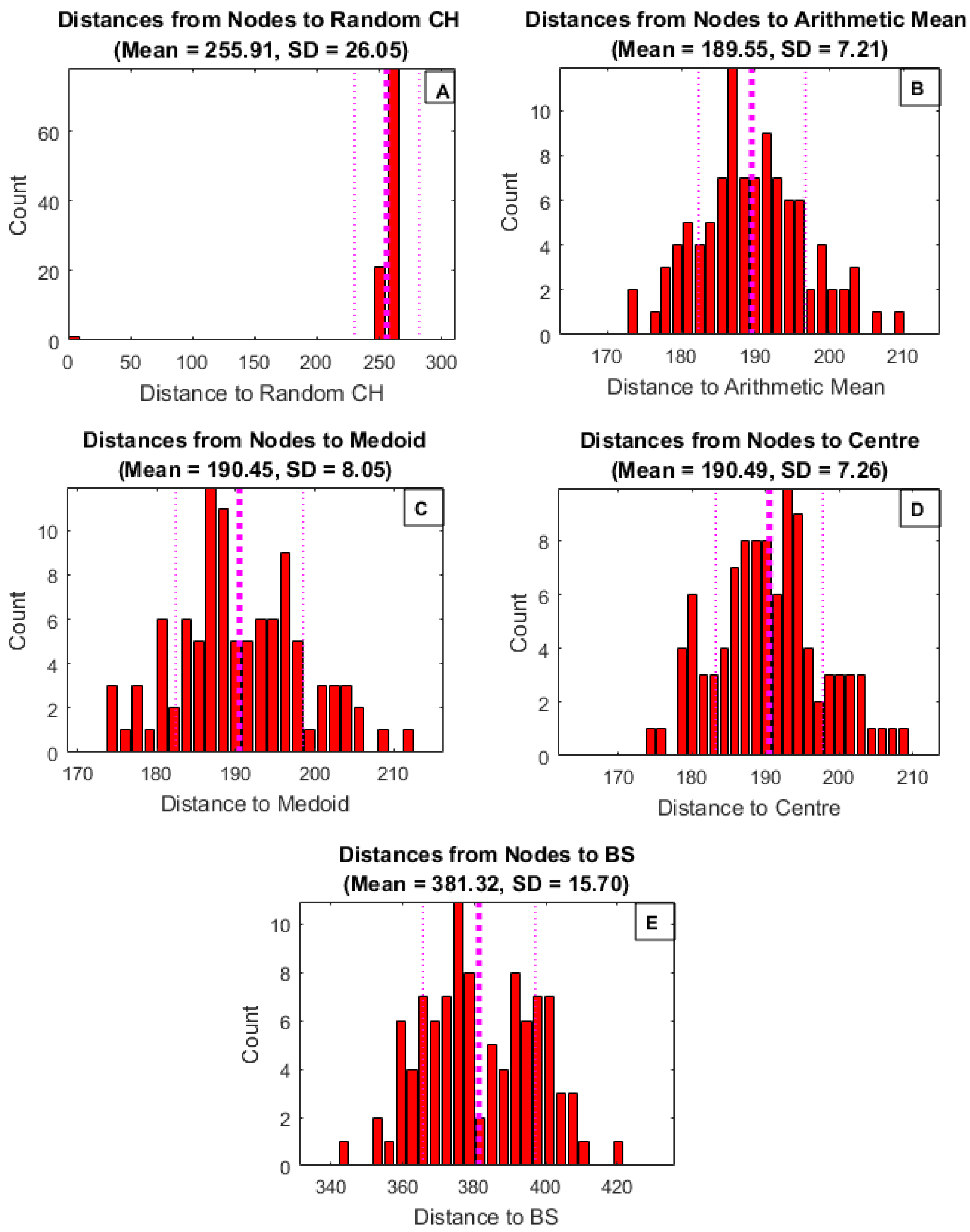

Fig. 5.3: Histograms of Euclidean distances of the nodes to (a) the random CH (b) the arithmetic mean, (c) the medoid, (d) the centre and (e) the BS

Fig. 5.5 shows the distributions of distances from random, arithmetic mean, medoid, nearest to centre, nearest to BS where it can be seen that the majority of the nodes are close to the centroid (red).

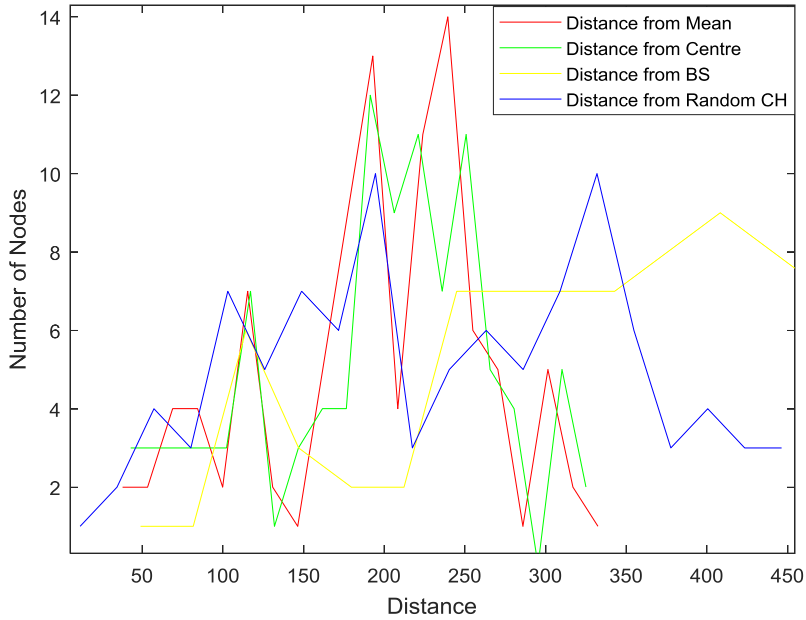

Fig. 5.4: Distributions of distances from: random, arithmetic mean, medoid, nearest to centre, nearest to BS

Table 5.2 describes the data examined. It provides a range of descriptive statistics, including the number of nodes involved (N), the mean value (Mean), standard deviation (Std. Deviation), standard error (Std. Error), and 95% confidence intervals of the dependent variable (Mean) for each single group (Centroid, Centre, Random and BS). It also provides the total of all groups when combined (Total). From the table it can be observed that the means of the Centroid (191.1753) and Centre (192.2465) are almost the same value. Whereas, the mean values of Random (261.3362) and BS (383.0188) are considerably different from the Centroid and the Centre mean values. They also vary significantly from each other.

Table 5.1: Descriptive table for the four node grouping schemes.

Descriptive table								
Mean								
	N	Mean	Std. Deviation	Std. Error	95% Confidence Interval for Mean		Minimum	Maximum
					Lower Bound	Upper Bound		
Random	100	261.33	11.27	1.12	259.09	263.57	220.05	290.84
Arithmetic mean	100	191.17	7.45	0.74	189.69	192.65	161.87	206.97
Medoid	100	192.00	7.49	0.74	190.52	193.49	175.71	216.46
Centre	100	192.24	7.42	0.74	190.77	193.72	162.60	206.23
BS	100	383.01	14.96	1.49	380.04	385.98	347.51	420.97
Total	500	243.95	75.32	3.36	237.33	250.57	161.87	420.97

Table 5.2 shows the output of the ANOVA analysis; and whether there is a statistically significant difference between the means of each CH selection scheme. P-value is shown in the last column (Sig.) and this value has to be less than 0.05 to be statistically significantly different. The table shows that the P-value is 0.000. However, from the ANOVA comparison it does not clearly suggest which one of the schemes is different. To explore this information, post-hoc tests Tukey was performed.

Table 5.2: ANOVA table comparison.

ANOVA Table					
Mean					
	Sum of Squares	df	Mean Square	F	Sig.
Between Groups	2442544.44	3	814181.48	7051.719	0.000
Within Groups	45721.60	396	115.45		
Total	2488266.04	399			

The Multiple Comparisons between the schemes is shown in Table 5.3, where the schemes' differences can be noticed such that:

- Centroid significantly differs from Random (P-value is = 0.000 < 0.05).
- Centroid significantly differs from BS (P-value is = 0.000 < 0.05).
- Centre significantly differs from Random (P-value is = 0.000 < 0.05).
- Centre significantly differs from BS (P-value is = 0.000 < 0.05).
- Random significantly differs from BS (P-value is = 0.000 < 0.05).

However, there were no differences between the schemes that were used for the centre and centroid as the P-value is = 0.895 > 0.05.

Table 5.3: Multiple comparisons table (post-hoc tests).

Multiple Comparisons							
Dependent Variable: Mean							
	(I) Group	(J) Group	Mean Difference (I-J)	Std. Error	Sig.	95% Confidence Interval	
						Lower Bound	Upper Bound
Tukey HSD	Random	Arithmetic mean	70.16*	1.43	0.00	66.21	74.10
		Medoid	69.32*	1.43	0.00	65.38	73.27
		Centre	69.08*	1.43	0.00	65.14	73.03
		BS	-121.68*	1.43	0.00	-125.62	-117.74
	Arithmetic mean	Random	-70.16*	1.43	0.00	-74.10	-66.21
		Medoid	-.83	1.43	0.97	-4.77	3.10
		Centre	-1.07	1.43	0.94	-5.01	2.87
		BS	-191.84*	1.43	0.00	-195.78	-187.90
	Medoid	Random	-69.32*	1.43	0.00	-73.27	-65.38
		Arithmetic mean	.83	1.43	0.97	-3.10	4.77
		Centre	-.23	1.43	1.00	-4.18	3.70
		BS	-191.01*	1.43	0.00	-194.95	-187.07
	Centre	Random	-69.08*	1.43	0.00	-73.03	-65.14

		Arithmetic mean	1.07	1.43	0.94	-2.87	5.01
		Medoid	.23	1.43	1.00	-3.70	4.18
		BS	-190.77*	1.43	.000	-194.71	-186.83
	BS	Random	121.68*	1.43	0.00	117.74	125.62
		Arithmetic mean	191.84*	1.43	0.00	187.90	195.78
		Medoid	191.01*	1.43	0.00	187.07	194.95
		Centre	190.77*	1.43	0.00	186.83	194.71

The suggested Homogenous Subsets are shown in Table 5.4 which compare the means of schemes with significant difference. The groups are listed in ascending order of means, where the means that are listed under each subset contain a set of means that are not significantly different from each other.

Table 5.4: Homogeneous Subsets comparison table.

Mean					
	Group	N	Subset for alpha = 0.05		
			1	2	3
Tukey HSDa	Arithmetic mean	100	191.17		
	Medoid	100	192.00		
	Centre	100	192.24		
	Random	100		261.33	

	BS	100			383.01
	Sig.		0.94	1.00	1.00
Means for groups in homogeneous subsets are displayed.					
a. Uses Harmonic Mean Sample Size = 100.000.					

The analysis can be summarised by testing the null hypothesis that: the means of the four schemes are all equal and equal to zero. P-value suggests that the null hypothesis is rejected in favour of the alternative hypothesis which recommends that the means are different from zero.

5.3.2 Dividing the Sensor Field into Four Quadrants

Fig. 5.5 shows the position of the BS at the centre of the field. The field was divided into four clusters; (A) shows the proposed, Location Energy Sleep Mood Saving (LESMS), scheme network structure and (B) shows the *k*-mean network structure. The average distance of the nodes in the same cluster was used to calculate the mean (X). The centre is represented by (*). The node closest to the mean was selected as the CH for each quadrant; other nodes in the cluster were member nodes, where all members of the same cluster were connected to their CH, which was connected to the BS.

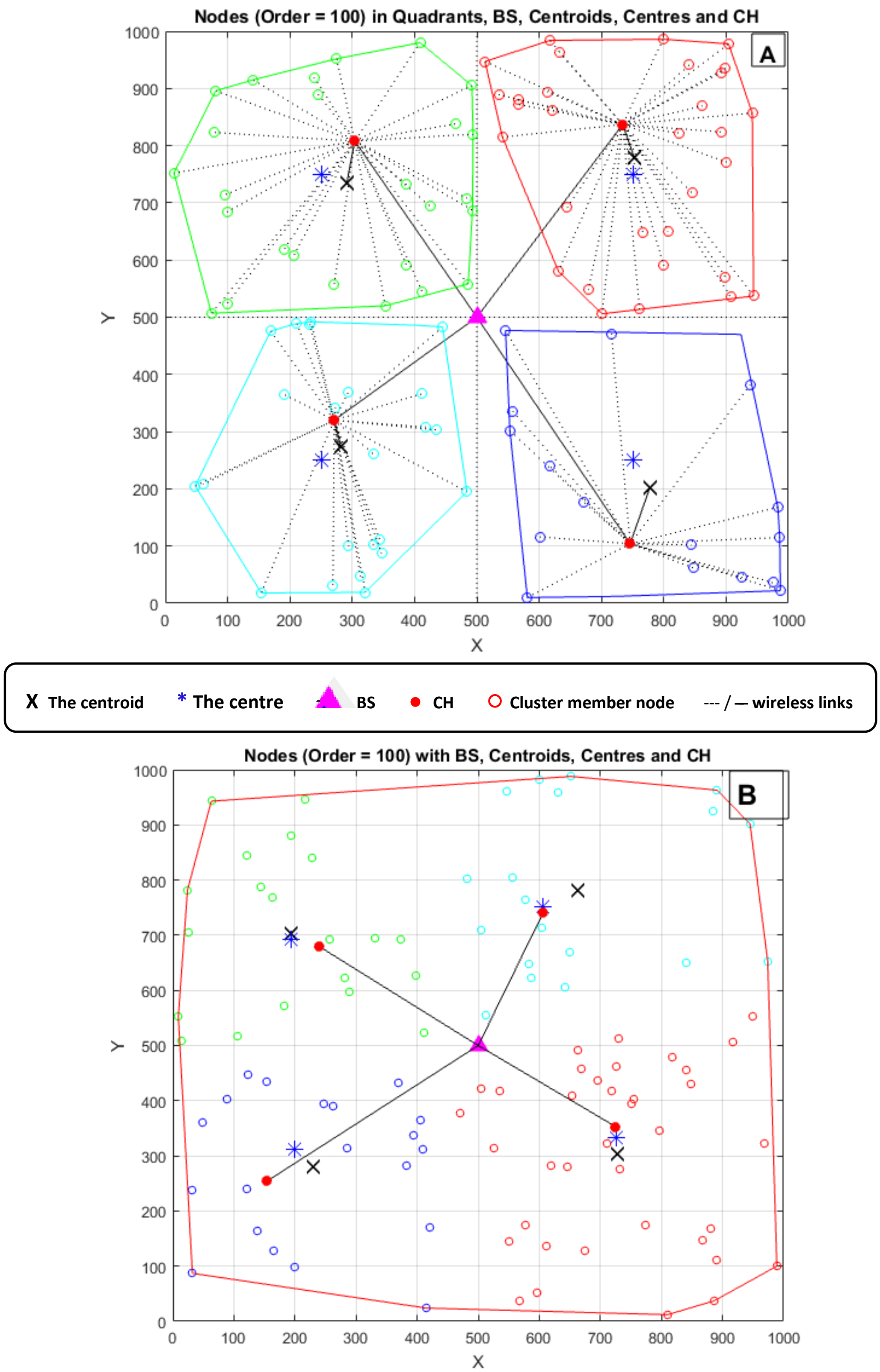

Fig. 5.5: Network base structure with 100 nodes including one CH for each cluster used in four cluster base on; A: proposed method and B: *k*-means. A convex hull has been added to the perimeter to highlight the spatial extent of the nodes.

