2020

Awaking guarantee in mobile wireless networks for mobile nodes based on stochastic mobility mode.

Totona, Bruce Lonyeiye
Faculty of Information Technology
Strathmore University

Awaking Guarantee in Mobile Wireless Networks for Mobile Nodes Based on Stochastic Mobility Model

Bruce Lonyeiye Totona

Thesis Submitted to the Faculty of Information in partial fulfillment of the requirements for the award of Master of Science in Information Technology degree at Strathmore University

Nairobi, Kenya

02 July 2020

Declaration

I Bruce Lonyeiye Totona declares that this thesis on **Awaking Guarantee in Mobile Wireless Networks for Mobile Nodes Based on the Stochastic Mobility Model** is our work and has not been submitted to any other University for the award of a Degree in Information Technology. Also, all the sources that have been used and indicated in this project are acknowledged appropriately and referenced.

Bruce Lonyeiye Totona

Student Number: 78307

Sign: ________________________ Date: ________________________

Approval

I certify that this work is being submitted for examination with the approval

Prof. Ismail Ateya Lukandu D. Sc.,

Associate Professor, Faculty of Information Technology,

Strathmore University.

Sign: ________________________ Date: ________________________

Abstract

The mobile wireless network is one of the areas in mobile computing getting more attention due to a plethora of innovations around the subject. The innovations and applications range from wireless sensor networks (Low-Rate Wireless Personal Area Network (LR-WPANs)), embedded systems and robotics. However, the technology provides some challenges in terms of mobile nodes awakening (active connection and synchronization) and services guarantee because the networks are highly dynamic. This research studied how to improve service provisions in mobile wireless networks and particularly in the wireless sensor networks using ZigBee modules by integrating synchronization and the routing procedure in the network. Ad-hoc On-demand Distance Vector (AODV) which is a common routing algorithm in wireless mobile networks was combined with the synchronization algorithm- non-beacon-enabled CSMA-based IEEE 802.15.4 MAC to overcome the challenge of hidden mobile nodes which are most often the new nodes joining the network. The beacon interval (Bi), the superframe duration (Sd), and the beacon time offset (Bto) were the key integration parameters and comes after active scanning. Random Waypoint (RWP) mobility model on Matlab was used to evaluate the mobility of the nodes considering their speeds, direction, and position from the coordinator. Elliptic Curve Digital Signature Algorithm (ECDSA) and the Advanced Encryption Standard (AES) algorithms were used to secure the modules and the channels.

Keywords: *Algorithm, dynamic network, low-level services, mobility, robustness, synchronization, ZigBee, Beacons, Routing, Connectedness, Awaking*

Definition of Terms

Algorithm - Set of rules used for problem-solving operations by a computer (Intellias, 2018).

Dynamic network - It is a network that changes with time and space, for example, the transportation networks and the mobile wireless devices (Függer, Nowak, & Charron-Bost, 2015).

Low-level services - Initial services in computer communication such as synchronization and communication (Függer, Nowak, & Charron-Bost, 2015).

Mobility - Versatility of new individual nodes joining the system (Shah, Ahmed, imran, & Zeadally, 2018).

Robustness - Condition of a system to show high quality, strength and proper health (Mazda & David, 2016).

Synchronization - Awaking of wireless ad-hoc networks to coordinate events in unison (Danda, Bhed, & Gongjun, 2016).

Table of Contents

Table of Figures

List of Tables

List of Equations

Chapter 1: Introduction

1.1 Background

The fifth-generation of mobile networks commonly referred to as 5G is ongoing, and it is one of the many developing technologies that aim to augment existing applications as well as introducing new applications that require highly reliable and low latency services such as communications and coordination. 5G is a radio technology that is predicted to provide for a greater density of network devices and wider compatibility with mobile wireless networks that supports low power low data rate devices. Some of these mobile wireless networks technologies have a greater influence on the design of 5G. ZigBee, near field communications (NFC) wireless fidelity (Wi-Fi) and Z-Wave, are just some of these technologies which provide a wide set of communication services to several applications and innovations (Shah, Ahmed, Imran, & Zeadally, 2018). The nature of mobile wireless networks is influence by the dynamicity of the mobile nodes. As the nodes move, synchronization and coordination are needed to enhance the provision of the required services. For example, the concept of Intelligent Connected Vehicles (ICVs) and Vehicular ad hoc networks (VANETs) (Yang, Jiang, & Zhao, 2018). Among the existing radio technologies, ZigBee based wireless mobile networks have been applied widely because of its ability to provide beacon-enabled innovations and its stability on applicability to the moving devices in the dynamic network. Mobile nodes in a dynamic network require an active connection, synchronization, and communication if a services guarantee is to be achieved (Threlfall, 2019).

Moreover, highly dynamic networks such as the communication networks typically comprise of nodes with some limitations such as the power limitations thus necessitate nodes to go into sleep mode when there is no communication hence saving energy. Based on the applications of the technology, some nodes are highly dynamic hence they require a higher level of synchronization. This would mean some nodes would consume more power based on the innovation and also, the nodes might be required to go into a sleep mode (inactive) for a shorter duration. Therefore, a low level of communication and synchronization is an issue in these types of networks (Függer, Nowak, & Charron-Bost, 2015). According to Pype, Daalderop, Schulz-Kamm, Walters, and von-Grafenstein (2017), some applications of the mobile wireless networks such as ICVs require efficient communication and synchronization which in return improves network signaling of the mobile nodes. Any compromise on the signaling of the mobile nodes will comprise the functionality of the entire application thus zero service provision. The research was, therefore, focused on applying strong routing, synchronization, and security algorithms in mobile wireless networks to enhance active connectedness and synchronization of all the nodes. Ad hoc on-demand distance routing vector (AODV) algorithm was combined with the IEEE 802.15.4 MAC and PHY protocol to improve active connectedness and synchronization while Elliptic Curve Digital Signature Algorithm (ECDSA) and the Advanced Encryption Standard (AES) algorithms were used to enhance the security of the nodes and the network. Figure 1.1 illustrates the concept of the mobile wireless network with a few random mobile nodes based on research by Függer et al., (2015).

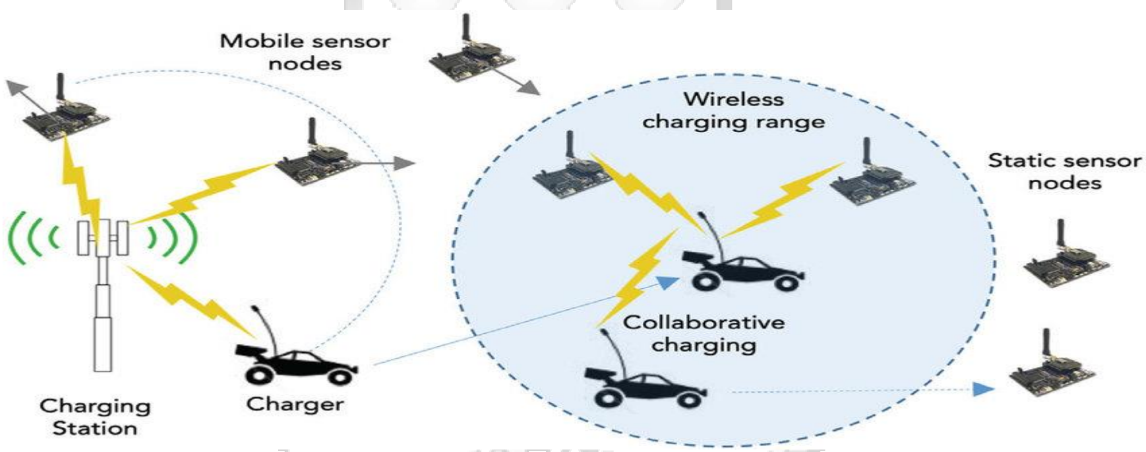

Figure1.1: An Example of a Mobile Wireless Network

1.2 Problem Statement

Giarré, Pesenti, and Tinnirello, (2010), present a decentralized synchronization in mobile wireless networks based on wireless sensor networks which aim to improve synchronization of mobile nodes. However, according to their research, another challenge that arises is the hidden nodes' complexities. which eventually takes more time for the mobile sensor network to update its routing table and topology. The hidden nodes would not be synchronized to other nodes immediately because the nodes would be idle (inactive) when existing nodes are transmitting to the coordinator. The approach adds some complexities to the network. A better possibility is to combine the routing data on joining the network with the synchronization procedure (Giarré et al., 2010).

Communication complexity and overhead of the clock synchronization algorithm itself is an issue in the dynamic systems. Wireless sensor networks for example is characterized by low power and low data rate mobile nodes which requires high efficiency and effectiveness. As a result, communication (routing) can not be done simultaneously with synchronization thus introducing some complexities in handling new unsecured mobile nodes and the existing mobile nodes (Függer et al., 2015). Current models present synchronization challenges in regards to the dynamic network of moving nodes. From an analysis of synchronization components interconnection of highly mobile nodes, signaling among the nodes is important but when it takes more time for nodes to converge, coordination among the nodes is compromised, therefore, vital service provisions would not be achieved. Active connections, high accuracy, and effective communication are necessary for such applications (Alex, 2016). The research, therefore, looked into improving active connectedness and synchronization by integrating routing and synchronization in the mobile wireless networks. It also considered the range and the strength of the communications among the moving nodes for active connectedness.

1.3 Aim

The research aimed to evaluate and implement routing, encryption and synchronization algorithms used in mobile wireless networks to enhance service guarantee by improving active connectedness and synchronization without compromising routing, mobility, and security of the highly mobile nodes Moreover, the study also aimed to solve the complexity of the hidden new mobile nodes joining the mobile wireless network.

1.4 Objectives

(i) To review the operation of a wireless mobile network and its applications.
(ii) To design a routing and synchronization procedure in mobile networks.
(iii) To analyze the routing and synchronization procedure based on mobility models.
(iv) To implement a prototype for routing and synchronization procedures in mobile networks.
(v) To test the prototype developed.