(Fig. 5.6, Fig. 5.7, and Fig. 5.8)show the mean and the standard deviation values that were acquired from the experiments for each cluster (I, II, III, and IV), and displays the histogram of Euclidean distances of each node to: the arithmetic mean (Fig. 5.6), to the medoid (Fig. 5.7) and the centre (Fig. 5.8).

Also the mean and the standard deviation values are displayed with their histogram of Euclidean distances of each node to the BS (Fig. 5.9).

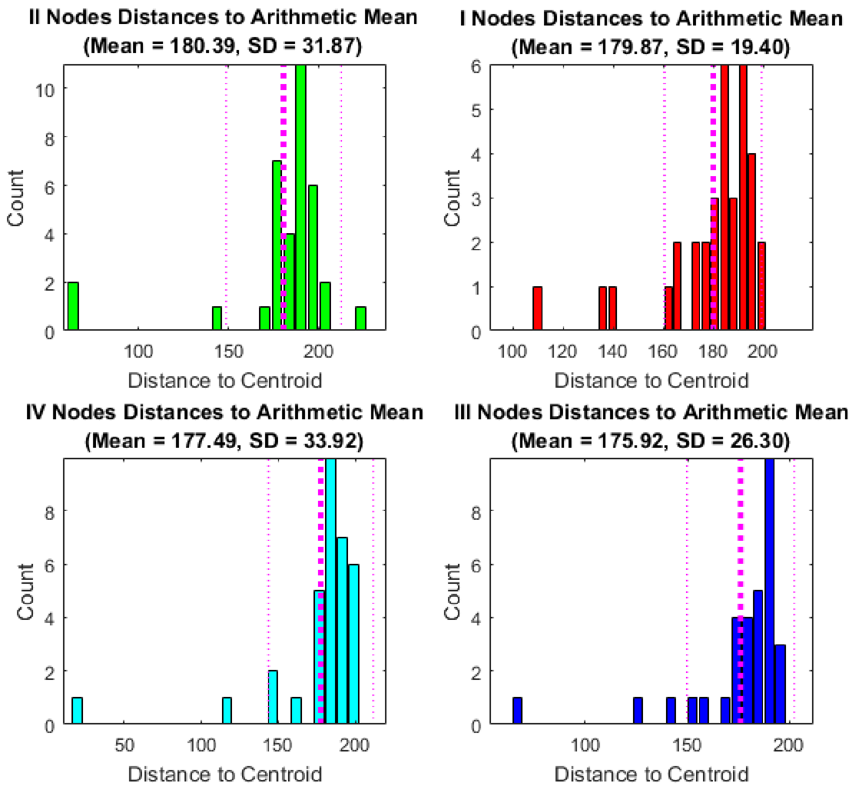

Fig. 5.6: Four quadrants histogram of Euclidean distances of the nodes to the Arithmetic Mean.

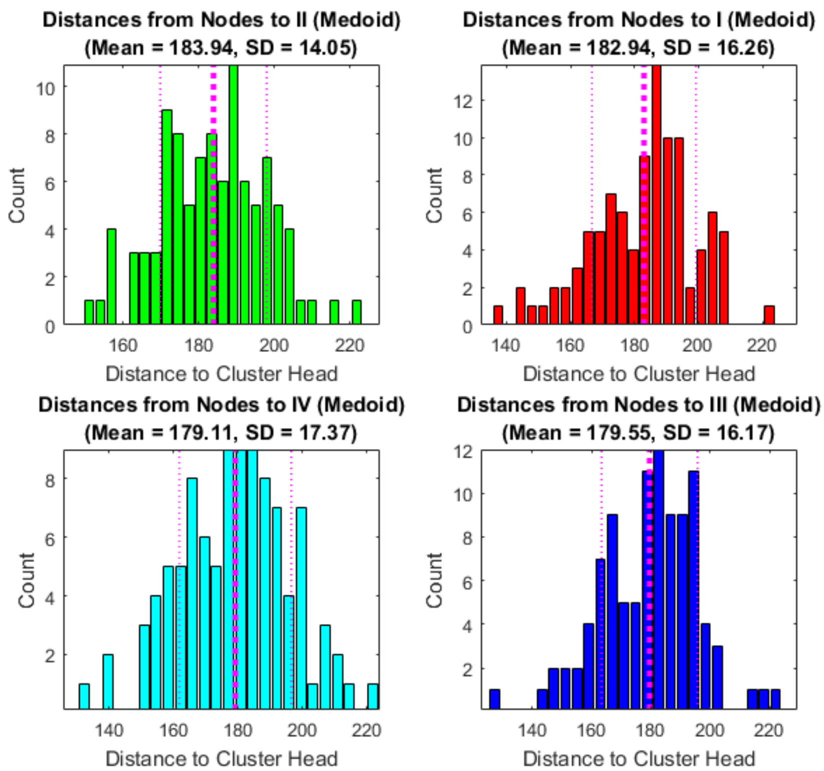

Fig. 5.7: Four quadrants histogram of Euclidean distances of the nodes to the CH.

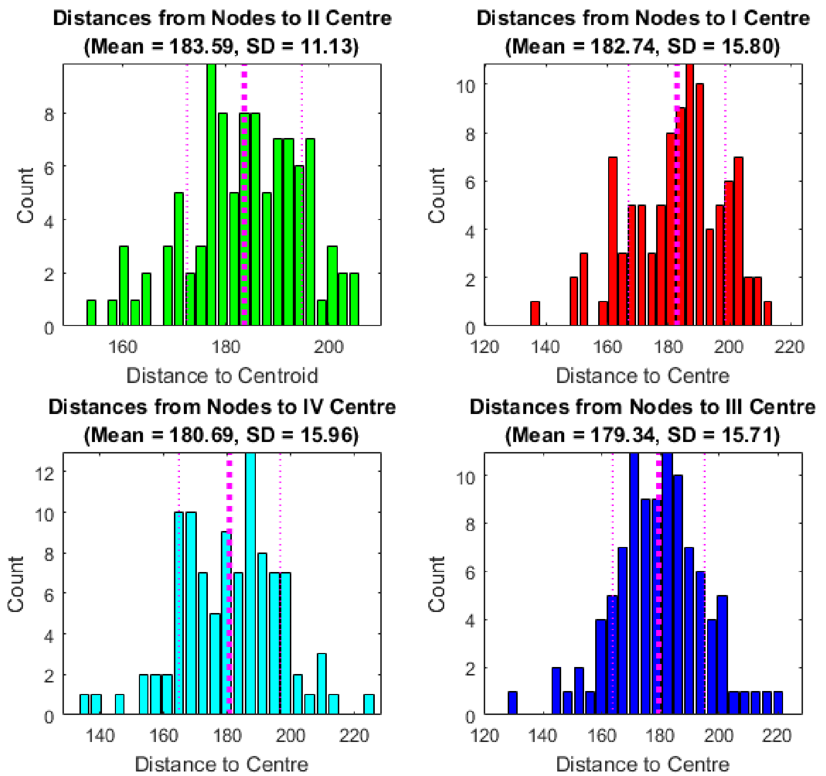

Fig. 5.8: Four quadrants histogram of Euclidean distances of the nodes to the centre.

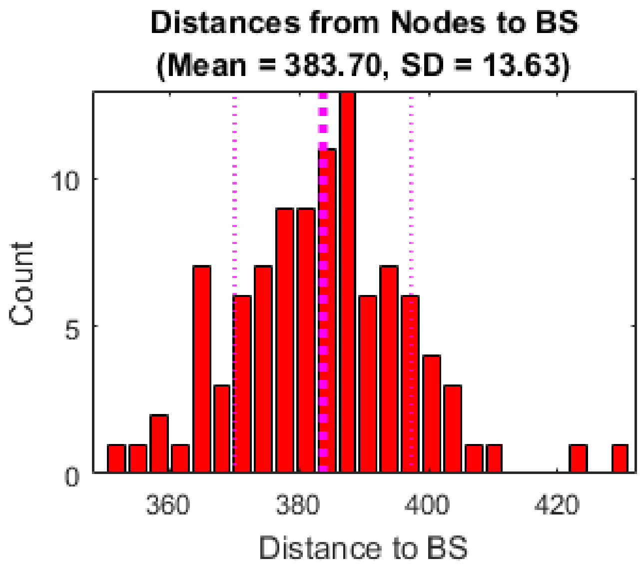

Fig. 5.9: The histogram of Euclidean distances of the nodes to the BS.

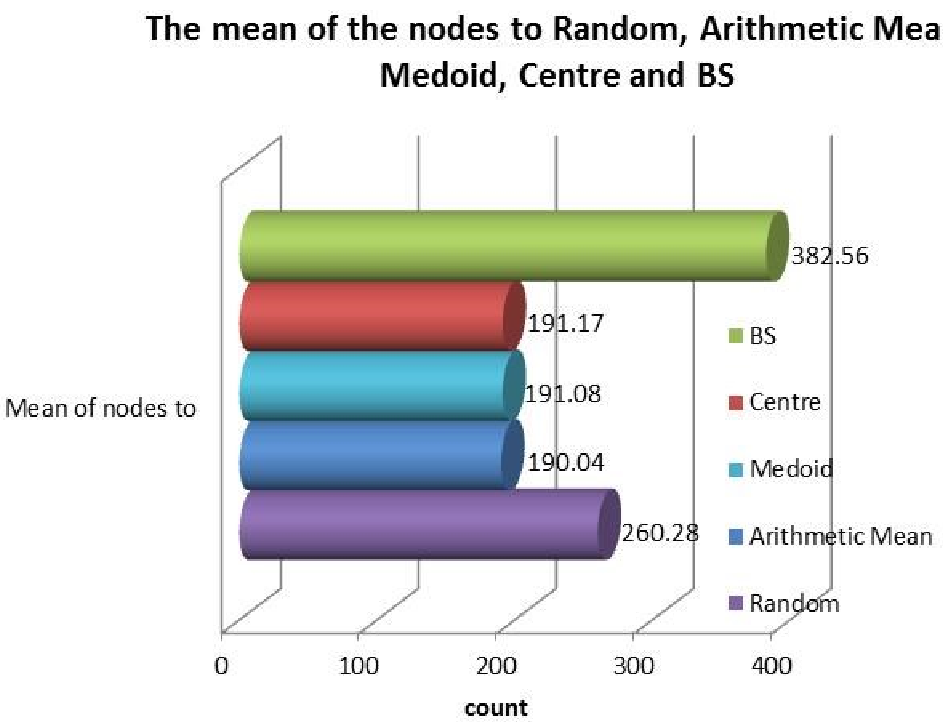

Fig. 5.10: A comparsin between the four mothed means to all the nodes (Centreoid, Centre, BS and Random).

A paired-samples t-test was conducted to compare the distance mean between the proposed method and *k*-mean. There was a significant difference in the scores for proposed method as shown in table (Table 5.5 and Table 5.6), where (M=188.7, SD= 19.88) and *k*-mean (M=319.44, SD= 178.76) conditions;

t (198) = 7.29, p = 0.00"

This mean that the nodes distance to the centroid using the proposed method is less than *k*-mean.

Furthermore, the same results were obtained deploying 1000 nodes, and their t-test result was:

t (198) = 11.93, p = 0.00

Table 5.5: t-test group statistics table for the proposed method and *k*-mean schemes of four cluster and 100 nodes.

T-Test Group Statistics					
4 Clusters	Group100N	N	Mean	Std. Deviation	Std. Error Mean
Experement1	proposed-Centroid	100	188.70	19.88	1.98
	K-mean-Centroid	100	319.44	178.76	17.87

Table 5.6: t-Test independent sample table proposed and *k*-mean schemes of four cluster and 100 nodes

Independent Samples Test								
4 Clusters 100 N		t-test for Equality of Means						
		t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference	
							Lower	Upper
Experement1	Equal variances assumed	-7.29	198	0.00	-130.74	17.98	-166.21	-95.27
	Equal variances not assumed	-7.29	101.45	0.00	-130.74	17.98	-166.41	-95.06

The p value is shown as 0.000 in all the experiments but this is not actual value because SPSS does not show very small values.

5.3.3 Dividing the Sensor Field into Eight Sectors

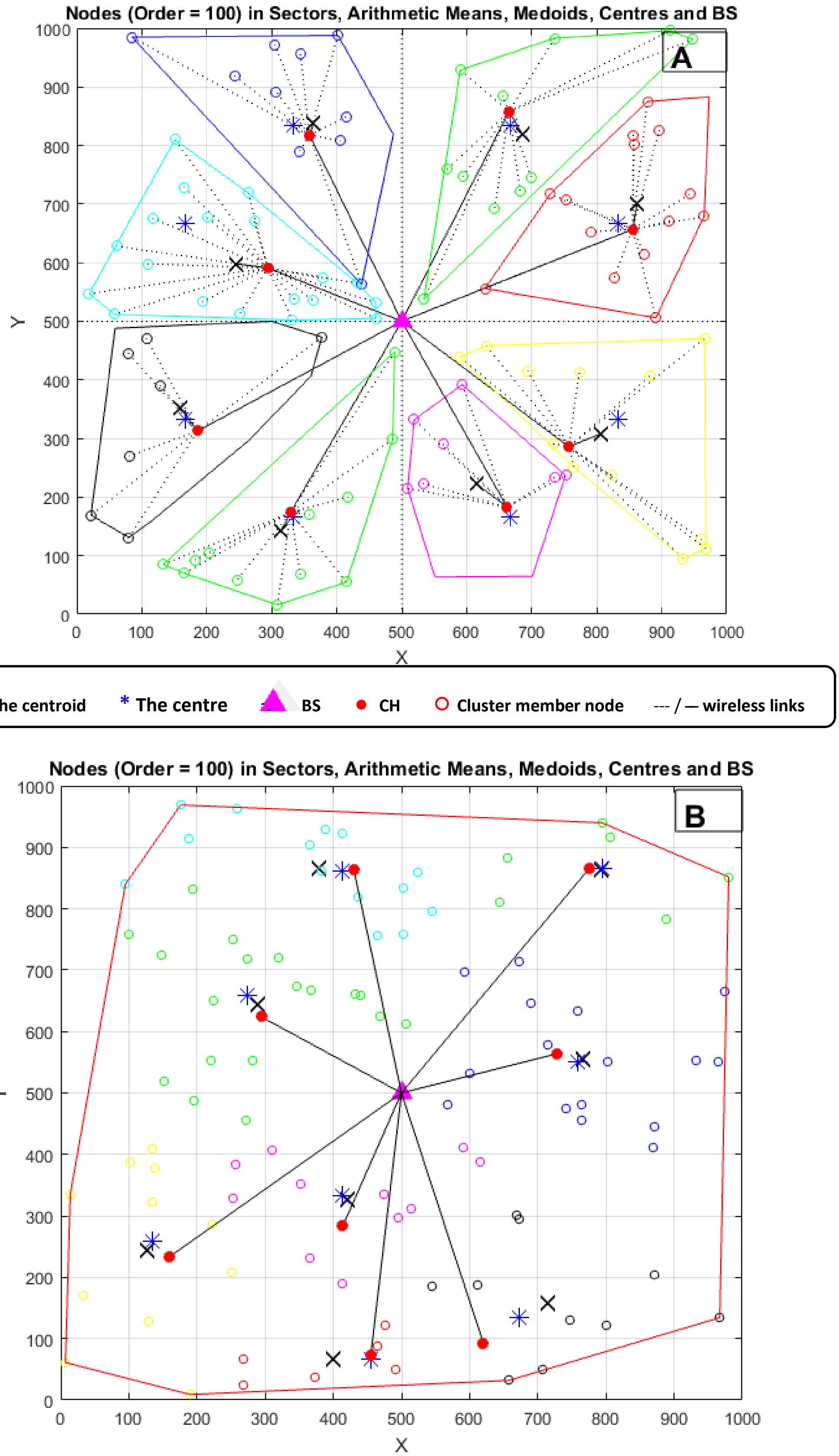

Fig. 5.11: Network base structure with 100 nodes including one CH for each cluster used in eight cluster base on; A: proposed method and B: *k*-means. A convex hull has been added to the perimeter to highlight the spatial extent of the nodes

Fig. 5.11 shows the division of the sensor field into eight sectors with the BS placed at the centre of the field. Where (A) represents the proposed scheme and (B) represents the *k*-mean scheme.

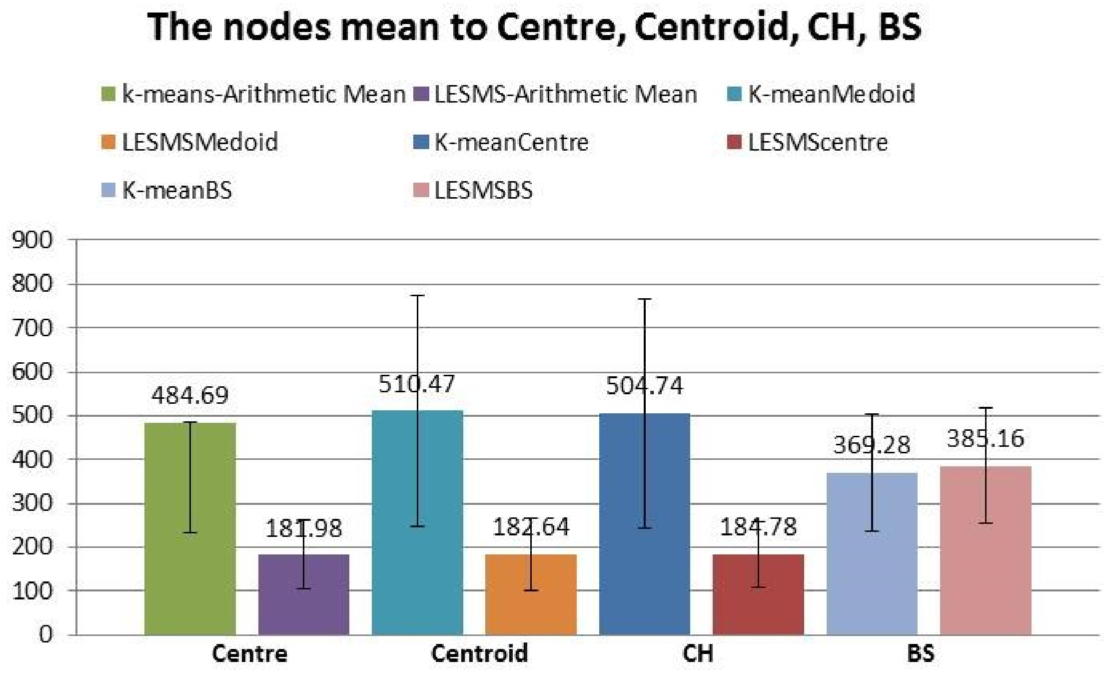

Fig. 5.12: A comparsin between proposed and *k*-mean mothed with regards of the mean of (Centreoid, Centre, CH) to all the nodes.

Another paired-samples t-test for eight sectors was conducted to compare the distance mean between the proposed method and *k*-mean. A significant difference was shown for proposed method as the t-test result was:

t (198) = 7.36, p = 0.00

The t-test result for 1000 nodes was:

t (198) = 10.49, p = 0.00

The P-value is (0.000), which is statistically significantly different and therefore, the null hypothesis is rejected.

5.4 DISCUSSION

Table 5.2: ANOVA table **comparison**, suggests that the P-value for the hypothesis stated in section 5.2.6 is 0.000. Therefore, there is a statistically high significant difference between the mean values of distances during the CH selection. The null hypothesis is then rejected, as the P-value is less than 0.05, which confirms that the mean values of the four schemes of the CH selection are different. Moreover, The Tukey test, Table 5.3: Multiple comparisons table (post-hoc tests), suggests that there are some statistically significant differences in the Mean values between the schemes used to select the CH,
Based on the tests results, it is clear that the location of the centre and the centroid in the middle of all the nodes provides the least distance mean. By contrast, the BS is placed at the furthest point of the network, which means it has the highest mean value of the sample test (383.0188). While the random scheme has a mean value higher than of the centre or centroid, but less than the value of the BS scheme. It may be possible, that the random CH selection is almost the same as the centre/centroid, but this is expected to be rare due to its nature.
The centroid scheme has shown the minimum distance for the entire node. Thus, this research will use the centroid as the main scheme to select the CH in the investigations that follow.
As for the division of the sensor field, Table 5.6 show that the proposed scheme has produced less distance mean than the *k*-mean scheme in both experiments (four and eight clusters). When there are more nodes involved in the network, the mean distance value is slightly smaller for the proposed scheme. Whereas, for the *k*-mean scheme it is significantly higher.

There are also large differences in the SD and SEM. This could be the result of the mean in the proposed scheme is being constrained in the quadrant, whereas, the *k*-mean scheme is based on the node distribution.
Fig. 5.13 shows that the proposed scheme has achived the shortest distance to all the nodes from centre, centiod and CH in comparison with the *k*-mean scheme. However, both schemes have similar values to BS, as the BS is fixed in the centre of the field.

CHAPTER 6:

LESMS

6.1 INTRODUCTION

At the early stages of using the NS2, the experiments were very simple where the main focus was to create a simple scenario with two nodes connected to each other with wire. Then, scenarios were devolved for multiple nodes. After that, the wireless scenarios were added. Having a wireless multi nodes scenarios has enabled this research to perform experiments using routing protocols such as, *Ad-hoc* On-Demand Distance Vector Routing (AODV), Destination Sequenced Distance Vector (DSDV), and Dynamic Source Routing (DSR). Various researches works have attempted to present comparisons between the different types of routing protocols based on packet delivery fraction (Pdf), Average End to End Delay and Routing Load.

Further research, aimed at providing a step by step comparative analysis of the three WSN routing protocols: AODV, DSDV and DSR, was carried out. Experiments were performed on each protocol to study three main aspects; the End to End Delay, the Jitter, and the Throughput. Each factor was executed with three scenarios, as most of the current studies and researches apply. These three scenarios were based on 16, 50, and 100 nodes. Results and discussions of this study were published in 2015 (Al-Baadani and Yousef, 2015). As a result, AODV was chosen as the base routing protocol for the investigations that follow.

The energy consumption was a negative effect on the proposed protocol LESMS (Location Energy Sleep Mode Saving) at the first stage. This was then enhanced by implementing the sleep mode for all the member nodes in the cluster when no sense or duty to perform. In addition, energy was used as a parameter to select the CH. Finally, a CH assistant technique was adopted to enable all duties of current CH to be transferred to the assistant CH before the CH reaches 20 % of its energy level.

6.2 METHOD

Experiments to test the network under different utilisation load environments were carried out in different inter-arrival times (1, 0.75, 0.5, 0.25, and 0.1). The inter-arrival time is the main influence in controlling the load of the network; this would have a direct impact on the total number of the sent packets every second. Sent packets can be calculated as, one divided by the inter-arrival value. This had allowed the research to demonstrate how the proposed protocol can work under a high load of exchanged messages between all the nodes in a network. Smaller inter-arrival times, means the network becomes busier, in other words, an inter-arrival time value of 1 represents a very quiet network where only one packet to be send every 1 millisecond, whereas an inter-arrival times of 0.1, involve each node sending 10 packets every 1 millisecond, represents the busiest network in these experiments.

6.2.1 The Sleeping Mode Technique

The limited source of power (battery) for the node in WSN has given the motivation to take this research further to find what other techniques can be used to obtain an optimum solution for the energy consumption as per the literature review. As a result, the sleeping mode technique, when member nodes are not in use, implementation (Norouzi et al., 2011) has been added to this research. This has allowed the proposed protocol to achieve respectable energy consumption from the sleeping modes, along with the optimum CH selection obtained previously.

In the sleeping mode, only the CH node needed to be active all the time, in order to send and receive messages, from all other nodes in its cluster, and forward them to the BS. However, all the other nodes in the cluster need only to be active when they sense data. The sensing device in each node would not go into a sleep mode. This would enable the nodes to wake up and send messages to the CH with the detected data. As soon as this task is completed, the nodes would go back to the temporary sleeping mode again.

Significant amount of energy in the network would be saved by using this technique as verified by the simulation results.

Having this feature does not consume a huge amount of energy; as most of the energy is used in the node processor and the radio parts, which would go into the sleep mode while, the main part, which senses the data, is ready all the time (Jurdak et al., 2010). Fig. 6.1 show the flow chart of the sleeping mode.

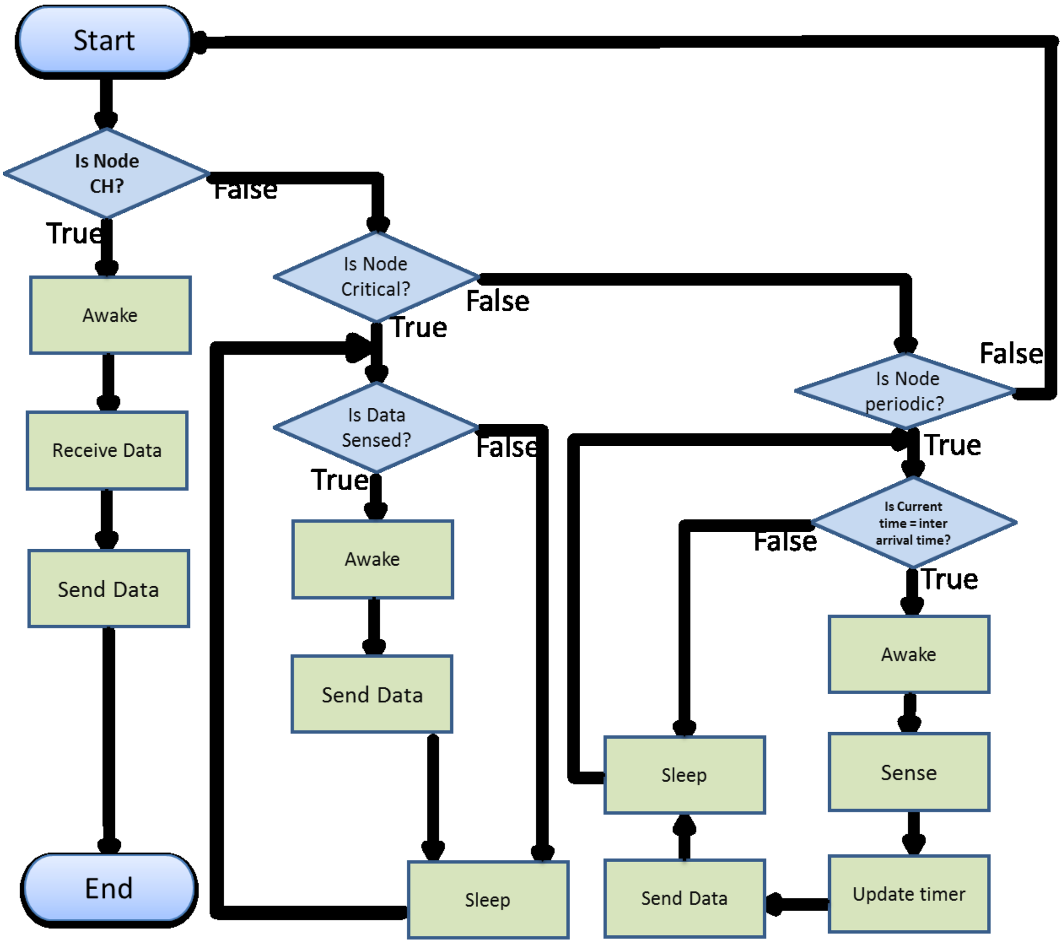

Fig. 6.1 Flow Chart for the sleeping mode system

There are three mode conditions available for the node to decide on which mode it should be in, these are:

a) If the node is a CH itself, then it is to remain awake all the time, and stay ready to receive data from the other cluster's nodes and forward it to the BS.

b) If the node type is critical, which means that the node might have had sensed something, in which case, it needs to send the information to the CH. The sensed data need to be verified, and if it were valid, the node would go into the wake-up mode, and then be able to send this data to the CH. However, if this information were not verified and not valid, the node would remain in its sleeping mode until it senses something. (Deng et al., 2005)

c) If the node type is a periodic node, the node's sensor would have a fixed time to go into awake-mode, sense the environment, and then send the data to the CH. This scenario can be used in an environment where the sensor network is used to study a room temperature at specific times. For such scenario, the node would first compare the current time with the interval time that has been set for the node to be awake (for instance every one hour). If the times were the same, the node would go into the awake-mode and then completes its tasks of sensing and sending the information to the CH. If the times were not the same, the node would remain in its sleeping mode until the current time matches the interval time.