1.5 Questions

(i) How do a wireless mobile network and its variations work in mobility applications?

(ii) How can a routing and a synchronization procedure for mobile nodes be designed?

(iii) How do mobility models support routing and synchronization?

(iv) How can routing and synchronization be implemented in mobile wireless networks?

(v) How valid is the prototype developed for the mobile nodes?

1.6 Justification

This research came in handy to add knowledge on the awaking (active connections and synchronization) and mobility in the wireless networks. It also contributes knowledge to the applications of mobile wireless networks such as the synchronization of autonomous vehicles for the future transport industry, robotics, military, and the use of drones in agriculture.

1.7 Scope and Limitation

The research covered the algorithms which are based on the stochastic mobility model to explain mobility and the ad hoc on-demand distance routing vector (AODV) algorithm was implemented as the routing protocol while the IEEE 802.15.4 MAC and PHY standard were implemented to illustrate synchronization in mobile wireless networks. The Random Walk and the Random Waypoint models were implemented to illustrate the concept of mobility in the network, for example, how the autonomous vehicles and drones systems would behave. A prototype to test the effectiveness of the routing and synchronization was developed based on the mention algorithms. This research did not involve building a real prototype of the autonomous cars but rather the mobility aspects of the nodes were illustrated with an animation on a MatLab tool. The procedure to achieve reliable connections, coordination, and synchronization were illustrated by testing the algorithms on a ZigBee module, Arduino module, and MatLab.

Chapter 2: Literature Review

2.1 Introduction

Mobile wireless networks services and applications are dependent upon several algorithms for various services such as navigation, communication, synchronization, mobility, connectedness, and many other services. Some of these algorithms have a similar feature while others are different from each other and maybe of different models but most of it has to work together in integration to provide the complete functionality of the systems. Some of the algorithms are even more robust, stronger efficient than others, and thus are the best suitable for implementing services with vital requirements in the nodes. For example, synchronization and signaling of mobile nodes such as in vehicular ad hoc networks (VANETs) and the network of autonomous vehicles (Ibañez-Guzmán, Laugier, Yoder, & Thrun, 2015). Synchronization in a mobile wireless network for instance ZigBee personal area network requires more advanced implementation or algorithm than what is already in use since new mobile nodes provide a challenge in connecting to the existing nodes, therefore, network convergence takes more time and resources. ZigBee network treats routing, mobility, and active connectedness (synchronization/awaking) as different services. However, for the innovations and applications, these services need to be integrated to enhance effective and efficient mobile services provision (Fugger et al, 2015).

VANETs and the network of autonomous cars are just an example of mobile wireless networks that require high mobile network services provisions. Following a mobile wireless network analysis of testing synchronization services among nodes components in an Uber autonomous car which got into an accident in March 2018, shows that signaling in its network was the primary problem as some sensors went into a sleep mode for a longer duration (Alex, 2016). Energy utilization and efficiency in the mobile nodes are very important but some services may require mobile nodes going into sleep mode (inactive mode) briefly or not all as coordination and communication are key and vital vehicles (Fugger et al, 2015). Apart from VANETs and autonomous cars, there is a large area of applications of mobile wireless networks innovations such as in robotics, agriculture, and military

2.2 Mobile Wireless Networks

Wireless communication systems are one of the rapidly advancing communication technology characterized by the deployment of several novelty ideas and applications in different subjects including automotive and sensor communications (Danda, Bhed, & Gongjun, 2016). These communication systems are widely categorized into network structures like infrastructure-based wireless and infrastructure-less wireless networks. The infrastructure-based wireless network requires a centralized point of communication to other nodes in the network while the infrastructure-less wireless network does not require a central point of communication to other devices. This network type can also be called a wireless ad hoc network and also peer-to-peer network as the devices are very mobile. The topology of the peer-to-peer or the wireless ad hoc network is dynamic due to the nodes continuously changing their position. This type of network is appropriate to the moving vehicles or drones due to the greater mobility of the nodes and the simplicity of integrating with other different network components and also types of networks thus can also scale very well (Danda et al, 2016). Figure 2.1 illustrates the situation of a mobile wireless network where the wireless connected nodes change in the network.

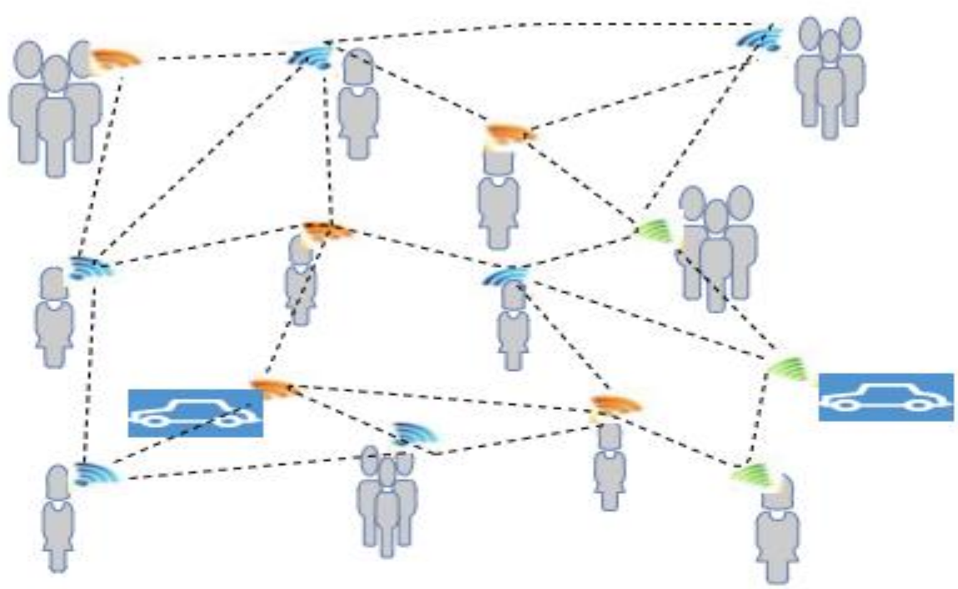

Figure 2.1: An Illustration of a Mobile Wireless Network (Joarder, 2011)

2.2.1 Mobility in Mobile Wireless Networks

Mobility in ad hoc networks or, mobility in mobile networks, originate from the idea that the nodes in communication within a network change their position frequency and the nodes themselves change more often thus assuming a distributed environment which is highly dynamic and has a wide range of applications (Attia, Rizk, & Arafat, 2015). The unique ability of mobile wireless networks is the provision of services and applications to other devices and users in dynamic network connectivity without physical cabling. Therefore, mobile wireless technology is expected to provide connectivity and services to different types of mobility characteristics (Griffin, 2018). Mobile wireless sensor network (MWSN) is an example of a mobile wireless network and it plays an important role in current practical applications such as in transport, smart spaces, meteorology, health care, and undersea navigation. MWSNs are much versatile as the nodes deployed in any terrain or scenario cope with rapid topology changes in the network (Ramasamy, 2017).

2.2.2 Awaking in Mobile Wireless Networks

Wireless ad hoc networks are systems where several nodes communicate among themselves thus forming an independent network. The basis of the network is the peer to peer communication od the nodes as each node posses an antenna for receiving a communication signal. Therefore, these networks allow for information exchange between the nodes and also providing them with the capability to communicate with remote sites (Attia, Rizk, & Arafat, 2015). Sensor networks being a type of these networks, it is comprised of mobile nodes each with a sensor of application-specific, a processor, and a transceiver that is wireless. Potential application areas of these networks (sensor networks) would be traffic management where the information among vehicles is shared. The nodes, in this case, assumed a random placement resulting from sensors being distributed over a certain region covered by moving vehicles. With such connections, the network characteristics will be changing due to the constantly changing network topology (Santi & Blough, 2002). Energy-efficient consumption is vital in the sensor network. However, a major concern as the mobile wireless networks advanced is the conditions necessary for connectivity (initial connections and ensuring nodes remain connected) as nodes migrate (Attia et al, 2015).

Attia et al. (2015) partially address the issue in probabilistic terms of stochastic properties through evaluation of the probabilities of events relating to the connectedness in the network. The approach examines the nodes required in the network and the transmission ranges necessary for establishing a wireless ad hoc network. The work, however, does not take into account the several protocols and methods used for the provisioning of the services in these networks. The technique used by the researcher in this work aimed at solving these issues through simulation. Figure 2.2 illustrates an idle state (disconnected time) between nodes resulting in a failure event. Autonomous vehicles or drones will require little or no of this failure event thus the best algorithm is key to such nodes.

Figure 2.2: Connectedness and Awaking Among Mobile Nodes (Mazda & David, 2016)

2.3 Stochastic Mobility Models

The stochastic model controls the mobility of nodes in a mobile ad hoc network thus it has a great effect on the coverage and throughput of the network. There are numerous mobility models and research of these models by Bandyopadhyay et al. (2007), categorized the existing mobility models into four classes: Random models, models capturing the temporal correlation in a node's movement, models capturing spatial correlation between nodes close to each other and models with geographic restrictions. Random Walk and the Random Waypoint mobility models are examples of random models. The two models do not consider geographical restrictions, temporal correlation, or spatial correlation. Gauss-Markov mobility models and Smooth Random models are also random models that consider temporal dependence but no considerations on spatial dependence or geographical restrictions (Bandyopadhyay et al, 2007).

On the contrary, group mobility models capture spatial dependence but do not account for temporal dependence or geographical restrictions. A significant effort has been made to study how realistic these models are and how they affect the performance of different MANET protocols and it was observed that the protocol performance is highly influenced by the mobility models. The authors studied the effect of mobility models on the distribution of link duration in a network which in turn affects the network protocol performance. Recent research by Pfeuffer (2018), found that the delay-capacity trade-offs are radically different for different mobility models like Brownian Motion and a Random Waypoint. The same research by Pfeuffer (2018), also paid attention to the study of stochastic properties of mobility models and propose a combination of two models, for instance, the Random Trip model which contains the Random Waypoint and Random Walk models though it finds the existence of stationary conditions. This work provided an idea to the researcher on combining two or more algorithms to enhanced active connections of the mobile nodes in the dynamic network of autonomous vehicles.