This has resulted in the network to:

- Have less communication overhead between nodes.
- Have short distance communication.
- Have energy fairness amongst all the nodes.
- Be able to save the batteries' life, thus prolongs the network life.
- Be more reliable.

6.2.2 Energy Level

This part of the research is focusing on the implementation of a power parameter (energy level), as well as the previous technique used to select the optimum CH based on the closest node to arithmetic mean. Therefore two parameters are used in this study to select the optimum CH.

The proposed method uses the average energy technique, where the indicator of the minimum battery energy level is required form all the nodes, in order to select a CH, as suggested by researchers (e.g. Mahajan et al., 2014), as well as the closest node to the arithmetic mean. As a result, a score value of the two different parameters can be determined. The average energy of all the nodes is represented by Eq. xxx:

$$\boldsymbol{EAVG_v = \frac{1}{N} \sum_{v \in Nv} E_v} \qquad \text{Eq. 6.1}$$

Where E_v represent the residual energy of all the cluster nodes.

Knowing the average energy level value required, as well as the minimum value of distance to the centroid, which can be determining using the Eq. 5.1, allows the sore formula to be calculated.

$$\boldsymbol{score_v = \alpha . d_v + \beta . EAVG_v} \qquad \text{Eq. 6.2}$$

Where α + β are the parameters weighting factors and they are both = 1.

In some circumstances, for example, In case of an emergency, the network needs to reform itself suddenly, and in doing so, it tries to back up, this would consume a lot of energy.

6.2.3 Assistant CH and Health Check-Up System

This research includes the implementation of an assistant CH (ACH) that can take over the main CH duties when the CH battery reaches low levels.

This technique works on the basis of:

- When more than two nodes are having the same energy level, the one with the closest distance to the arithmetic mean is going to declare itself as the CH and the second closest distance to the arithmetic mean is going to be the ACH.
- The node with a second heights energy level and closest to the arithmetic mean is going to be set as the ACH.

Another feature of the energy techniques, the CH's health check-up, was used in this research. The CH's health check-up system monitors the energy level of the CH, once it is below the required level, 20 % or less Fig. 6.2 Energy Level, the CH passes its duties to the ACH before it dies. This would help to save network's energy that needed to reselect a CH.

The ACH would declare itself as the new CH and would be responsible for all the duties that the CH used to handle. The CH would become a normal member node.

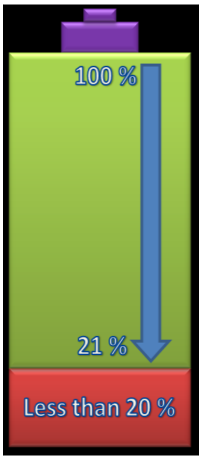

Fig. 6.2 Energy Level

Fig. 6.3 shows the process of the CH health check-up.

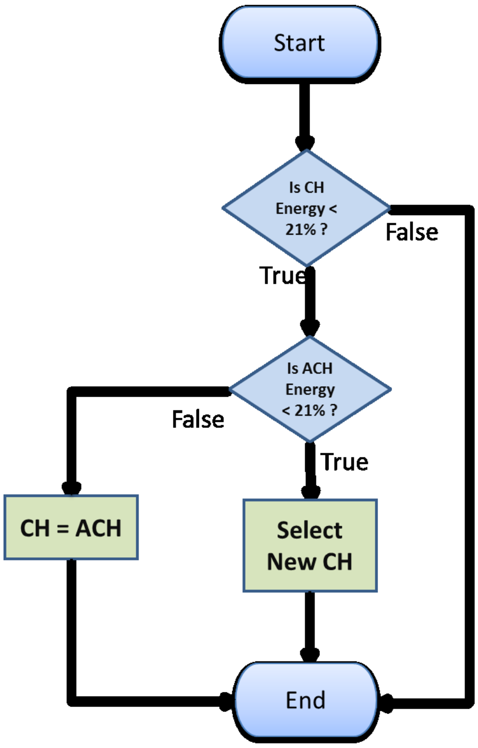

Fig. 6.3 Flow chart for CH health check-up

In the case of not being able to locate the ACH, such as, had been moved to another cluster or destroyed, the optimum option would be to reselect the CH again.

One of the worst-case scenarios would be, if the energy levels of all the nodes fell below 21 %. The node closest to the arithmetic mean would then be chosen as the new CH.

6.2.4 Assumptions

In addition to the assumptions stated in section 5.2.1 the following are assumed:

- Nodes are mobile;
- Nodes are powered by batteries;
- Nodes are physically similar with the same type of batteries;
- CH to consume more energy than an ordinary node;
- Nodes to wake-up and send when the sensor detects data;

6.2.5 Apparatus

All simulations presented in this chapter were created in NS2 (Network simulator two). AODV was used as the base protocol for this investigation. TCL (Tool Command Language) was used to create the scenarios in NS2 while AWK (Aho, Weinberger, and Kernighan) language was used to extract the result from the output file. NAM (Network AniMator developed as part of VINT project), was used as a tool for visualising the output.

6.2.6 Procedure

In addition to the procedure stated in section 5.2.3,

- Setup the mobile node parameters. This is essential to define, the energy model and mobility node's features;
- Define agents to setup a UDP connection;
- Define applications to setup a CBR application over the UDP connection.
- Repeat each simulation 20 times to obtain statistically representative behaviour for analysis. Average of the result iterations, standard deviation, and standard error was used to draw the result figures.

Table 6-1 Simulation Parameters

Parameters	Values
Channel	Channel/WirelessChannel
Radio Propagation Model	Propagation/TwoRayGround
Network interface	Phy/WirelessPhy
MAC	Mac/802_11
Interface Queue	Queue/DropTail/PriQueue
link layer	LL
Antenna model	Antenna/OmniAntenna
Mobility	Random
Interface Queue Length	50
Routing protocol	AODV/LESMS
Packet Size	512
Time of simulation	100s

6.2.7 Network Characteristic Parameters

Parameters used to analyse the result.

a) Delay
b) Throughput
c) Jitter
d) Packet Delivery Ratio (PDR)
e) Control Overhead
f) Numbers of Packets Sent
g) Numbers of Packets Dropped
h) Average Energy Consumption
i) Average Residual Energy
j) Total Energy Consumption
k) Total Residual Energy

6.3 OVERALL RESULT AND ANALYSIS FOR AODV AND LESMS

A comparison of the delay parameter between the AODV and the LESMS protocols is presented in Fig. 6.4. As it can be noticed, that the LESMS has been enhanced to reach its best performance with respect to delay, where it started at 0.25 milliseconds and gradually increased until it reached 0.35 milliseconds at the inter-arrival time of 0.75 ms. It can be analysed from the graph that the LESMS has performed better than AODV. AODV best performance started at the inter-arrival time of 1 ms where the delay was 1.15 ms. The highest AODV delay was 1.6 millisecond at the inter-arrival time of 0.75.

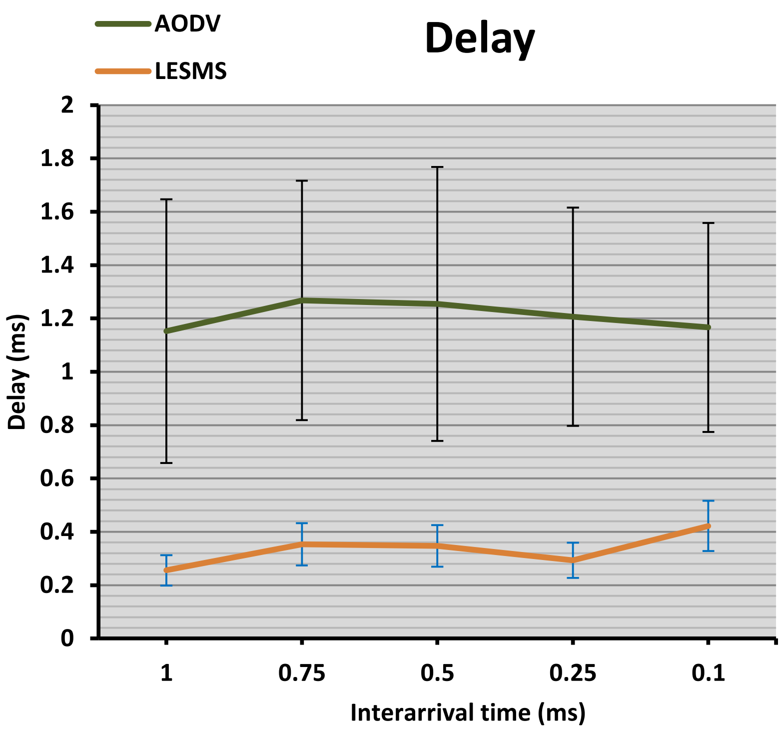

Fig. 6.4 Delay comparison between LESMS and AODV protocols

Fig. 6.5 shows the throughput comparison of the AODV and the LESMS protocols. As it can be observed from the graph, that the throughput is 8.44 kbps at the inter arrival time of 0.75 then it slightly decreased until it reached 7.27 kbps at the inter arrival time 0.25. As for the AODV the throughput is almost half of LESMS throughput, where it is fluctuating between the values 3.99 kbps and 3.57 kbps.

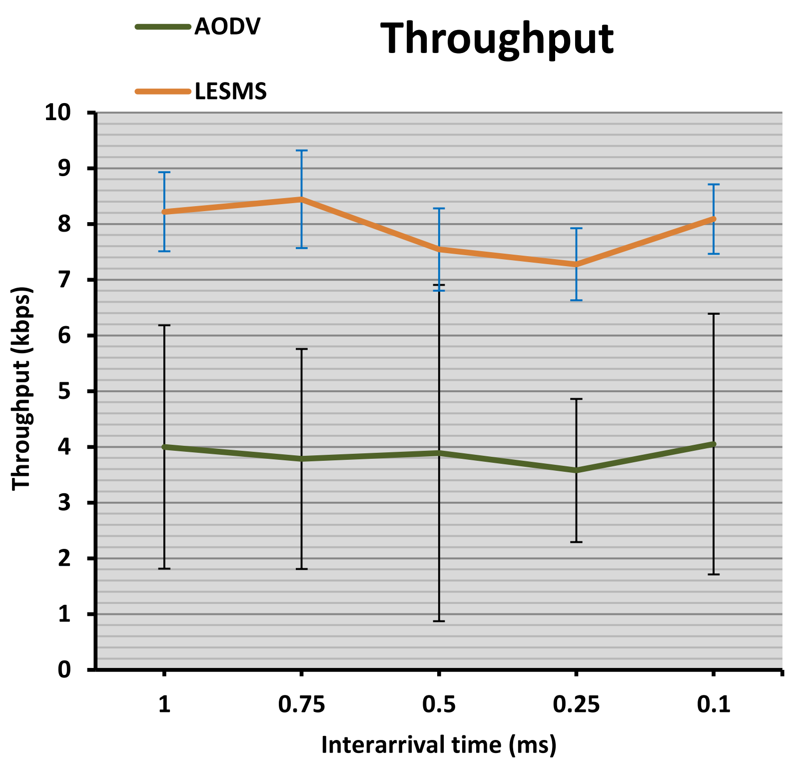

Fig. 6.5 Throughput comparison between LESMS and AODV protocols

Fig. 6.6 shows the Jitter comparison between the AODV and the LESMS protocols. The Jitter is defined as the difference in packet inter-arrival time to its destination, which has been measured in milliseconds. From the graph it can be observed, that the Jitter for AODV is fluctuating between 12 and 16 ms, whereas the LESMS figures are almost steady in between 0.04 and 0.06 ms, throughout the experiments.

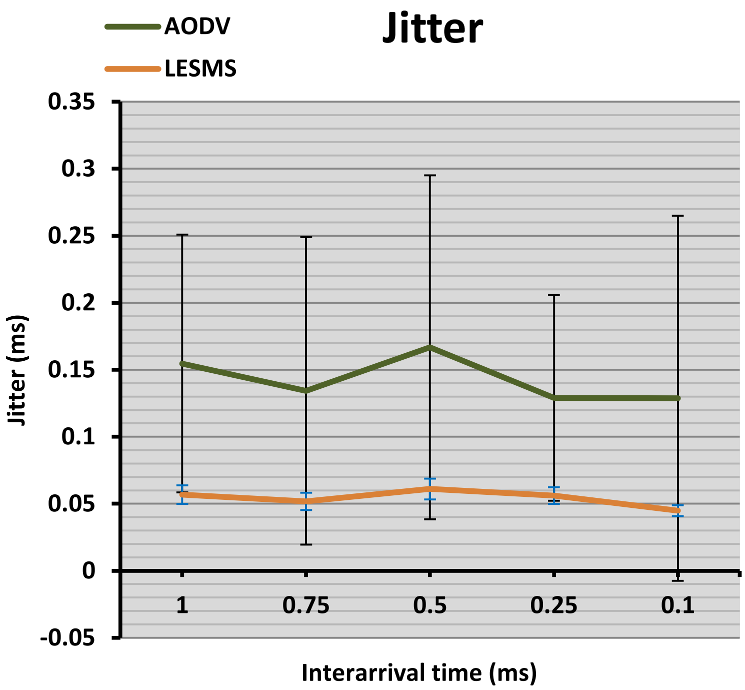

Fig. 6.6 Jitter comparison between LESMS and AODV protocols

Fig. 6.7 shows a comparison of the Packet Delivery Ratio (PDR) parameter. A high PDR value reflects a successful protocol in terms of Packets delivery; it takes the ratio of packets which has been delivered successfully to their destination and compare them with the number of packets that have actually been sent by the sender (source). The graph shows that the PDR is decreasing whenever the inter-arrival time value is low. This is due to the high number of packets sent when the network is busy, thus the number of dropped packets is high.

The AODV clearly shows low PDR in comparison with the LESMS. The highest AODV PDR was 11 % at the first experiment. Where LESMS started with 83% up to 88.9% at the inter-arrival time of 0.25 ms, before a sharp drop to 28.4% at 0.1 ms.

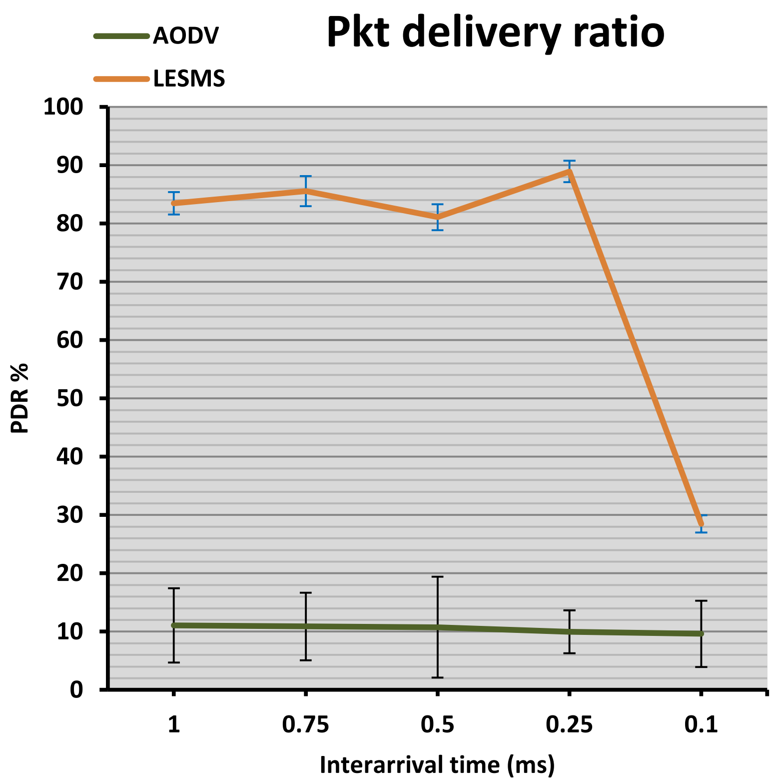

Fig. 6.7 Packet Delivery ratio comparison between LESMS and AODV protocols

Fig. 6.8 shows the control overhead for the AODV and the LESMS. The control overhead is the ratio of the routing packets to the delivered data packets.

AODV has shown a higher control overhead in the performance. At the inter-arrival time of 1, AODV started at 19400 as the highest value where LESMS at the same point was 11800 This leaves a difference of 6700 which is a significant difference for the network quality of services.

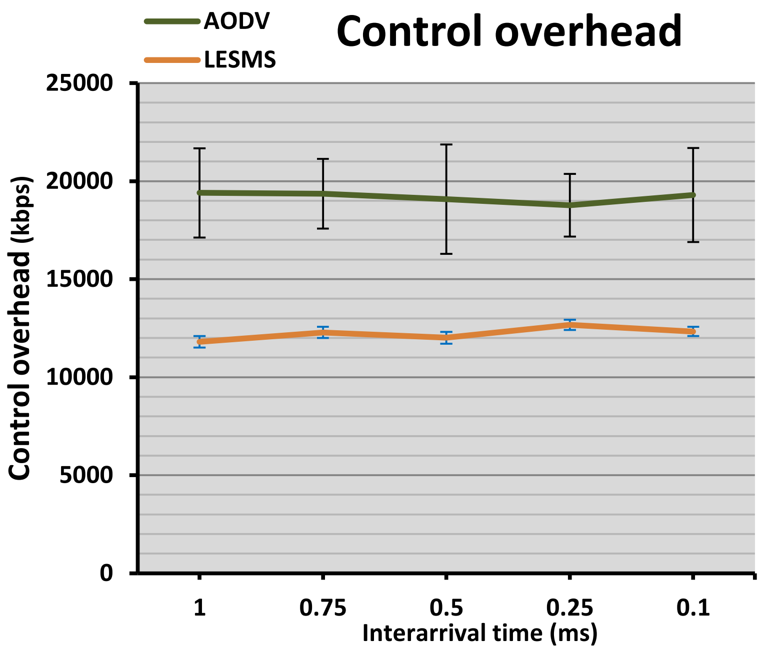

Fig. 6.8 Control overhead comparison between LESMS and AODV protocols

Fig. 6.9 and Fig. 6.10 show the number of packets sent and packets dropped respectively. The number of packets sent, should increase every time the inter-arrival times decrease. The number of packets sent is almost the same for both protocols this indicates that, both systems were performing in the same manners. However, the AODV showed a higher number of dropped packets in Fig. 6.10.

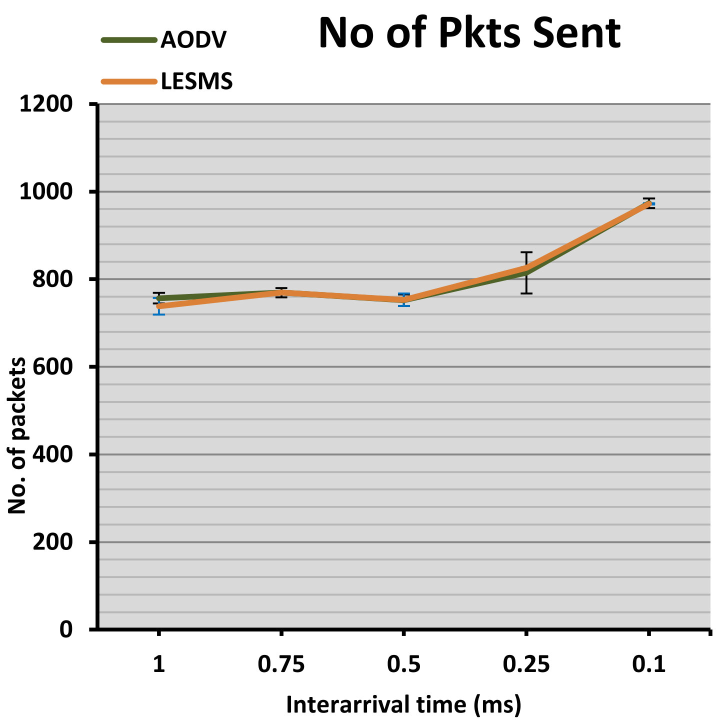

Fig. 6.9 Number of Packet sent comparison between LESMS and AODV protocols

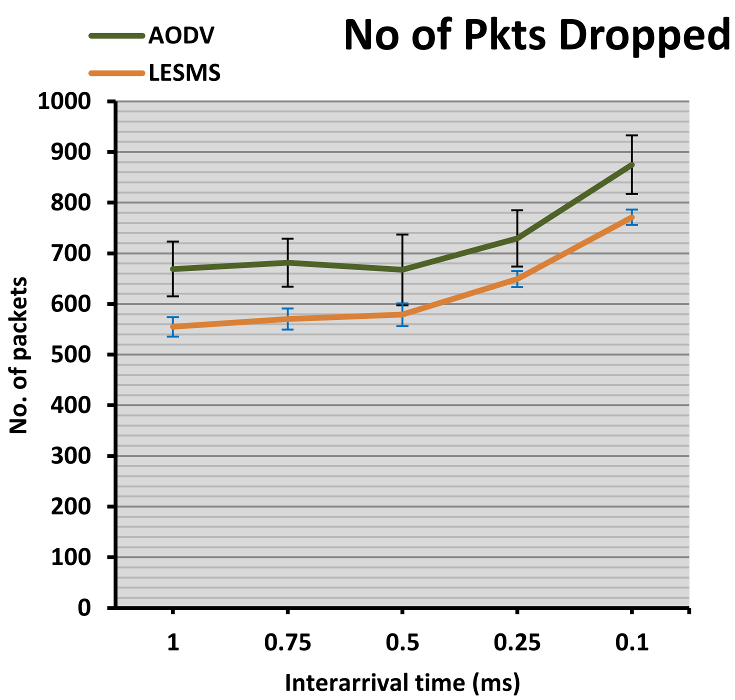

Fig. 6.10 Number of Packet dropped comparison between LESMS and AODV protocols

Fig. 6.11 to Fig. 6.14 show the average and total energy consumption and residual energy comparison for both AODV and LESMS.

The LESMS has outperformed the AODV in all workload conditions. As per the figures, the average energy consumption for the AODV, at the inter-arrival time of 1 ms, is 0.46 joules. The LESMS has a lower average energy consumption of 0.31 joules.

The differences can also be seen on the total residual energy, Fig. 6.14, and the average and total residual energy, Fig. 6.13.