Random synthetic stochastic Vector Mobility Model (VMM) and Yates algorithms have been the most tested algorithms to simulate car movements in a very dynamic network - though the tests have been independent with an interest in testing the general stochastic characteristics of mobile vehicles in a changing network (Pfeuffer & Li, 2018). Moreover, other random synthetic stochastic mobility algorithms like Random Walk, Gauss-Markov mobility, Fluid Flow, and Random Waypoint have been tested independently and compared to the stochastic characteristics of wireless mobile networks and each of these algorithms have shown a varying level of connectivity and synchronization of the moving nodes (Chung & Go, 2012).

Several researchers have had to rely much on analysis and simulations to prove this. The accuracy of the predictions made about network performance in the real-life scenarios has depended on how well the mobility model used in the simulations captured the real-life behavior of nodes which have then driven to the implementation of the most and best realistic mobility models (Bandyopadhyay, Coyle, & Falck, 2007). Figure 2.3 displays sample paths of three mobility models — random walk, asymmetric CRW (ACRW), and symmetric CRW — on a 200×200 grid for 100 units of time. The random walk path is shown in blue, the asymmetric CRW in cyan, and the symmetric CRW in red.

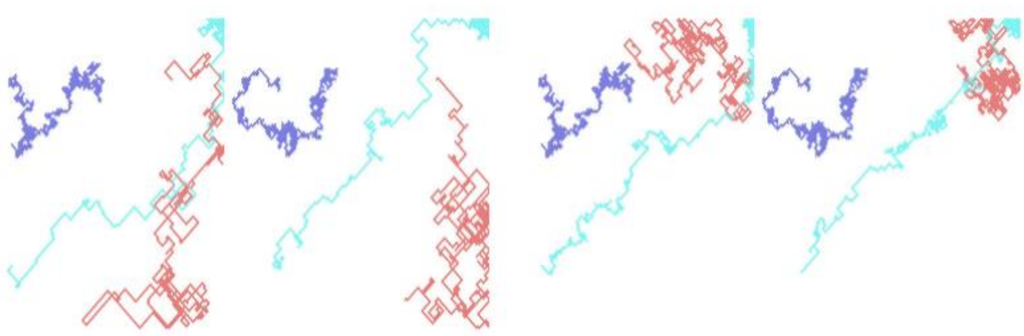

Figure 2.3: The Random Walk, Asymmetric CRW (ACRW) and Symmetric CRW (Bandyopadhyay et al, 2007)

2.3.1 Mobility Aspect in Stochastic Mobility Model

An essential characteristic of vehicular or automotive mobility is that it tends to move in a certain direction towards a destination while displaying a certain degree of randomness in speed and angle as it moves (Chung & Go, 2012). Different mobility models have been used to model these features. Bandyopadhyay et al. (2007), compared mobility models applicable to highly mobile nodes such as intervehicle and drones communications like the Random Walk, Random Waypoint and Manhattan Mobility models in terms of the number of condition states over a certain period, the time is taken for each condition state, and the effect of these mobile nodes on the time taken on each state. The results were that the overall time and the taken for each condition state was very important in building profile for the acceptable events detections system. The research focused on evaluating detection rates in surveillance systems and applications using a wireless-sensor-equipped vehicle in a city. The results were further used to provide guidelines on the density distribution of the mobile devices within a specific area coverage for effective detection of events by the surveillance systems. Figure 2.4 depicts the application of a Random Waypoint for direction mobility in an autonomous vehicle

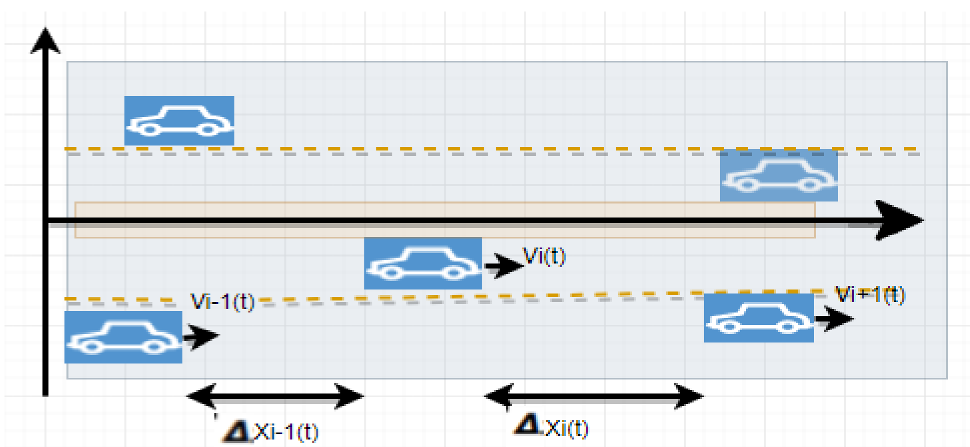

Figure 2.4: Random Waypoint Algorithm Application on Road Scenario (Xiong & Qing-Quan, 2018)

Pfeuffer and Li (2018), proposed and investigated Correlated Random Walk (CRW) based mobility model that is designed to characterize the movements of nodes, such as wireless sensors-equipped vehicles on the streets of a city. In this approach, the streets are modeled as a two-dimensional square grid and each point on the grid corresponds to an intersection of streets.

Random synthetic stochastic vector mobility model (VMM) has been used to simulate vehicle traffic movement for mobile and vehicle ad hoc networks. The objective of the VMM is to easily regenerate random mobility patterns that accurately match the stochastic characteristics of vehicles, for instance, cars and buses better than other random synthetic stochastic mobility simulators, which include the Random Walk, Gauss-Markov mobility, Fluid Flow, and Random Waypoint. Chung and Go (2012), also proposed the destination targeted VMM (DTVMM) model in which nodes can be configured to randomly move towards the exact location or the neighborhood of the targeted destination nodes or point based on the DTVMM stochastic parameters. When the stochastic parameters of actual traffic traces are used, the simulation results show that the DTVMM movement patterns are very similar to actual traces and have the same stochastic characteristics (Bandyopadhyay et al, 2007).

Gauss and semi-Markov (SGM) mobility models are characterized by the smooth flow of mobile nodes similar to the law of motion. This characteristic removes the sharp turns, unsteady speed change, and unpredicted stops that other mobility models posses. The SGM model has five phases, namely: acceleration phase, stable phase, turn phase, deceleration phase, and pause phase which includes the movement states in realistic conditions. This, therefore, shows that the model can be easily and flexibly applied for simulating node mobility in mobile wireless networks (Hengyang, Zheng, & Zhuo, 2015).

2.4 Routing and Synchronization Algorithms in Mobile Wireless Networks

2.4.1 Routing Algorithms in Mobile Wireless Networks

Routing is the procedure of choosing a path in a network along which to transmit a packet. When a network is mobile, a transmitter must first find a path for each packet it wishes to send (Emmanuel & Sanders, 2013). Several routing protocols such as Destination sequenced distance vector(DSDV), wireless routing protocol(WRP), cluster-head gateway switch routing(CGSR), Ad hoc on-demand distance vector routing (AODV), dynamic source routing(DSR), zone routing protocol(ZRP) and source tree adaptive routing (STAR) have been implemented for the mobile networks routing. The terms used in routing include but not limited to: a hop which is a connection in a path, the hop length which is the distance of the path, and the sequence number which is the measure of how the information is new. The higher the sequence number the newer the corresponding information. Furthermore, the network devices have some communication techniques which include; Unicast, where a packet is sent from a sender to one receiver, group cast where packets are sent from one or more devices to a specific group of devices in the network, multicast where packets are sent from one or more devices to a set of other and the broadcast where the packets are sent from one sender to all its neighbors (Emmanuel & Sanders, 2013).

A route request is a message sent by a routing device finding a list of the best routes in a network. The route reply, on the other hand, is the message sent by the device in response to the request. The request is initiated by a transmitting device. With several route requests and responses, a routing device keeps a routing table which are the records stored and maintained and contains routing information of the other devices in a similar network. Ad hoc On-Demand Distance Vector Routing (AODV) and Dynamic Source Routing (DSR) are the most scalable and secure routing algorithms used to send packets in a mobile network (Emmanuel & Sanders, 2013).

2.4.2 Routing with AODV in Mobile Wireless Networks

Ad hoc on-demand distance vector routing (AODV) is a routing protocol mostly used in MANETs (mobile ad hoc networks) and other wireless ad hoc networks. It is a reactive routing protocol thus establishes a link to the destination when needed only. AODV routing is built over the Destination sequenced distance vector (DSDV) algorithm and is, therefore, a significant improvement over DSDV (Emmanuel & Sanders, 2013). The devices that are on a specific path maintain routing information and take part in exchanging the routing tables. When a transmitter wants to send a packet to a receiver and does not have a correct route to it, the transmitter initiates a route discovery process. Transmitter sends a route request message to all its neighbors, the neighbors forward the request to all their neighbors, and so on until either the receiver or an intermediate device with the current route to the receiver is reached. To optimize the route performance, intermediate devices keep a record of the addresses (Java T Point, 2018). Figure 2.5 illustrates the routing process in a ZigBee network using Ad hoc on-demand distance vector routing (AODV).