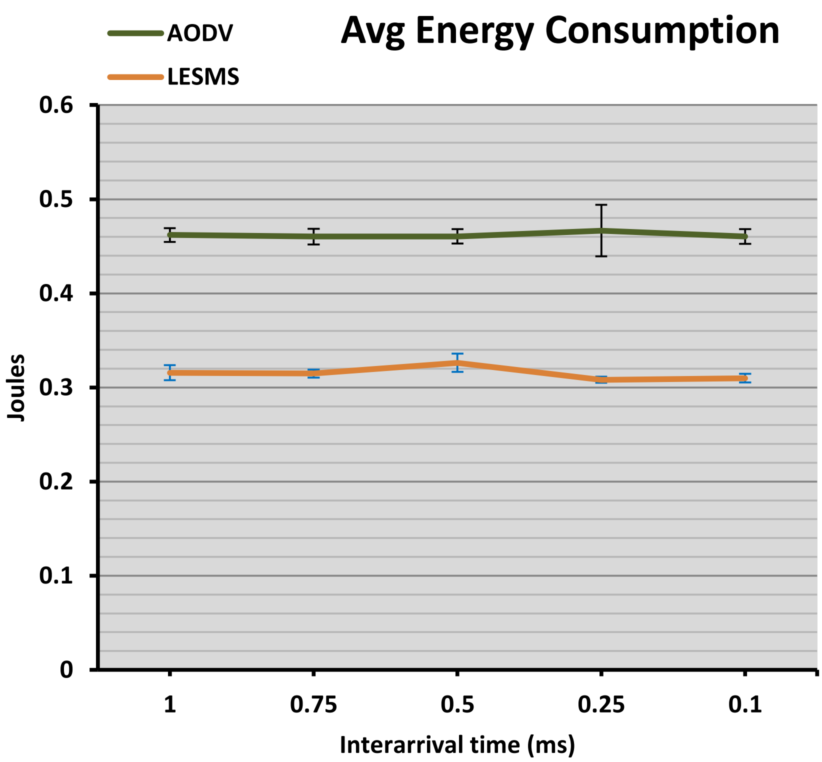

Fig. 6.11 Average energy consumption comparison for LESMS and AODV protocols

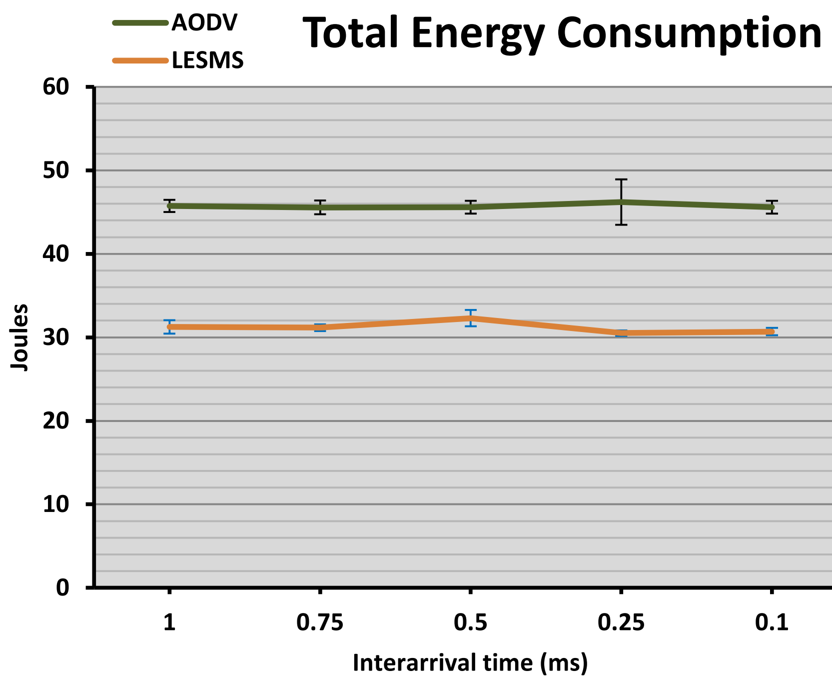

Fig. 6.12 Total energy consumption comparison for LESMS and AODV protocols

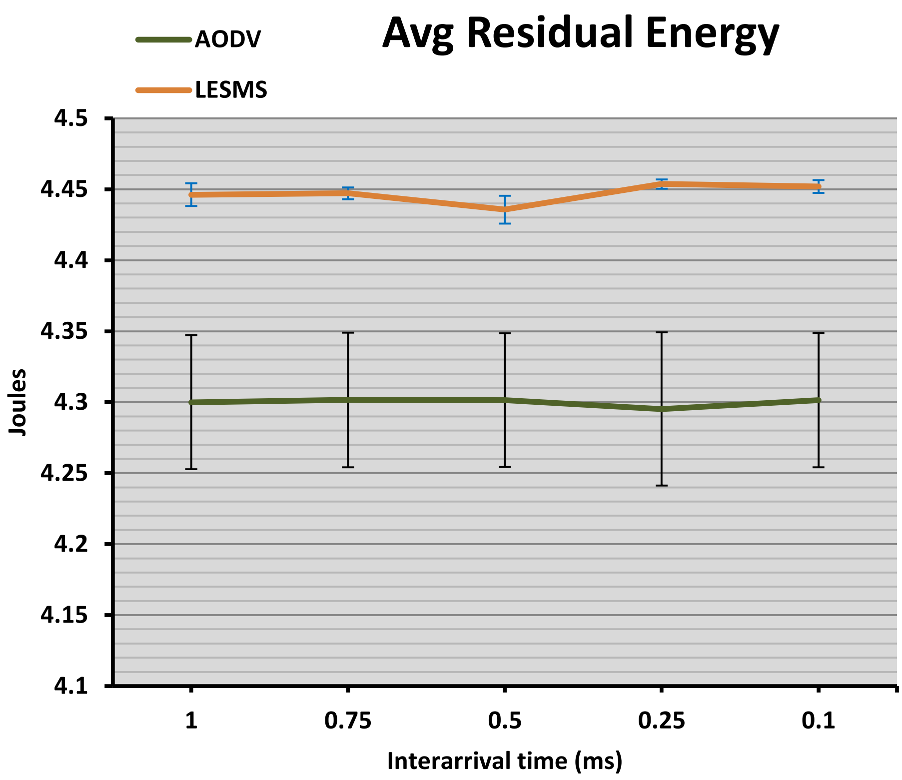

Fig. 6.13 Average energy residual comparison for LESMS and AODV protocols

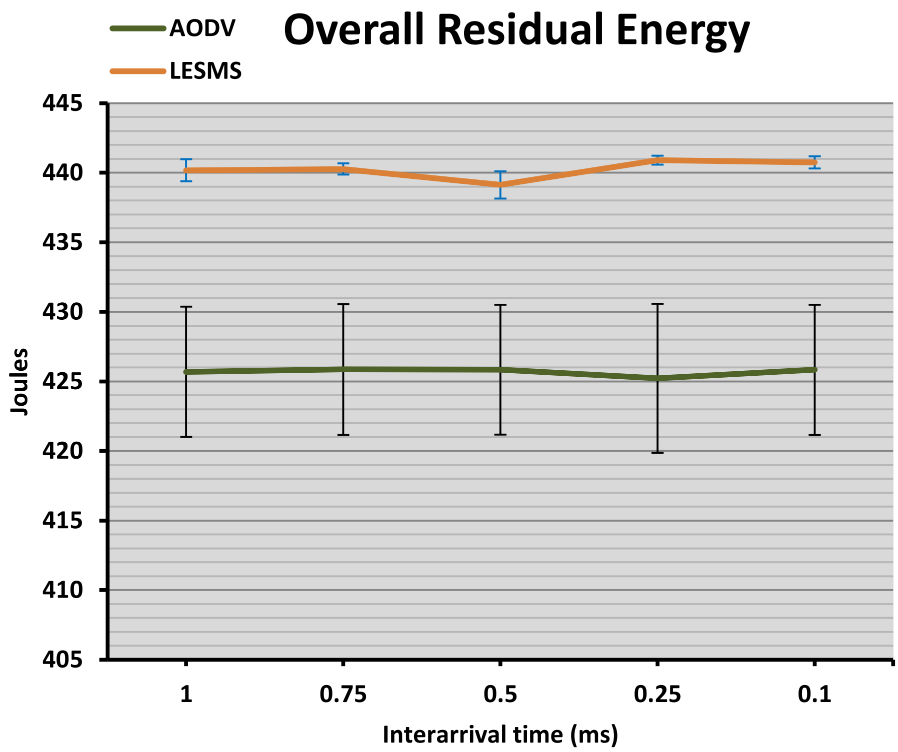

Fig. 6.14 Total energy residual comparison for LESMS and AODV protocols

6.4 DISCUSSION

As per the results, it can be concluded, that the AODV provides significant delay, particularly when the network is loaded and busy. The LESMS, undoubtedly, outperforms the AODV and provides better delay results at different inter-arrival times. It is normal for the control overhead to increase when more packets are circulating in the network, this was seen when the AODV was implemented, while the LESMS enhancements were implemented, a minimal impact in control the overhead was observed. This shows the significant advantage of the CH selection mechanism, especially in busy networks environment. Another positive aspect of the LESMS is that the throughput was uniform at various inter-arrival times.

Furthermore, the LESMS performed better in packet delivery at all inter-arrival time values, mostly when the network was busy. The WNS would work with a lower inter-arrival time due to the fact that the nodes only send packets to indicate they have sent something.

The average energy consumption and the average residual energy values were better when using the LESMS, which managed to improve the nodes' energy consumption by implementing the sleep mode for member nodes in the cluster. The increase of energy consumption, as the network is busier, is due to burstiness of traffic, which involves irregular increase and decrease of the packets.

CHAPTER 7:

CONCLUSION AND FUTURE WORK

7.1 SUMMARY

WSNs have become very popular in IoT and various real-time applications due to their ability to sense different environmental conditions in an *ad-hoc* manner. The WSN cluster-based topology consists of many cluster-heads, which act as routers in the wireless backbone architecture. Clustering in WSN is a proven technique to avoid redundant transmissions to the sink for better utilisation of scarce network resources such as energy. The process of cluster maintenance incurs message exchanges between nodes and cluster-heads. A high number of message exchanges leads to energy depletion of nodes which results in a decrease in the network lifetime. Chapter three of this thesis showed that most of the clustering algorithms proposed in the literature involved a high number of message exchanges, which resulted in unnecessary energy consumption. This thesis proposed the LESMS protocol, which uses the arithmetic mean to select the CH in a divided sensor field, to avoid such message exchanges and energy consumption.

Chapter four, described in detail the research methodology used in the simulation design and the implementation of LESMS. A brief description for different WSN simulation tools were listed as well as a description of the simulators used for this research. Furthermore, this research has benefited further from the theoretical work proposed in this thesis. This has allowed the research to be applied in real-life scenarios where hardware as well as the software were involved. The hardware experiments have demonstrated the single node architecture, which was an independent module that can sense, analyse and transmit information. The multi node architecture was demonstrated with the application of three sensor nodes connected to each other.

Chapter five compared different approaches for the CH selection which were then presented in the simulations and statically analysed. The approaches compared were: (i) random selection; (ii) selecting the node closest to the arithmetic mean node coordinate; (iii) selecting the node at the medoid coordinate; (iv) selecting the node nearest to the region centre; (v) selecting node nearest the BS. The mean distance of all nodes to their local CH was taken as the dependent variable. Moreover, three node grouping schemes were investigated: (i) all nodes in the same group; (ii) dividing the sensor field into four rectangular quadrants and allocating nodes accordingly; (ii) dividing the sensor field into eight sectors and allocating nodes accordingly. Assumptions and procedure for the experiments were discussed in details. In addition, the schemes proposed were compared to the *k*-means clustering. A full analysis of the obtained results, in light of the P-value, was provided. The P-values, examined using t-test, one-way ANOVA and post-hoc tests, were used to find the differences of the mean values between the schemes. The P-value showed a significant difference, which mean that the nodes distance to the arithmetic mean using LESMS is less than k-mean.

Due to the limitation of the power in WSN, LESMS was supported with the sleeping mode technique in order to overcome the excessive power consumption, as shown in chapter Six. This has required participating member nodes to go into a temporary sleep-mode when there is no data to be sensed, yet the sensing device in each node is to be awake at all times, to guarantee a reliable sensor network. Furthermore, the energy was introduced as a main factor in addition to the arithmetic mean in the CH selection process. This was achieved by taking the average residual energy form all the nodes as an indicator of the minimum battery energy level required for a node to be the CH. The average energy values as well as the minimum value of distance to the centroid were the parameters weighting factors to select the CH, and both values were equal to one. The node with highest score of both was selected as the CH.

The ACH concept was implemented. It allowed a second node to take over the main CH duties when the CH battery reaches low levels. The node with a second heights energy level and closest to the arithmetic mean was the criteria for selecting the ACH. The final stage of LESMS was the creation of the health check-up technique, which is a unique way of performing a periodic CH health check-up by monitoring the energy level of the CH. Once the CH energy level; reaches below the required level of 20 %, the CH passes its duties to the ACH before it dies. This saves the network’s energy that is needed to reselect a CH. The LESMS protocol performance was compared to the cluster AODV protocol, and experiments were performed under a mixture of workload environments. The results and analyses used the same comparative parameters, which showed a significant difference in favour of the LESMS. Using LESMS resulted in low power consumption, low delay, low packet drop, low control overheads and low throughput. It also showed high packet delivery ratio (PDR) and an increase in the total residual energy.

7.2 FUTURE WORK

The LESMS needs to consider more than two parameters in selecting the CH; such is mobility, which needed to be investigated in future researches. This would result in enhancing LESMS to prevent the expected negative impact of node mobility.

As for the practical work, there are many applications that can be considered as an extension using the mobile sensor node, for instance, in battle scenarios.

The practical aspect of this research has focused on the fire flame application, where the future work would consider different applications to be applied, such as:

- Home security systems that work on the infrared bases.
- Stores and environment temperature monitoring systems.
- Carbon dioxide monitoring system for industrial areas.
- Vegetable and fruit monitoring system, to help identify expiring products.

Moreover, a generic technique would be required as a method to decide on the optimal number of clusters and the maximum and minimum number of nodes in each cluster. This consideration would be valid for the purposes of maximizing energy efficiency.

WSNs research is full of opportunities to be considered, especially when using the advantage of the WSN technology to manage human activities in IoT and on smart home/city environments.

Practically, WSNs have been applied with limited applications scope, while the widespread vision of WSNs is to be applied on thousands or millions of nodes yet to be attained. Recent studies have shown that the WSN research in the near future will have a large influence on our daily life.

REFERENCES

Abbasi, A. A., & Younis, M. (2007). A survey on clustering algorithms for wireless sensor networks. *Computer Communications*, *30*(14), 2826-2841.

Afsar, M. M., & Tayarani-N, M. H. (2014). Clustering in sensor networks: A literature survey. *Journal of Network and Computer Applications*, *46*, 198-226.

Akkaya, K., & Younis, M. (2005). A survey on routing protocols for wireless sensor networks. *Ad-hoc Networks*, *3*(3), 325-349.

Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). Wireless sensor networks: a survey. *Computer Networks*, *38*(4), 393-422.

Al-Baadani, F. and Yousef, S., (2015). A Comparison Study for Different Wireless Sensor Network Protocols. In *Global Security, Safety and Sustainability: Tomorrow's Challenges of Cyber Security: 10th International Conference, ICGS3 2015,* (Vol. 534, p. 252). London, UK, 15-09-2015. UK: Springer.

Al-Karaki, J. N., Kamal, A. E., & Ul-Mustafa, R. (2004). On the optimal clustering in mobile ad hoc networks. In *Consumer Communications and Networking Conference, 2004, CCNC 2004* (pp. 71-76). Ames, IA, USA, 05-08-2004. Las Vegas, USA: IEEE.

Al-Karaki, J.N. and Kamal, A.E., (2004). Routing techniques in wireless sensor networks: a survey. *IEEE wireless communications, 11*(6), pp. 6-28. Jordan, 06-12-2004. IEEE

Amis, A. D., Prakash, R., Vuong, T. H., & Huynh, D. T. (2000). Max-min d-cluster formation in wireless ad hoc networks. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE* (Vol. 1, pp. 32-41). Dallas, TX, USA, 06-08-2002. Tel Aviv, Israel: IEEE.

Ammari, H.M. (2014). *The Art of Wireless Sensor Networks.* 1st ed. New York, USA: Springer, Berlin, Heidelberg.

Angione, G., Bellavista, P., Corradi, A., & Magistretti, E. (2006, July). A k-hop clustering protocol for dense mobile *ad-hoc* networks. In *26th IEEE International Conference on*

Distributed Computing Systems Workshops (ICDCSW'06) (pp. 10-10). Bologna BO, Italy, 24-06-2006. Lisboa, Portugal: IEEE.

Anker, T., Bickson, D., Dolev, D., & Hod, B. (2008). Efficient clustering for improving network performance in wireless sensor networks. In *Wireless sensor networks* (pp. 221-236). Israel, 01-02-2008. Bologna, Italy: Springer.

Arumugam, M. and Kulkarni, S.S., (2005). Self-stabilizing deterministic TDMA for sensor networks. , *in: Proceedings of the 2nd International Conference on Distributed Computing and Internet Technology (ICDCIT). 3816*, pp.69-81. East Lansing, USA, 2005. Bhubaneswar, India: Springer, Berlin, Heidelberg.

Baker, D., & Ephremides, A. (1981). The architectural organization of a mobile radio network via a distributed algorithm. *IEEE Transactions on Communications*, *29*(11), 1694-1701.

Bandyopadhyay, S., & Coyle, E. J. (2003, April). An energy efficient hierarchical clustering algorithm for wireless sensor networks. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies* (Vol. 3, pp. 1713-1723). West Lafayette, USA, 30-04-2003. San Francisco, CA, USA: IEEE.

Baronti, P., Pillai, P., Chook, V.W., Chessa, S., Gotta, A. and Hu, Y.F., 2007. Wireless sensor networks: A survey on the state of the art and the 802.15. 4 and ZigBee standards. *Computer communications*, *30*(7), pp.1655-1695.

Bhagwat, P., Perkins, C., & Tripathi, S. (1996). Network layer mobility: an architecture and survey. *IEEE Personal Communications*, *3*(3), 54-64.

Buratti, C., Conti, A., Dardari, D., & Verdone, R. (2009). An overview on wireless sensor networks technology and evolution. *Sensors*, *9*(9), 6869-6896.

Chatterjee, M., Das, S. K., & Turgut, D. (2002). WCA: A weighted clustering algorithm for mobile ad hoc networks. *Cluster Computing*, *5*(2), 193-204.

Chen, Y. P., & Liestman, A. L. (2003). A Zonal Algorithm for Clustering An Hoc Networks. *International Journal of Foundations of Computer Science*, *14*(02), 305-322.

Chen, Z., Xiao, Y., Li, X. and Li, R., (2013). A clustering protocol for wireless sensor networks based on energy potential field, *The Scientific World Journal*, vol. 2013, Article ID 829861, 7 pages.

Chinara, S., & Rath, S. K. (2009). A survey on one-hop clustering algorithms in mobile ad hoc networks. *Journal of Network and Systems Management*, *17*(1-2), 183-207.

Chlamtac, I., Conti, M., & Liu, J. J. N. (2003). Mobile ad-hoc networking: imperatives and challenges. *Ad-hoc Networks*, *1*(1), 13-64.

Chong, C. Y., & Kumar, S. P. (2003). Sensor networks: evolution, opportunities, and challenges. *Proceedings of the IEEE*, *91*(8), pp. 1247-1256.

Cui, S., Madan, R., Goldsmith, A. and Lall, S., (2005). Joint routing, MAC, and link layer optimization in sensor networks with energy constraints. In *Communications, 2005. ICC 2005. 2005 IEEE International Conference on* (Vol. 2, pp. 725-729). USA, 20/05/2005. Seoul, South Korea: IEEE.

Dargie, W. and Poellabauer, C., (2010). *Fundamentals of wireless sensor networks: theory and practice*. UK: John Wiley & Sons.

Das, S. R., Belding-Royer, E. M., & Perkins, C. E. (2003). *Ad hoc on-demand distance vector (AODV) routing.* Cincinnati, USA: The Internet Society.

Dash S.K., Sahoo J.P., Mohapatra S., Pati S.P. (2012) Sensor-Cloud: Assimilation of Wireless Sensor Network and the Cloud. In: *Meghanathan N., Chaki N., Nagamalai D. (eds) Advances in Computer Science and Information Technology. Networks and Communications. CCSIT 2012.* Bhubaneswar, India, 04-01-2012. Bangalore, India: Springer, Berlin, Heidelberg.

Deng, J., Han, Y.S., Heinzelman, W.B. and Varshney, P.K., 2005. Scheduling sleeping nodes in high density cluster-based sensor networks. *Mobile Networks and Applications*, *10*(6), pp.825-835.

Ding, P., Holliday, J., & Celik, A. (2005). Distributed energy-efficient hierarchical clustering for wireless sensor networks. In *international conference on Distributed*

Computing in Sensor Systems (pp. 322-339). Los Angeles, USA, 01-07-2005. USA: Springer Berlin Heidelberg.

Doherty, L., Simon, J. and Watteyne, T., (2012). Wireless sensor network challenges and solutions. *Microwave Journal*, *55* (8), (pp. 22-34).

Egea-Lopez, E., Vales-Alonso, J., Martinez-Sala, A. S., Pavon-Marino, P., & García-Haro, J. (2005). Simulation tools for wireless sensor networks. In *Proceedings of the International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS05)* (p. 24). Cartagena, Spain, 2005.

El-Bazzal, Z., Kadoch, M., Agba, B. L., Gagnon, F., & Bennani, M. (2006, October). A flexible weight based clustering algorithm in mobile ad hoc networks. In *Systems and Networks Communications, 2006. ICSNC'06. International Conference on* (pp. 50-50). Montreal, Canada, 03-11-2006. Tahiti: IEEE.

Er, I. I., & Seah, W. K. G. (2004). Mobility-based d-hop clustering algorithm for mobile ad hoc networks. In *Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE* (Vol. 4, pp. 2359-2364). Singapore, 21-05-2004. Atlanta, USA: IEEE.

Eren, H. (2005). *Wireless sensors and instruments: networks, design, and applications*. NW, USA: CRC Press.

Finn, G. G. (1987). Routing and addressing problems in large metropolitan-scale internetworks. *University of Southern California Marina Del Rey Information Sciences Inst,* (No. ISI/RR-87-180). Marina Del Rey, USA, March-1987. Washington, USA: EDRS.

Fontanelli, D., Palopoli, L., & Passerone, R. (2009, April). Convergence of distributed wsn algorithms: The wake-up scattering problem. In *International Workshop on Hybrid Systems: Computation and Control* (pp. 180-193). Trento, Italy, 15-04-2009. San Francisco, USA: Springer Berlin Heidelberg.

Garcia-Macias, J. A., & Gomez, J. (2007). Manet versus wsn. In: *Sensor networks and configuration.* Ensenada, Mexico: Springer Berlin Heidelberg. pp. 369-388.

Guo, S. and Yang, O.W., (2007). Energy-aware multicasting in wireless ad hoc networks: A survey and discussion. *Computer Communications*, *30*(9), pp.2129-2148.

Haas, Z. J., & Pearlman, M. R. (1998, June). The performance of a new routing protocol for the reconfigurable wireless networks. In *Communications, 1998. ICC 98. Conference Record. 1998 IEEE International Conference on* (Vol. 1, pp. 156-160). NY, USA, 11-06-21998. USA: IEEE.

Han, X., Cao, X., Lloyd, E. L., & Shen, C. C. (2010). Fault-tolerant relay node placement in heterogeneous wireless sensor networks. *IEEE Transactions on Mobile Computing*, *9*(5), 643-656.

Hazewinkel, Michiel, ed. (2001), "Quadratic deviation" (http://www.encyclopediaofmath.org/index. php?title=p/q076030), Encyclopedia of Mathematics, Springer, ISBN 978-1-55608-010-4).

He, F., & Wang, F. (2007, September). A cost sensitive best network selection scheme in heterogeneous wireless networks. In *2007 International Conference on Wireless Communications, Networking and Mobile Computing* (pp. 1753-1756). Hubei Sheng, China, 25-09-2007. Shanghai, China: IEEE.

Heinzelman, W. B., Chandrakasan, A. P., & Balakrishnan, H. (2002). An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on Wireless Communications*, *1*(4), 660-670.

Hudedagaddi, D., Ravishankar, P., Rakesh, T. M., & Dengi, S. Comparison of WCA with AODV and WCA with ACO using clustering algorithm. In: *IPASJ International Journal of Computer Science (IIJCS).* Volume 3, Issue 7.

Intanagonwiwat, C., Govindan, R., Estrin, D., Heidemann, J., & Silva, F. (2003). Directed diffusion for wireless sensor networking. *IEEE/ACM Transactions on Networking*, *11*(1), 2-16.

Issariyakul, T. and Hossain, E., (2011). *Introduction to network simulator NS2*. .2nd ed. New York, USE: Springer Science & Business Media.

Jarchlo, E. A., & Bazlamaçci, C. F. (2014). Life Time Sensitive Weighted Clustering on Wireless Sensor Networks. In *SENSORNETS* (pp. 41-51).

Jiang, M., Lee, J., Jeong, K., Cui, Z., Kim, B., Hwang, S. and Choi, Y.J., (2015). A data stream-based, integrative approach to reliable and easily manageable real time environmental monitoring. *International Journal of Distributed Sensor Networks*, *11*(12), p.914612.

Jin, Y., Wang, L., Kim, Y., & Yang, X. (2008). EEMC: An energy-efficient multi-level clustering algorithm for large-scale wireless sensor networks. *Computer Networks*, *52*(3), 542-562.

Jindal, V., & Bawa, S. (2011). How the two Adhoc networks can be different: MANET & WSNs. Punjab, India, Vol. 2, Issue 4.

Joa-Ng, M., & Lu, I. T. (1999). A peer-to-peer zone-based two-level link state routing for mobile ad hoc networks. *IEEE Journal on Selected Areas in Communications*, *17*(8), 1415-1425.

Johnson, D. B., & Maltz, D. A. (1996). Dynamic source routing in ad hoc wireless networks. In *Mobile Computing* (pp. 153-181). US: Springer.

Johnson, D., Hu, Y., & Maltz, D. (2007). *The dynamic source routing protocol (DSR) for mobile ad hoc networks for IPv4* (No. RFC 4728).

Jurdak, R., Ruzzelli, A.G. and O'Hare, G.M., (2010). Radio sleep mode optimization in wireless sensor networks. IEEE Transactions on Mobile Computing, 9(7), pp.955-968.

Jurdak, R., Ruzzelli, A.G. and O'Hare, G.M., 2010. Radio sleep mode optimization in wireless sensor networks. *IEEE Transactions on Mobile Computing*, *9* (7), pp.955-968.

Kaebeh, Y.S.B, Soni, M.K. and Tyagi, S.S., (2016). Dynamic and real-time sleep schedule protocols for energy efficiency in WSNs. *International Journal of Computer Network and Information Security*, *8*(1), pp.9-17.

Kang, M.J., Jeong, S., Yoon, I. and Noh, D.K., (2017). Energy-aware determination of compression for low latency in solar-powered wireless sensor networks. *International*

Journal of Distributed Sensor Networks, [e-journal] *13* (2), p. 155. https://doi.org/10.1177/1550147717694168.

Kanungo, T., Mount, D.M., Netanyahu, N.S., Piatko, C.D., Silverman, R. and Wu, A.Y., 2002. An efficient k-means clustering algorithm: Analysis and implementation. *IEEE transactions on pattern analysis and machine intelligence*, 24(7), pp.881-892.

Karimi, A., Afsharfarnia, A., Zarafshan, F., & Al-Haddad, S. A. R. (2014). A novel clustering algorithm for mobile ad hoc networks based on determination of virtual links' weight to increase network stability. *The Scientific World Journal*, Volume 2014 (2014), Article ID 432952, 11 pages.

Karp, B., & Kung, H. T. (2000, August). GPSR: Greedy perimeter stateless routing for wireless networks. In *Proceedings of the 6th annual international conference on Mobile computing and networking* (pp. 243-254). Boston, USA, 06-08-2000. NY, USA: ACM.

Keh, H. C., Huang, N. C., Wang, Y. H., & Cheng, Y. C. (2011). The optimal sleep control for Wireless Sensor Networks. In *Electric Information and Control Engineering (ICEICE), 2011 International Conference on* (pp. 5521-5524).

Khan, Z.A. and Sampalli, S., (2012). AZR-LEACH: an energy efficient routing protocol for wireless sensor networks. In: *International Journal of Communications, Network and System Sciences,* 5(11), (pp. 785 -795). http://dx.doi.org/10.4236/ijcns.2012.511082.

Kim, N., Heo, J., Kim, H. S., & Kwon, W. H. (2008). Reconfiguration of clusterheads for load balancing in wireless sensor networks. *Computer Communications*, *31*(1), 153-159.

Ko, Y. B., & Vaidya, N. H. (2000). Location-Aided Routing (LAR) in mobile ad hoc networks. *Wireless Networks*, *6*(4), 307-321.

Kothari, D., Rao, D. S., & Singh, A. (2014). An Efficient Clusterhead Election Algorithm for Mobile Ad-Hoc Network. In: *International Journal of Computer Applications.* Volume 99– No.19.

Kulkarni, S.S. and Arumugam, M., (2006). SS-TDMA: A self-stabilizing MAC for sensor networks. In *Sensor network operations*, CHAPTER 1, pp.186-218. [pdf] Available at:

<https://www.researchgate.net/profile/Sandeep_Kulkarni6/publication/228348121_SS-TDMA_A_self-stabilizing_MAC_for_sensor_networks/links/00463534fe0f4217ef000000.pdf> [Accessed 07 November 2017].

Kumar, S., & Shepherd, D. (2001, August). SensIT: Sensor information technology for the warfighter. In: *Proc. 4th Int. Conf. on Information Fusion* (pp. 1-7). Arlington, USA, 06-2001. USA: DARPA.

Kurkowski, S., Camp, T., & Colagrosso, M. (2005). MANET simulation studies: the incredibles. In: *ACM SIGMOBILE Mobile Computing and Communications Review*, *9*(4), 50-61.

Letswamotse, B., Modieginyane, K. and Malekian, R., (2017). SDN Based QoS Provision in WSN Technologies. In: *Southern Africa Telecommunication Networks and Applications Conference (SATNAC).* Hatfield, South Africa, 27-02-2017. George, South Africa: DBLP.

Li, G., & Znati, T. (2007). RECA: a ring-structured energy-efficient clustering architecture for robust communication in wireless sensor networks. *International Journal of Sensor Networks*, *2*(1-2), 34-43.

Liao, W. H., Sheu, J. P., & Tseng, Y. C. (2001). GRID: A fully location-aware routing protocol for mobile ad hoc networks. *Telecommunication Systems*, *18*(1-3), 37-60.

Lin, C. R., & Gerla, M. (1997). Adaptive clustering for mobile wireless networks. *IEEE Journal on Selected Areas in Communications*, *15*(7), 1265-1275.

Ling, H., & Znati, T. (2009, April). Energy efficient adaptive sensing for dynamic coverage in wireless sensor networks. In: *2009 IEEE Wireless Communications and Networking Conference* (pp. 1-6). Budapest, Hungary, 08-04-2009. USA: IEEE.

Liu, C., Wu, K., & Pei, J. (2005, September). A dynamic clustering and scheduling approach to energy saving in data collection from wireless sensor networks. In Sensor and Ad Hoc Communications and Networks (*SECON),* (Vol. 5, pp. 374-385). Santa Clara, USA, 29-09-2005. USA: IEEE.

Mahajan, S., Malhotra, J. and Sharma, S., 2014. An energy balanced QoS based cluster head selection strategy for WSN. *Egyptian Informatics Journal*, 15(3), pp.189-199.

Mai, K. T., Shin, D., & Choo, H. (2009, January). Toward stable clustering in mobile ad hoc networks. In *2009 International Conference on Information Networking* (pp. 1-3). Seoul, South Korea, 24-11-2009. Chiang Mai, India: IEEE.

Mainwaring, A., Culler, D., Polastre, J., Szewczyk, R., & Anderson, J. (2002, September). Wireless sensor networks for habitat monitoring. In*Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications* (pp. 88-97). Georgia, USA, 28-09-2002. NY, USA: ACM.

Mamalis, B., Gavalas, D., Konstantopoulos, C., & Pantziou, G. (2009). Clustering in wireless sensor networks. In: *RFID and Sensor Networks: Architectures, Protocols, Security and Integrations*. USA: ACM. Ch.12.

Mamun, Q., 2012. A qualitative comparison of different logical topologies for wireless sensor networks. In: *Sensors*, *12*(11), pp.14887-14913.

Manzoor, B., Javaid, N., Rehman, O., Akbar, M., Nadeem, Q., Iqbal, A. & Ishfaq, M. (2013), "Q-LEACH: A New Routing Protocol for WSNs", In Proceeding Computer Science, (vol. 19), (pp. 926-931). https://doi.org/10.1016/j.procs.2013.06.127.

Mármol, F.G. and Pérez, G.M., (2009, June). TRMSim-WSN, trust and reputation models simulator for wireless sensor networks. *In Communications, 2009. ICC'09. IEEE International Conference* (pp. 1-5). Murcia, Spain, 18/06/2009. Dresden, Germany: IEEE.

Matin, M. A., & Islam, M. M. (2012). Overview of wireless sensor network. In: Wireless Sensor Networks-Technology and Protocols. Brunei, Darussalam: INTECH Open Access Publisher. Ch.1.

Matin, M. A., & Rahman, M. N. (2011, May). Lifetime improvement of wireless sensor network. In *Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on* (pp. 475-479). Dhaka, Bangladesh, 29-05-2011. China: IEEE.

Khan, A., Tamim, I., Ahmed, E. and Awal, M.A. (2011). Multiple parameters based clustering (MPC): Prospective analysis for effective clustering in wireless sensor network (WSN) using K-Means algorithm. In: *Wireless Sensor Network*, (pp. 18-24). [e-journal] Vol. 4 No. 1, 2012, pp. 18-24. doi:10.4236/wsn.2012.41003.

Moghadam, M. N., Zare, A., & Taheri, H. (2011, May). A self-organized communication efficient clustering algorithm in wireless sensor networks. In *2011 19th Iranian Conference on Electrical Engineering* (pp. 1-6). Tehran, Iran, 19-05-2011. Iran: IEEE.

Murthy, C. S. R., & Manoj, B. S. (2004). *Ad hoc wireless networks: Architectures and protocols, portable documents*. NJ, USA: Prentice Hall PTR.

Muthuramalingam, S., RajaRam, R., Pethaperumal, K., & Devi, V. K. (2010). A dynamic clustering algorithm for manets by modifying weighted clustering algorithm with mobility prediction. *International Journal of Computer and Electrical Engineering*, *2*(4), p. 709.

Myers, C., Oppenheim, A., Davis, R., & Dove, W. (1984, March). Knowledge based speech analysis and enhancement. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'84.* (Vol. 9, pp. 162-165). Cambridge, MA, USA, 21-03-1984. San Diego, USA: IEEE.

Nagpal, R., & Coore, D. (1998). An algorithm for group formation in an amorphous computer. In *Proc. 10th International Conference on Parallel and Distributed Computing Systems (PDCS'98)*. Cambridge, MA, USA, 1998. USA: DARPA.