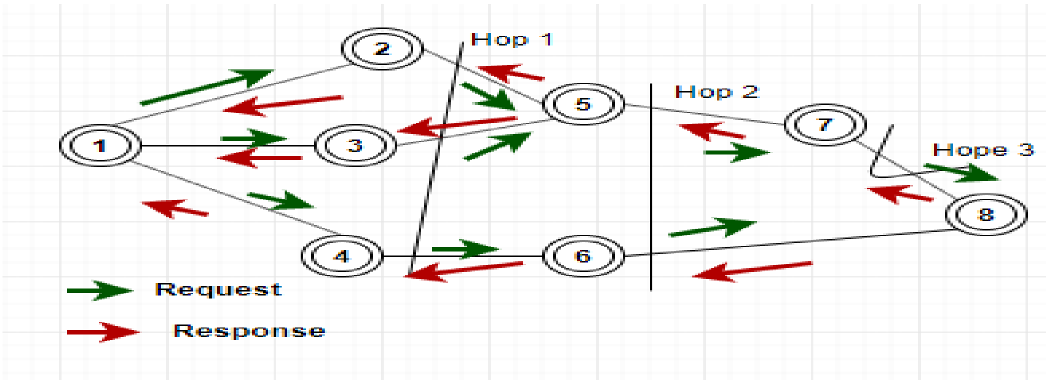

Figure 2.5: Ad hoc On-Demand Distance Vector Routing Algorithm

2.4.3 Synchronization Algorithms in Mobile Wireless Networks

Time synchronization is key in Mobile Ad-hoc Networks (MANETs) due to the constant movement of the nodes in MANET. Time synchronization in such systems helps in ensuring that routing information is accurate and that the communication and other related processes are secure (Swathi & Saravanan, 2014). The ability to maintain a standard time among the several devices in a distributed and dynamic environment is necessary and important concerning communication, coordination, and service provisioning. Synchronization orders events related to the devices such as the readings, configurations, and other activities that take place inside the devices. The intervals of these events are measured in real-time and some of the application areas range from sensor networks to the dynamic network of drones and autonomous vehicles in which the network topology changes unexpectedly at every transmission interval (Fugger et al, 2015).

Clock synchronization algorithms can be categorically be defined as being centralized or decentralized. Centralized synchronization is not suitable for highly dynamic networks such as that of moving drones and cars because the synchronization of events is not flexible with the network changes. Decentralized clock synchronization algorithms overcome this issue because it is well-suited for the highly dynamic networks where the events of the mobile nodes can be synchronized. The synchronization approach, however, is characterized by low communication and synchronization complexities (Fugger et al, 2015). Diffusive algorithms can then be used to solve clock synchronization complexity for networks that have directed spanning trees since it allows for changes in clocking events. More research is necessary to illustrate the efficiency and effectiveness of the approach in real-world dynamic networks (Fugger et al, 2015). Figure 2.6

illustrates diffusive clock synchronization in a highly dynamic network with the very least delay requirements as in communications among the mobile nodes.

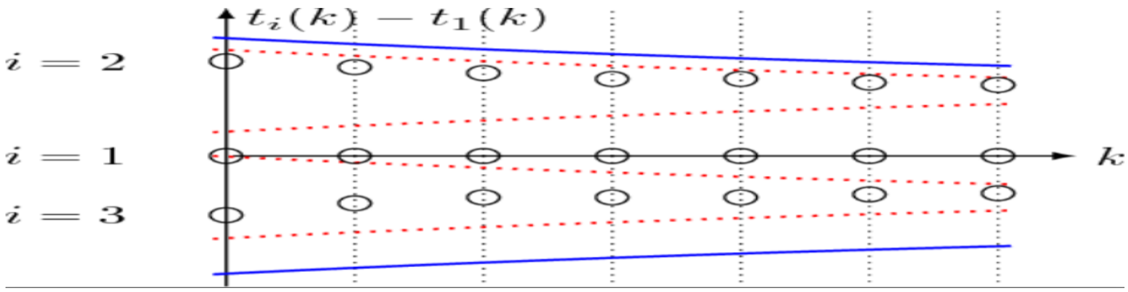

Figure 2.6: Diffusive Clock Synchronisation in Highly Dynamic Networks (Fugger et al, 2015)

2.4.4 Routing Algorithms Based on Mobility Models

Vehicular Ad-Hoc Network (VANET) must be capable of communicating in any environment regardless of the traffic densities and vehicle locations (Morales, 2011). The application of the clustering algorithm in VANET is effective because the algorithm can make the network more robust and scalable. A great challenge though is obtaining stable clusters due to the high mobility of nodes. Several packets are also dropped due to the constant route repairs or failure thus increasing significantly the overhead. This also leads to low delivery ratios and high retransmission delays (Marzak, Toumi, Talea, & Benlahmar, 2015). Regarding this work, Marzak et al. (2015) proposed a model- YATES algorithm that calculates the value of stable and the nodes are designed to overcome the stability of the cluster.

Table 2.1: Cluster Head Selection Algorithm Simulation

Parameter	**Values**
No. of Nodes	500
Area size	3000 X 4000 m
Routing protocol	AODV
Beacon packet size and packet size	512 bits, 2500 bits
Simulation time	500 secs
Traffic Source	CBR
Transmission range	250 m

Metrics	Cluster head duration, cluster member duration and cluster lifetime
Destination node	10

Table 2.1 shows some inputs used for simulating the cluster head algorithm. To evaluate the algorithm stability and the general performance, the following metrics were used:

(i) Average cluster head duration: Longer cluster head duration was important for reliability and security on communication.

(ii) An average number of cluster member duration: This metric indicates the general stability of the previous clustering.

(iii) Cluster Life Time: This metric measures the stability of a cluster

(iv) Reliability: Reliability is concerned with the ability of a network to carry out a desired operation such as synchronization and communication. If the system has higher reliability, then the network has more security.

Ad hoc on-demand distance vector routing (AODV) and Dynamic source routing (DRP) are the most used and reliable routing algorithms used in mobile wireless networks. AODV is a reactive routing protocol in mobile ad hoc networks meaning that a route to the receiver is set only when needed. AODV routing protocol is built over the DSDV algorithm thus it has a significant improvement over the DSDV. DRP is an on-demand routing protocol that is based on transmitter routing and is very similar to AODV in that it forms a route on-demand when transmitting route requests only. Moreover, the algorithm uses source routing rather than the routing table at each intermediate device. The algorithm also operates in route discovery and route maintenance phases as when a device initiates a request to send data, the route cache must be contacted for the routes updates to the receiver. If a working route to the receiver exists, the device sends data using it (Java T Point, 2018).

2.4.5 Synchronization Algorithms Based on Mobility Models

With the characteristics of high transmission delay and mobility, distributed time synchronization of sensor networks can be very challenging. The existing works cannot get high accuracy, because most of them ignore the long transmission latency or dynamic changing transmission delay caused by the mobility of the devices (Wen & Lin, 2016). A mobility model is always used considering the effect of mobility and long transmission latency on time synchronization. Based on the parameters of mobility aspects of the nodes, equations would conform to the algorithms used. For example, a time synchronization algorithm (MM-sync) experiment by (Wen & Lin, 2016), results show that MM-sync can reduce the consumption of energy, and can get higher accuracy than state-of-art solutions in high-density underwater sensor networks with rapid movement.

2.5 Conceptual Framework

The framework illustrated by figure 2.7 shows the connection between the research objectives and the literature reviewed and these, therefore, determine research methodology to be used. The framework illustrated in Figure 2.7 indicates the process to be followed to achieved awaking in mobile nodes through simulation.

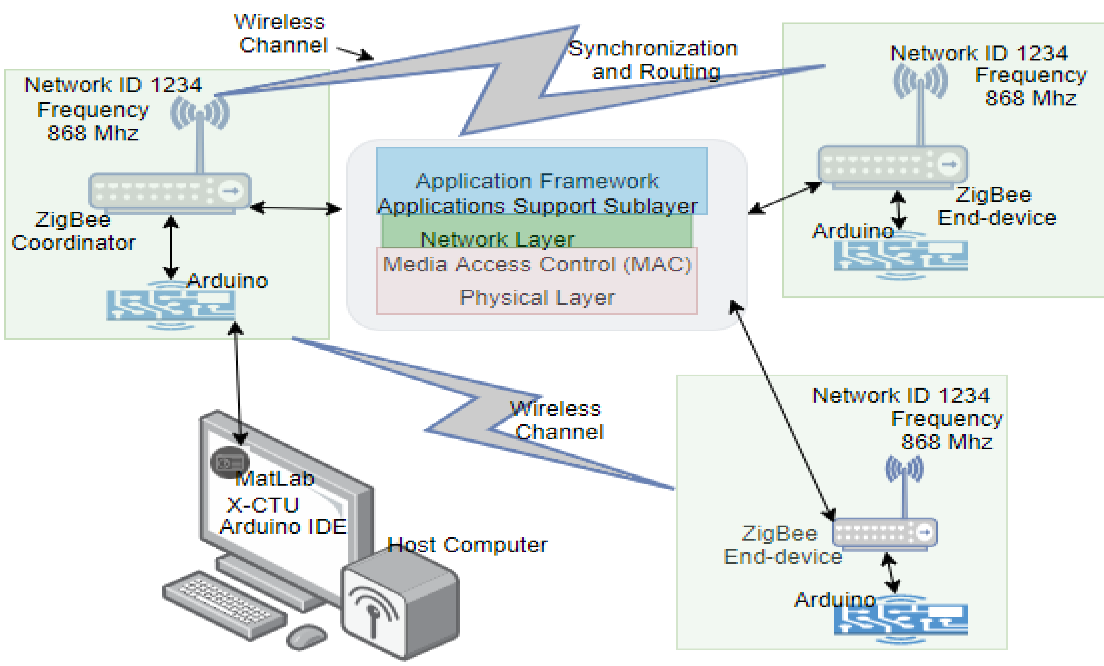

Figure 2.7: Conceptual Framework

2.6 Applications of Mobile Wireless Networks

2.6.1 Intervehicle Communications

Vehicular Ad Hoc Networks (VANET) comprise one of the application areas of mobile ad hoc networks (MANETs) that standout and with unsurprising mobility limitations, able to sort out without predefined infrastructure (Singh & Dadhich, 2015). These networks enable vehicles to communicate with one another or with the roadside framework and will eventually have more secure and increasingly effective streets through the trading of convenient data to the navigation and other control vehicle systems. Discoveries: The routing of data in VANETs is a noteworthy test since they are described by high mobility bringing about a profoundly unique and highly dynamic topology (Marzak, Toumi, El-Guemmat, Benlahmar, & Talea, 2016). This, therefore, prompt me to evaluate and test some of the algorithms inbuilt to routing devices in regards to information exchange and service provisioning using MatLab, Arduino, and ZigBee module. Figure 2.8 shows how autonomous cars form a dynamic network and communicate wirelessly.

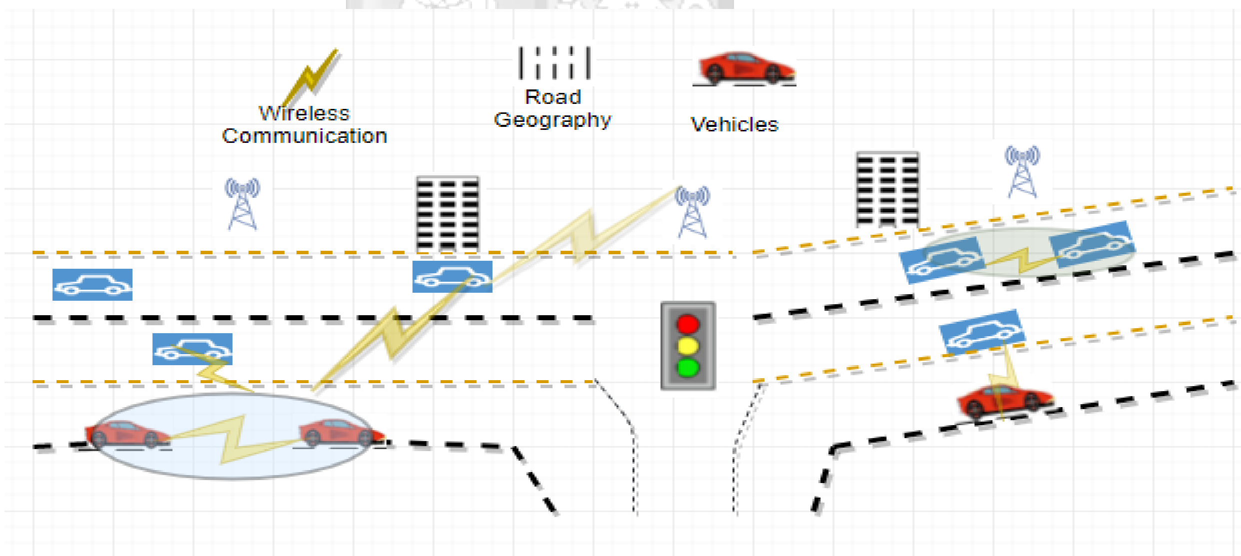

Figure 2.8: Mobility and Connectedness in VANETs   (Cheng et al, 2018)

2.6.2 Drones Driven Operations

Drones are increasingly used in a wide range of application areas for several services provisions thus a great improvement to several industries. However, technology needs to be working effectively and efficiently to ascertain safety in operations and the monitoring of this growing fleet of drones (Lin, Wiren, Euler, Sadam, Maattanen, Muruganathan, & Yajnanarayana, 2019). According to Lin et al. (2019), current mobile networks have the capacity and ability to serve drones operating in low-altitudes. Drones in this airspace are experience low-latency and high-data-rate in terms of communications since they are connected to the mobile network. The drones-connected network is secure in this case since it purely depends on the Long Term Evolution (LTE) network and not connection amongst themselves (Lin et al, 2019). Therefore, the greatest challenges in this scenario are related to interference and mobility.

2.6.3 Wi-Fi Setup as a Mobile Wireless Network

Wireless Fidelity (Wi-Fi) is a general word referring to the IEEE 802.11 standards for a wireless communication system. It is a system that connects several devices forming a single network and the devices are connected in a wireless mode using radio technology. Wi-Fi transmissions are of higher data rates compared to the many radio technologies (Kursus, 2016). Wi-Fi topologies can be categorized into peer-to-peer topology (Ad-hoc Mode) and access point (AP) based topology (Infrastructure Mode). Peer-to-peer topology does not require a central point of communication which is an AP, therefore, the devices communicate with each other directly as long as they are withing the coverages of the signal. This is, therefore, the quickest and easiest way of setting up a wireless network. The infrastructure network requires a central point of communication which is an access point (AP) since all the communications from the devices must be handled by the AP. For instance, if a mobile phone or a laptop or wants to send information to another device, it needs to send the information to AP first, then AP forward it to the receiving device (Kursus, 2016). An infrastructure network can also be called a basic service set (BSS). Figure 2.9 illustrates a Wi-Fi peer to peer network while Figure 2.10 illustrates infrastructure network topologies.

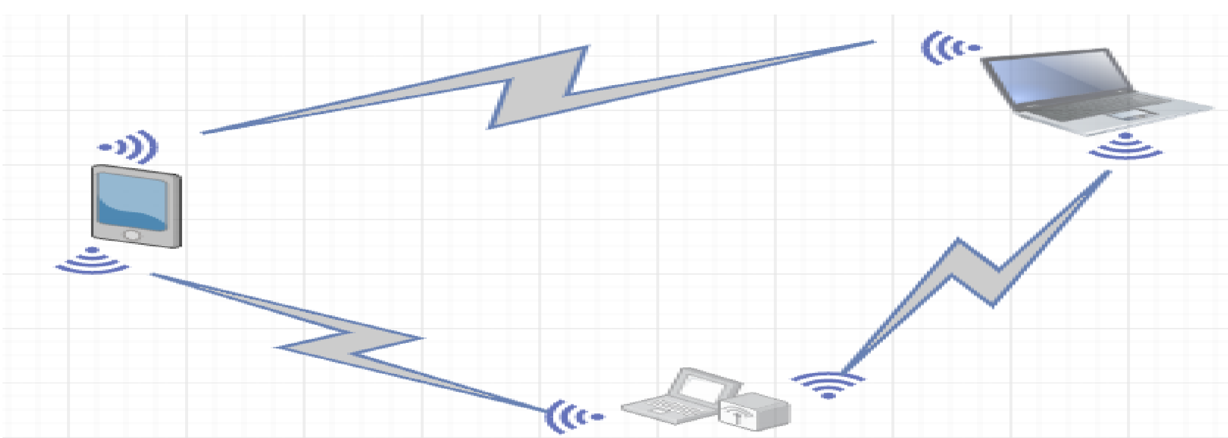

Figure 2.9: Peer-to-Peer Wi-Fi Topology

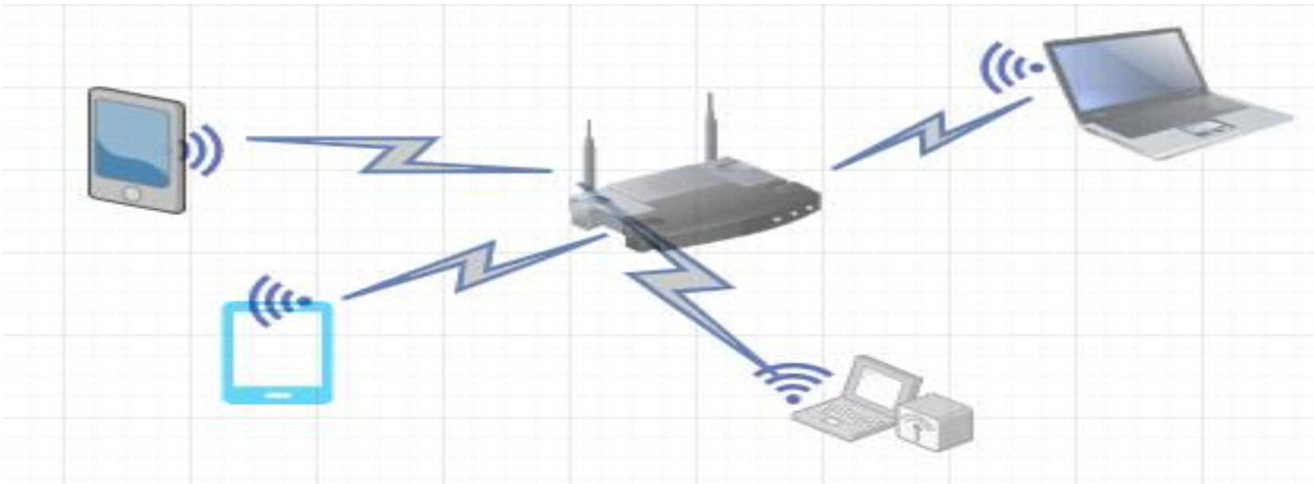

Figure 2.10: Infrastructure Wi-Fi Network

Extended Service Set (ESS) is a distribution system (DS) where two or more BSSs work together to form seamless connections from mobile devices (Rabbit Technologies, 2016). Figure 2.11 shows a representation of an ESS comprised of BSS 1, 2 and 3. An ESS in figure 2.11 is a Wi-Fi network of size and complexity. The distribution system is not part of the ESS. The primary role of the distribution system is to allow mobility in the Wi-Fi network through a means of tracking down the physical locations of the devices. This ensures that mobile devices transmit data to the corresponding access point and vice versa. Mobile devices can change the access point as they move within the coverage area of ESS and still keep constant connections (Rabbit

Technologies, 2016). The service set identifier (SSID) or network name must always be similar for all the access points in the same ESS.