Nassuora, A. B., & Hussein, A. R. H. (2011). CBPMD: a new weighted distributed clustering algorithm for Mobile Ad Hoc Networks (MANETs). *American Journal of Scientific and Industrial Research*, *22*, 43-56.

Norouzi, A., Babamir, F. S., & Zaim, A. H. (2011). A new clustering protocol for wireless sensor networks using genetic algorithm approach. *Wireless Sensor Network*, *3*(11), 362-370.

Ohta, T., Inoue, S., & Kakuda, Y. (2003, April). An adaptive multihop clustering scheme for highly mobile ad hoc networks. In *Autonomous Decentralized Systems, 2003. ISADS*

2003. The Sixth International Symposium on (pp. 293-300). Univ, Japan, 11-04-2003. Pisa, Italy: IEEE.

Panichpapiboon, S., Ferrari, G. and Tonguz, O.K., (2006). Optimal transmit power in wireless sensor networks. IEEE Transactions on Mobile Computing, 5(10), pp.1432-1447. Pittsburgh, USA, 28-08-2006. USA: IEEE.

Parkinson, B. W., & Gilbert, S. W. (1983). NAVSTAR: Global positioning system—Ten years later. *Proceedings of the IEEE*, *71*(10), 1177-1186. Cambridge, MA, USA, 10-1983. USA: IEEE.

Perkins, C. E. (2008). *Ad hoc networking*. 1st ed. Boston, USA: Addison-Wesley.

Perkins, C. E., & Bhagwat, P. (1994, October). Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. In *Proceedings of the Conference on Communications Architectures, Protocols and Applications* (SIGCOMM '94), Volume 24 Issue 4, (pp. 234-244). London, United Kingdom, 02-09-1994. NY, USA: ACM.

Perkins, C., & Das, S. (1997). *Ad hoc on demand distance vector (aodv) routing ietf*. Internet Draft, draft-ietf-manet-aodv-00. txt.

Perkins, C., Belding-Royer, E., and Das, S. (1997) "Ad hoc On-Demand Distance Vector (AODV) Routing", *RFC 3561,* [online] DOI: 10.17487/RFC3561, Available at: <https://www.rfc-editor.org/info/rfc3561> [Accessed 23-11-2017].

Purtoosi, R., Taheri, H., Mohammadi, A., & Foroozan, F. (2004, September). A light-weight contention-based clustering algorithm for wireless ad hoc networks. In *Computer and Information Technology, 2004. CIT'04. The Fourth International Conference on* (pp. 627-632). Tehran, Iran, 16-09-2004. Wuhan, China: IEEE.

Qu, Y., & Georgakopoulos, S. V. (2012). An average distance based self-relocation and self-healing algorithm for mobile sensor networks. In: *Wireless Sensor Network*, Vol. 4 No. 11, 2012, pp. 257-263. doi: 10.4236/wsn.2012.411037.

QualNet Network Simulator Software. *SCALABLE Develops High-Performance Communications Simulation Software*. [online] Available at: <http://web.scalable-networks.com/qualnet-network-simulator-software > [Accessed 04 October 2017].

Rachedi, A., & Baklouti, H. (2011, June). muDog: Smart monitoring mechanism for wireless sensor networks based on IEEE 802.15. 4 MAC. In*2011 IEEE International Conference on Communications (ICC)* (pp. 1-6). Champs-sur-Marne, France, 09-06-2011. Kyoto, Japan: IEEE.

Rangaswamy, P., & Rangaswamy, R. (2012). Power Aware Routing Protocol (PARP) for Wireless Sensor Networks. *Wireless Sensor Network*, Vol. 4 No. 5, 2012, pp. 133-137. doi: 10.4236/wsn.2012.45019.

Rappaport, T. S. (2002). *Wireless Communications--Principles and Practice*. 2nd ed. Upper Saddle River, N.J: Prentice Hall.

Rashid, R. F., & Robertson, G. G. (1981). Accent: A communication oriented network operating system kernel, *In Proceeding SOSP '81 Proceedings of the eighth ACM symposium on Operating systems principles* (Vol. 15, No. 5), (pp. 64-75). California, USA, 05-12-1981. USA: ACM.

Reina, D.G., Toral, S.L., Barrero, F., Bessis, N. and Asimakopoulou, E., (2013). The role of ad hoc networks in the internet of things: A case scenario for smart environments. In *Internet of Things and Inter-Cooperative Computational Technologies for Collective Intelligence* (pp. 89-113). Sevilla, Spain, 2013. Spain : Springer Berlin Heidelberg.

Rocha, A. R., Pirmez, L., Delicato, F. C., Lemos, É., Santos, I., Gomes, D. G., & de Souza, J. N. (2012). WSNs clustering based on semantic neighborhood relationships. *Computer Networks*, *56*(5), 1627-1645.

Romer, K., & Mattern, F. (2004). The design space of wireless sensor networks. *IEEE wireless communications*, *11*(6), 54-61.

Savvides, A., Han, C. C., & Strivastava, M. B. (2001, July). Dynamic fine-grained localization in ad-hoc networks of sensors. In *Proceedings of the 7th annual* *international conference on Mobile computing and networking* (pp. 166-179). Rome, Italy, 07-2001. NY, USA: ACM.

Selvakennedy, S., & Sinnappan, S. (2007). An Adaptive Data Dissemination Strategy for Wireless Sensor Networks. *International Journal of Distributed Sensor Networks*, *3*(1), 23-40.

Shakkottai, S., Rappaport, T. S., & Karlsson, P. C. (2003). Cross-layer design for wireless networks. *IEEE Communications Magazine*, *41*(10), 74-80.

Sharma, K. and Singla, P., (2016). Threshold based Routing Protocol for WSN with Sleep/Awake Scheduling. *International Journal of Computer Applications,* (0975 – 8887) Volume 133 – No.1, pp. 21-24.

Sharma, S., & Jena, S. K. (2011, February). A survey on secure hierarchical routing protocols in wireless sensor networks. In *Proceedings of the 2011 International Conference on Communication, Computing & Security* (pp. 146-151). Odisha, India, 14-02-2011. NY, USA: ACM.

Shorey, R., Ananda, A., Chan, M. C., & Ooi, W. T. (Eds.). (2006). *Mobile, wireless, and sensor networks: technology, applications, and future directions*. Canada: John Wiley & Sons.

Shuman, D. and Liu, M., (2006). Optimal sleep scheduling for a wireless sensor network node. In *Signals, Systems and Computers, 2006. ACSSC'06. Fortieth Asilomar Conference on* (pp. 1337-1341). Ann Arbor, USA, 01/11/2006. Grove, USA: IEEE.

Sobeih, A., Hou, J.C., Kung, L.C., Li, N., Zhang, H., Chen, W.P., Tyan, H.Y. and Lim, H., (2006). J-Sim: a simulation and emulation environment for wireless sensor networks. *IEEE Wireless Communications,* 13(4), pp.104-119.

Sowmya, K. S., & Cauvery, N. K. (2014). Weighted Clustering Algorithm with Ant Colony Optimization to Provide Better Quality of Service. *International Journal of Computer Science and Information Technologies*, Vol. 5 (6) , 7702-7705.

Stemm, M. (1997). Measuring and reducing energy consumption of network interfaces in hand-held devices. *The Institute of Electronics,Information and Communication Engineers (IEICE) transactions on Communications*, *80*(8), 1125-1131.

Su, W. and Akyildiz, I.F., (2005). Time-diffusion synchronization protocol for wireless sensor networks. *IEEE/ACM Transactions on Networking (TON)*, *13*(2), pp.384-397.

Sucec, J., & Marsic, I. (2004). Hierarchical routing overhead in mobile ad hoc networks. *IEEE Transactions on Mobile Computing*, *3*(1), 46-56.

Tanenbaum, A. S. (2003). *Computer networks*, 4th ed. Jacksonville, USA*: Prentice Hall*.

Tao, L. (2004). Mobile Ad-hoc Network Routing Protocols: Methodologies and Applications. *Ph.D. thesis in Computer Engineering*, Virginia Polytechnic Institute and State University.

Tolba, F. D., Magoni, D., & Lorenz, P. (2007, November). Connectivity, energy and mobility driven clustering algorithm for mobile ad hoc networks. In *IEEE GLOBECOM 2007-IEEE Global Telecommunications Conference* (pp. 2786-2790). Mulhouse, France, 30-11-2007. Washington, USA: IEEE.

Townsend, C. and Arms, S., 2005. Wireless sensor networks. *MicroStrain, Inc*, *20* (9), pp.15-21.

Trigoni, N., Yao, Y., Demers, A., Gehrke, J. and Rajaraman, R., (2004). Wavescheduling: Energy-efficient data dissemination for sensor networks. In *Proceeedings of the 1st international workshop on Data management for sensor networks: in conjunction with VLDB* (pp. 48-57). Toronto, Canada, 30-08-2004. New York, NY, USA: ACM.

Venkataraman, G., Emmanuel, S., & Thambipillai, S. (2007). Size-restricted cluster formation and cluster maintenance technique for mobile ad hoc networks. *International Journal of Network Management*, *17*(2), 171-194. Singapore, 05-03-2007. NY, USA: John Wiley & Sons.

Virrankoski, R., & Savvidees, A. (2005, November). TASC: topology adaptive spatial clustering for sensor networks. In *IEEE International Conference on Mobile Adhoc and Sensor Systems Conference, 2005.* (pp. 10-pp). IEEE.

Wilson, J. S. (2004). *Sensor technology handbook*. Burlington, MA, USA: Elsevier.

Xu, K., & Gerla, M. (2002). A heterogeneous routing protocol based on a new stable clustering scheme. In *MILCOM 2002. Proceedings* (Vol. 2), (pp. 838-843). Los Angeles, CA, USA, 10-10-2002. Anaheim, CA, USA: IEEE.

Xu, Y., Heidemann, J., & Estrin, D. (2001, July). Geography-informed energy conservation for ad hoc routing. In *Proceedings of the 7th annual international conference on Mobile computing and networking* (pp. 70-84). Rome, Italy, 07-2001. NY, USA: ACM.

Yang, W. D., & Zhang, G. Z. (2007, March). A weight-based clustering algorithm for mobile ad hoc network. In *Wireless and Mobile Communications, 2007. ICWMC'07. Third International Conference on* (pp. 3-3). China, 9-03-2007. Guadeloupe, France: IEEE.

Ye, M., Li, C., Chen, G., & Wu, J. (2005). EECS: an energy efficient clustering scheme in wireless sensor networks. In *PCCC 2005. 24th IEEE International Performance, Computing, and Communications Conference, 2005.* (pp. 535-540). China, 09-04-2005. Phoenix, AZ, USA: IEEE.

Yi, S., Heo, J., Cho, Y., & Hong, J. (2007). PEACH: Power-efficient and adaptive clustering hierarchy protocol for wireless sensor networks. *Computer Communications*, *30*(14), 2842-2852.

Yick, J., Mukherjee, B. and Ghosal, D., (2008). Wireless sensor network survey. *Computer networks*, *52*(12), pp.2292-2330.

Younis, O., & Fahmy, S. (2004). HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Transactions on Mobile Computing*, *3*(4), 366-379.

Youssef, A., Younis, M., Youssef, M., & Agrawala, A. (2006, November). Wsn16-5: Distributed formation of overlapping multi-hop clusters in wireless sensor networks. In *IEEE Globecom 2006* (pp. 1-6). USA, 01-12-2006. San Francisco, CA, USA: IEEE.

Yu, Y., Govindan, R., & Estrin, D. (2001). *Geographical and energy aware routing: A recursive data dissemination protocol for wireless sensor networks*. [pdf] Available at: < https://www.csee.umbc.edu/~younis/Sensor_Networks/Reading_Assignments/Old/13. pdfhttp://www.vam.ac.uk/content/articles/i/englishembroidery-introduction/>[Accessed 23-11-2017].

Zainalie, S., & Yaghmaee, M. H. (2008, August). CFL: A clustering algorithm for localization in wireless sensor networks. In *Telecommunications, 2008. IST 2008. International Symposium on* (pp. 435-439). Iran, 28-08-2008. Tehran, Iran: IEEE.

Zeng, X., Bagrodia, R. and Gerla, M., (1998, May). GloMoSim: a library for parallel simulation of large-scale wireless networks*. In Parallel and Distributed Simulation, 1998. PADS 98. Proceedings. Twelfth Workshop on* (pp. 154-161). Canada: IEEE.

Zhang, H., & Arora, A. (2003). GS 3: scalable self-configuration and self-healing in wireless sensor networks. *Computer Networks*, *43*(4), 459-480.

Zhao, F. and Guibas, L.J., (2004). *Wireless sensor networks: an information processing approach.* San Francisco, USA: Morgan Kaufmann Publishers Inc.

Zheng, J. and Jamalipour, A., (2009). *Wireless sensor networks: a networking perspective*. USA: John Wiley & Sons.

Zhong, L. C., Rabaey, J., Guo, C., & Shah, R. (2001). Data link layer design for wireless sensor networks. In *Military Communications Conference*, 2001. Military Communications Conference, 2001. MILCOM 2001. Communications for Network-Centric Operations: Creating the Information Force. Berkeley, USA, 31-10-2001. McLean, USA: IEEE

Zia, T., & Zomaya, A. (2006, October). Security issues in wireless sensor networks. In *Systems and Networks Communications, 2006. ICSNC'06. International Conference on* (pp. 40-40). Australia, 03-11-2006. Tahiti: IEEE.

APPENDIX

Researcher Publications

Al-Baadani, F., & Yousef, S. (2015). A Comparison Study for Different Wireless Sensor Network Protocols. In 10th *International Conference on Global Security, Safety, and Sustainability* (pp. 252-259). London, 21/09/2015. UK: Springer International Publishing.

Al-Baadani, F., Bharti, S., Yousef, S. (2016). Standard deviation based weighted clustering algorithm for wireless sensor networks. *In 8th Computer Science & Electronic Engineering Conference(CEEC).* Essex, UK, 04/09/2016. UK: IEEE.

Al-Baadani, F., Yousef, S., Aldmour, R., Al-Jabouri, L., Tapaswi, S., Patnaik, K. K., & Cole, M. (2016). A Novel Energy Sleep Mode based on Standard Deviation 'ESMSD' Algorithm for Adaptive Clustering in MANETs. In 11th *International Conference on Global Security, Safety, and Sustainability*. London, 26/10/2016. UK: Springer International Publishing.

Al-Baadani, F., Yousef, S., Tapaswi, S., Patnaik, K.K. and Cole, M., 2017. Investigation of the AODV and the SDWCA QoS handling at different utilisation levels in adaptive clustering environments. *International Journal of System Assurance Engineering and Management*, *8*(1), pp.33-36.