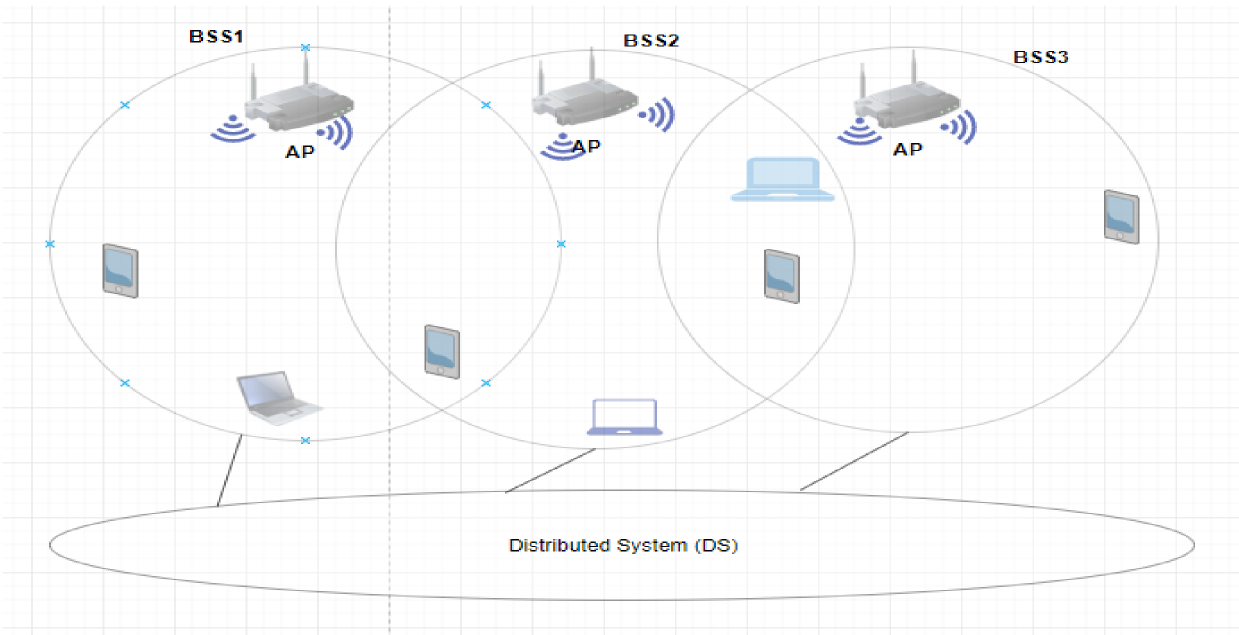

Figure 2.11: Extended Service System

2.6.4 ZigBee Setup as a Mobile Wireless Network

Communication among ZigBee devices results in a mesh topology network as several devices connect and internetworked forming multiple redundant connections linking the devices. The interconnections in this network are frequently updated and automatically through an advanced built-in mesh routing table (ZigBee Wireless Mesh Networking, 2019). ZigBee network is highly distributed due to the ability of the devices to carry out self-discovery in the network. The ZigBee network also has a self-building and self-healing mechanism where when some devices leave the network, the mechanism enables the existing devices to reconfigure and update the paths automatically to match the new network design. Mesh networks and ad-hoc routing greatly support the topology in the case a single node fails and this permits devices to drop out of the network without any disruption to internal routing (ZigBee Wireless Mesh Networking, 2019).

ZigBee has advanced network security methods that be centralized or decentralized. The centralized method utilizes a trust center which is a coordinator that controls and manages the assignment of network addresses and links authentication keys to the new nodes. The distributed security method does not require a trust center (coordinator control) and the security is managed by the routers therefore, any ZigBee router devices can provide authentication key to nodes joining the network. New joining nodes adopt whichever security method used by the network they join (Qiu, 2018). ZigBee is highly scalable and can cope with the wireless network complexities of greater than 250 nodes. It can also handle the dynamic behavior of these networks where the devices appear, disappear and re-appear in the network. Figure 2.12 illustrates the operations of the nodes in a ZigBee network (Qiu, 2018).

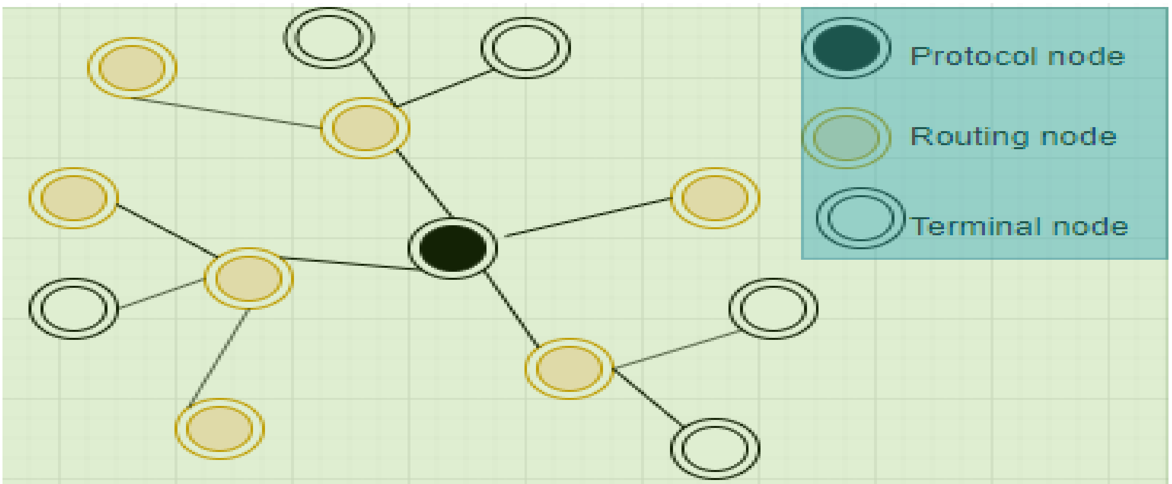

Figure 2.12: ZigBee Mobile Network

Chapter 3: Research Methodology

3.1 Introduction

Research methodology diagrams the procedure employed in directing the research. It outlines the method utilized in gathering the data on the variability of nodes and the mode of inclusion within the study. Remarkably, the chapter features the research design, data collection, data analysis, system analysis, system design, and system implementation. Besides, it depicts the framework system or rather system methodology utilized in the study for the research clarifying the outstanding advantages of the approach over other existing procedures. In conclusion, the ethical considerations of the research are laid out in the document as well as the limitations of these ethical concerns.

3.2 Research Design

This research was practical as it involved simulating the connectedness and awakenings of nodes in a dynamic network characterize by high mobility of the nodes. The approach sought to utilize the data generated from the ZigBee modules and the results modeled on X-CTU and MatLab tools to show the level of active connections in the network. The simulation was based on the stochastic capabilities of dynamic wireless networks and the experiments were necessary to generate data that helped in modeling and simulation. Therefore, the population and the sample, as well as the location of the study was not necessary.

The research approach used for the study was quantitative as the data generated from the ZigBee modules were represented in numbers and thus can be quantified. According to Eyisi (2016), quantitative research is a more reliable and objective way of using statistics to generalize a finding. The quantitative data was, therefore, the best way to collect data on mobile nodes for the dynamic networks as the movement of autonomous cars presents such kind of scenario where the cars change their positions every millisecond as the move. To ensure the mapping of active awakening and connectedness (synchronization) quantitative data is necessary. The research involved several simulations and therefore, the population and the sample to this study were irrelevant. The population and sample were only tied to the person (researcher) interacting with the system. Since the research was purely based on simulations, the dataset was formed from all the simulations done. The research was done within the university (Strathmore university) but it could have been done anywhere else so long as the working environment was enabling, conducive and spacious (at least two table-size working space due to ZigBee range).

3.3 Data Collection

The experiments were used as the primary data collection technique. The observation was also necessary. The experiments involved connecting the ZigBee modules and the Arduino to the X-CTU and the MatLab simulation tool. The random stochastic mobility models (Random Waypoint and Random Walk) were used to plot the distribution of the nodes on MatLab software through animation. Observation involved confirming any system changes on the behavior of the simulated nodes on the MatLab tool. The data from the ZigBee modules served as input to the simulation tools. Inbuilt algorithms based on stochastic mobility models are better to analyze the data as they provide different levels of accuracy and completeness (Cui, Yongquan, Chuanai, & Ning, 2018).

3.4 Data Analysis and Presentation

The data from the ZigBee modules and the simulated data with the same characteristics to those from the ZigBee modules were gathered and analyzed using the MatLab software. The Random Waypoint and the Random Walk mobility models were generated through this data. Synchronization data was also illustrated using the MatLab software. The models were applied to conclude from the experiments and the results were represented using images with accompanying analysis.

3.5 System Development Methodology

3.5.1 Agile System Development

Agile system methodology integrates the incremental and iterative aspects of system development. It is incorporated in a way that allows for constant development of the system while new requirements gathered are integrated into it. It also provides for a better way of handling corrections from testing and at the same time continuing with system development. This methodology reinforced the system design as the system required repetitive and periodic evaluation and testing of the various algorithms to achieve the best suitable system for active connections in a ZigBee mobile wireless network.

3.5.2 Design Thinking Approach

This design approach was integrated to capture the limitations of the agile system development design. It played an important role in the early stage of the proposal as it was used in identifying and defining the problem of the research. This design approach also involves a lot of prototyping thus complements the agile system development methodology which is iterative as well. The series of prototyping and testing helps in building the best prototype.

3.6 System Analysis

The developed system simulated active connection and synchronization between dynamic mobile nodes through the IEEE 802.15.4 MAC standard. The first step was ensuring the proper connections among the hardware devices (ZigBee module, Arduino Uno, and the computer) and then the configuration of the devices. The source code on the Arduino IDE was written first after the hardware connections then the source code on the X-CTU software tool and the Matlab software tool. The modules were then connected to the simulation tools and several virtual modules of the same characteristics were simulated to provide a high number of the nodes as well as enough datasets for the research. The dataset, therefore, was used to make inferences about the research. The system demonstrated the synchronization and the connections of the nodes in the network. The least sleep mode of 15 milliseconds was set in the ZigBee personal area network (PAN) coordinator. Random Waypoint and the Random walk were the algorithms selected to illustrate the mobility of the nodes in the network. From this analysis, the interaction between the different components of the system was illustrated by a sequence diagram and a synchronization network diagram in chapter 4.

3.7 System Design

The architectural structure of the system was composed of both software and hardware components. The hardware components comprise the ZigBee modules, Arduino microcontroller, jumper wires, and laptop. A desktop personal computer was used for the prototype. The software components comprised of the X-CTU and MatLab simulation tools, Arduino IDE, and the source code required for the device interconnections. There was serial communication between the ZigBee module, Arduino microcontroller, X-CTU, and MatLab tools. The connection messages were displayed on the X-CTU software tool. A diagram was used to highlight how synchronization in IEEE 802.15.4 MAC standard work. The same diagram explained what the inputs were, how they were processed, and what the outputs were. Furthermore, the diagram illustrated the level of synchronization among the devices configured for the experiment. Waiting time and the beacon duration was the basis of the acceptable level of active connections required for the research.

3.8 System Implementation

Several tools were needed to ensure that the system works correctly and produce accurate results. These tools include both software and hardware components.

3.8.1 Programming and simulation tools

The system used MatLab programming language for coding as well as for simulation. The applications and the customizable code in Matlab toolboxes help in the faster exploration of the various design techniques, testing live data, and analyzing simulation results and measurements ("MATLAB for Wireless Communications", 2019). Matlab also allows for creating algorithms and using existing algorithms to build standard-compliant systems that are dynamic thus creating a golden reference model to verify standard conformance ("MATLAB for Wireless Communications", 2019). Matlab has also the capability to incorporate other simulation tools as the inbuilt libraries for the several applications of the technology are easy to use. Additionally, I used the Attention (AT) commands to transmit packets between the ZigBee modules and to the Arduino microcontroller to check for connections. These AT commands can be configured, inputted and outputted using the X-CTU tool through both a console window and graphical user interface. The system also used micro python (python libraries in the Xbee radios) in API mode of the Xbees radios to configure personal area network coordinator and the Elliptic Curve Digital Signature Algorithm (ECDSA) used for encrypting the channels. Matlab and Python complemented each other and fitted well during the development of the system as some libraries in the Matlab tool were written in the python programming language. These files were compatible with Matlab and easy to understand and use.

3.8.2 Security Tools

The system provided some security measures which facilitated authentication in regards to the new mobile nodes joining the dynamic mobile network. The ZigBee personal area network implemented security in the form of a mesh network of the different nodes through the Elliptic Curve Digital Signature Algorithm (ECDSA). Each XBee radio generates a key pair (private and public keys) for secure communication and file sharing through the File System Manager (FSM).

FSM allows for easy and secure key distribution for the new nodes. ECDSA is the primary algorithm used by the XBee modules for digital signature and channel encryption.

3.9 Data Reliability and Validity

The unwavering quality and legitimacy of research information essentially influence the nature of the research. Reliability involves the proportion of consistency of results while validity tests whether the achieved values speak to the right values (Kimberlin & Winterstein, 2008). To achieve reliability and validity, a systematic and methodological technique was applied to the study. The experiments generated lots of data for the synchronization and the active connections testing as well the Random Walk and the Randon Waypoint mobility models illustrations. The validity of these data was achieved by comparing it to the golden reference of the algorithmic expectations. From the experiments and simulations of the research, the researcher found that combining the AODV routing protocol and the beacon-enabled IEEE 802.15.4 MAC and PHY synchronization protocol handled the complexity of the hidden new nodes in the network and also facilitated active connectedness and synchronization for services provision. Mobility was also tested using the Random Waypoint mobility model which illustrated the mobility aspect of the mobile nodes.

3.10 Ethical Consideration

The simulation process of the system involves programming the Xbee modules to mimic the movement of the mobile nodes in a network. As a result, practical ethics was required as not to interrupt or cause any harm or discomfort to any individual working in an office, a student studying, or individuals moving around. The proposal was taken through the ethical approval process by the Strathmore University – Institutional Ethics Review Committee (SU-IERC) which is accredited by the National Commission for Science, Technology, and Innovation (NACOSTI) in Kenya to conduct ethics reviews of research protocols in the human and behavioral sciences. After the ethical review process, the proposal was approved and a green light for the research was given by the SU-IERC.

Chapter 4: System Design and Analysis

4.1 Introduction

This chapter focuses on requirement analysis that the proposed synchronization and active connectedness in mobile networks should meet. It outlines the functional and non-functional requirements. It proceeds to define the environment the researcher used to achieve the implementation of the system. This is coupled with a description of the components the researcher utilized to meet the research goals outlined in the previous sections (Chapter 1, 2 and 3). The chapter then proceeds to present the analysis and the diagrammatic designs based on the required parameters to implement the proposed system.

4.2 Requirement Analysis

To achieve the objectives that were set out in chapter 1, this section of the dissertation looks at the various requirements that need to be met by the system.

4.2.1 Functional Requirements

The routing procedure should be able to ensure, first, connectivity between all the hardware components of the system.

All the components should be able to detect the commands being generated for the different algorithms from ZigBee modules in the network and forward them to appropriate MatLab controllers that will handle such connections and communications.

The components' interconnections should allow the various algorithms to be analyzed based on their characteristics.

The ZigBee modules require special activations as some inbuilt algorithms to the microcontroller need to pick up specific signal strength. The algorithm's output should be captured on the graphical user interface of the MatLab tool to prove the level of synchronization and active connectedness of the mobile nodes.

4.2.2 Non-functional requirements

The proposed system should be able to illustrate the characteristics of the various algorithms when static and mobile. Therefore, the system should show a certain level of integration where two or more algorithms can work together for better synchronization and security of the mobile nodes. The system should also demonstrate the interoperability and compatibility of the various software and hardware communication standards necessary in running systems in a mobile network environment.

4.3 Diagrammatic Representation of the system

4.3.1 System Architecture

Figure 4.1 shows how the ZigBee modules, Arduino modules, and the MatLab controllers will be able to be configured and to communicate together in running the various algorithms. The ZigBee microcontrollers have some inbuilt mobility algorithms configured to it and the MatLab tool would help in illustration and analysis of these algorithms. It is also important on configuring the algorithms to meet the intended purpose. Arduino microcontrollers use a different frequency and thus would provide an interoperability test among the components in a mobile network.

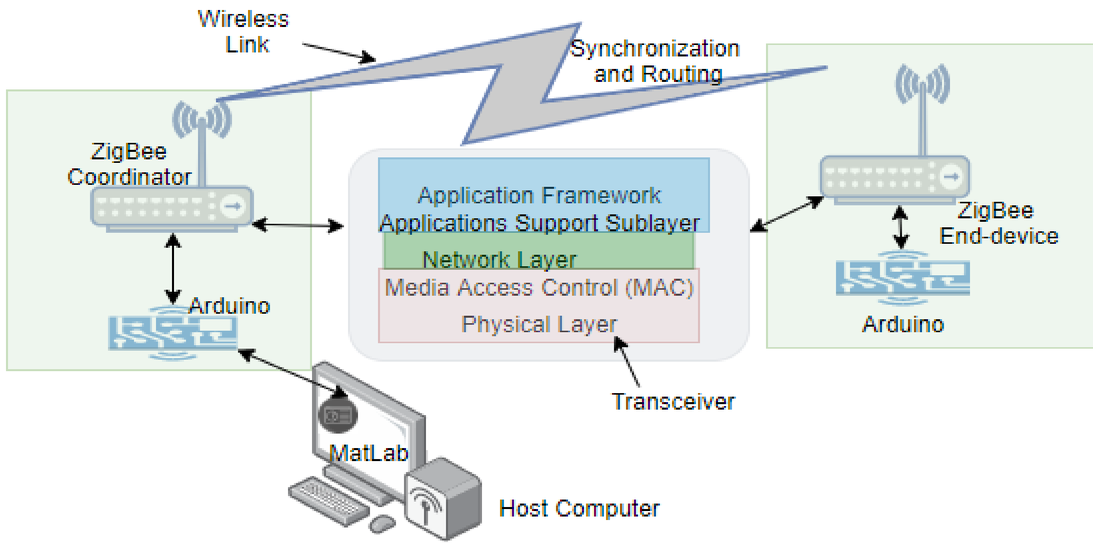

Figure 4.1: System Architecture

The MAC and the Physical layers define how the transmission works and the frequency and channels used. The Xbee devices can use the 868 Mhz, 915Mhz or 2.4Ghz at a low power rate and the modulation-demodulation technique applied to the research was the Offset Quadrature Phase Shift Keying (O-QPSK). The Xbee also implements Cyclic Redundancy Check (CRC) at these layers using a 16 bits data scheme. The network layer defines how the network is formed as well as the address assignment. The application layer defines the several addressing objects such as the clusters and devices profiles and the application sublayer is a security point of attachment to the ZigBee network and protocols. This is the layer where the keys generation algorithm is defined.

4.3.2 Use Case Diagram

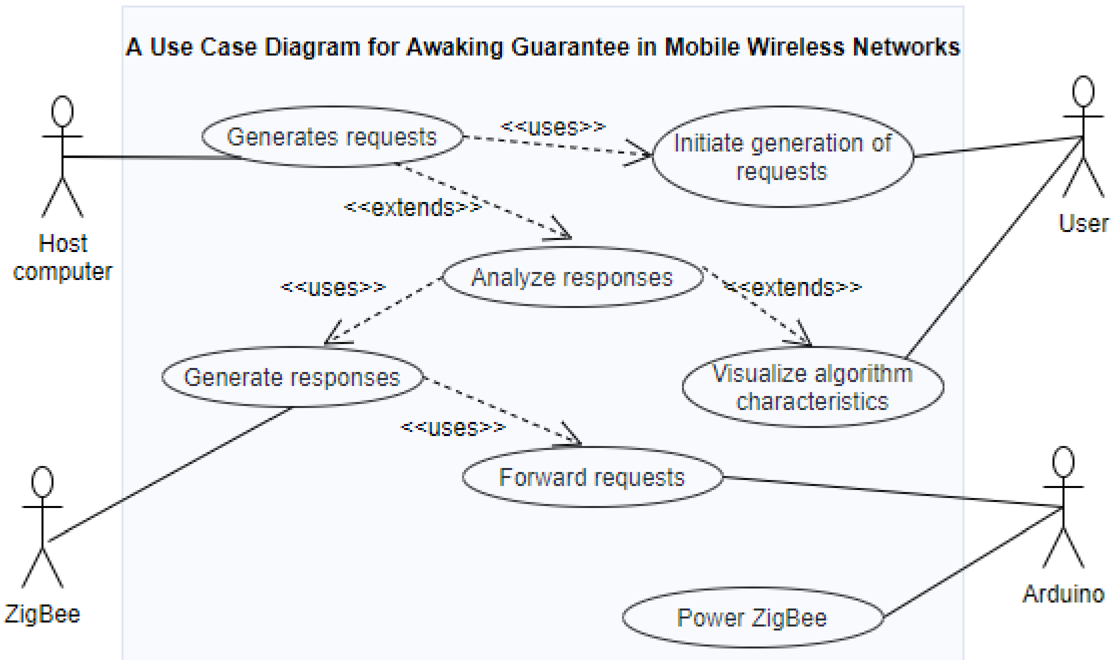

Figure 4.2: Use case Diagram

The use case diagram in Figure 4.2 shows the four possible actors that will interact with the system. The ZigBee module is responsible for the routing of messages and generation of timestamps IDs use for synchronization. It works as a router in a mobile wireless network with synchronization capabilities. The User in the system will be responsible for initiating the association requests that will target both the ZigBee and the Arduino modules as well as the remote microcontroller. The software tools such as the MatLab on the host computer is responsible for requests generation necessary for the ZigBee to scan and signal the nearby nodes and also to analyzed the responses. The system at this point will scan for the nearby microcontrollers and establish a connection link based on the signal strength which then generates timestamps for synchronization. The Arduino microcontroller is used as an intermediary microcontroller here to receive and forward requests and responses even though it has to be configured for this to achieve interoperability among the hardware components. Moreover, the ZigBee module inbuilt algorithms are used in routing and synchronization. Therefore, it is responsible for updating the routing table for the mobile wireless network. The distance vectors are also important as the signal strength varies with the position of the mobile node in a network. For any node with the same microcontrollers joining the mobile wireless network, the process of scanning, signaling, and synchronization are repeated.

4.3.3 Sequence Diagram

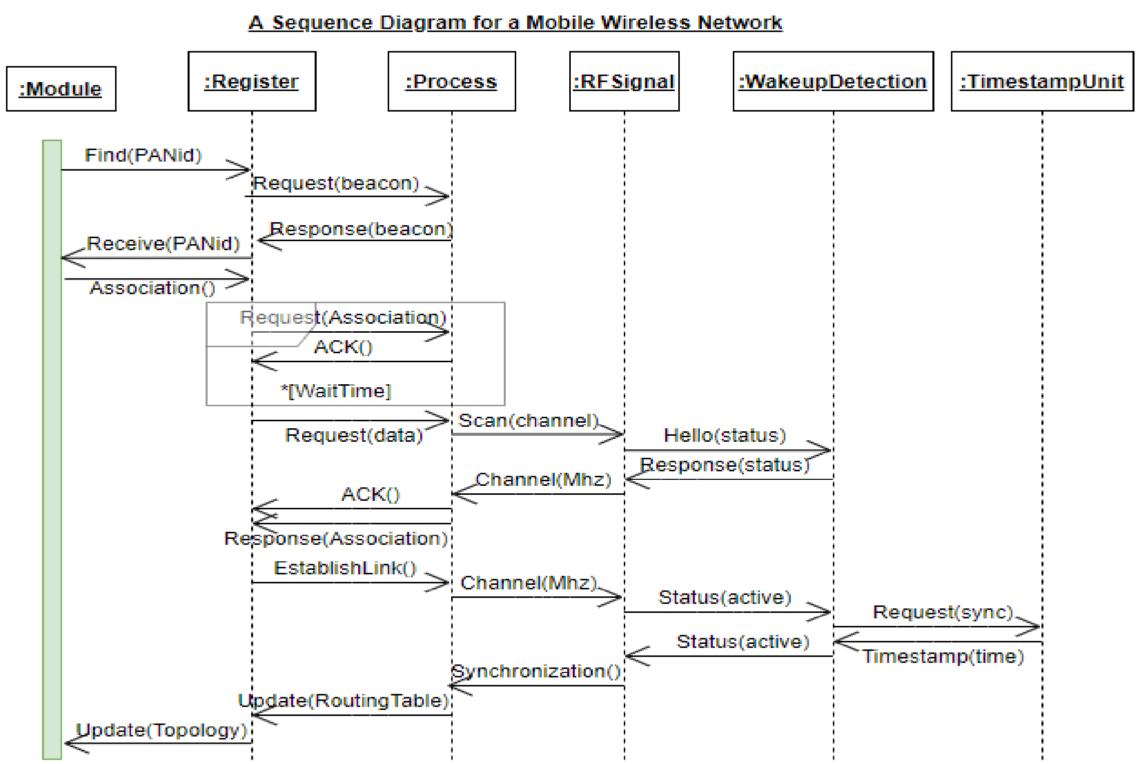

Figure 4 3: Sequence Diagram

The sequences diagram illustrated in Figure 4.3 outlines the series of events the various actors and the system will be interacting to enhance synchronization and awaking guarantee in the mobile wireless networks base on ZigBee and Arduino microcontrollers. The user who will work as a network administrator will be responsible for the general configuration of the components making up the system as well as the interconnection of the components. The responses analysis and timestamps IDs will be printed back to the user. The user will initiate an association request which will be responsible for searching nearby nodes in the network. Each node has a unique personal area network IDs (PANid) in which the ZigBee processor updates with the routing table for effective routing and synchronization. The processor also ensures that there are a constant synchronization and awaking guarantee in routing regardless of the communication link being idle through the wakeup detection unit attached to the timestamp unit thus suitable for wireless mobile nodes that can be implemented in the moving vehicles.

The beacons are used to synchronize the devices in the personal area network, identify the personal area network and it can also give a description of the frames through the data exchange among the devices. The personal area network coordinator set the beacon intervals and they vary from 15ms to even over 4 minutes. The beacon interval in this research will be kept as low as 15 ms to achieve high connectivity and synchronization. Acknowledgment (ACKs) are very important is a network and particularly in such wireless mobile network as the nodes join and leave the network at a higher frequency. To keep track of what nodes are currently in the network and their positional changes, the ZigBee acting as the master nodes or the nodes themselves send frequent "Hello" messages to the other nodes. This is also necessary to enhance the security of the network by checking out nodes without standard PANid. The frequency generating and analysis unit does this as the signal strength gives more information on the distance of the node from that node.

4.3.4 ZigBee Network Topology and Data Transmission

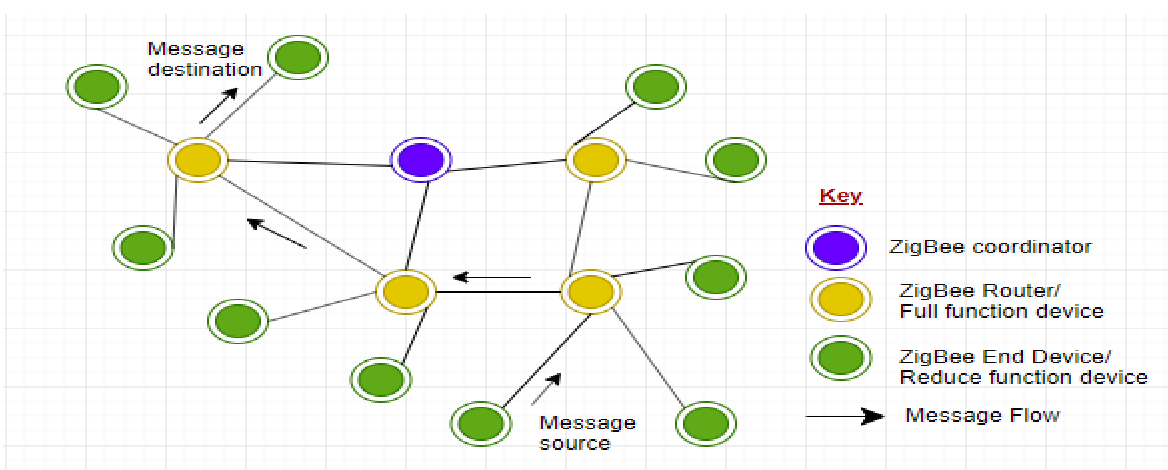

Figure 4.4: ZigBee Network Topology

A ZigBee network is made up of the router, coordinator, and end device. The primary role of the router is to transmit and receive data. It also keeps network information that can be used by the nodes joining the network. The coordinator creates and manages the network thus there can only be one in each network. It is also responsible for addressing the network. The end devices are the consumption devices and only send data periodically. It also controls the sleeping modes of nodes of the related key.

Data sent from the terminal of a serial port in a coordinator is received by all the routers. However, by default, a router sends data from the terminal of a serial port to only one coordinator as there is only one coordinator in the network. A programmer, however, can alter the router one-way communication in a ZigBee network to multicast (to specific devices) and broadcast (to all devices) through a set of commands. When the personal area network identify (PANid) of several nodes is the same, the nodes form a network automatically thus a ZigBee network is distinguished by a PANid. Several ZigBee networks can co-exist and do not interfere with each other as the PANid are unique. Modules join the respective network on power ON.

4.3.5 ZigBee Synchronization Network Setup

An IEEE 802.15.4 personal area network (PAN) is set up by a level process between ZigBee end devices and a coordinator. The end devices which include routers that would like to join a network, perform network scanning (active or passive). Active scanning is a situation where the device first sends a beacon request and then it does passive scanning whereas, in passive scanning, the device eavesdrop on the media to collect beacon frames from the coordinator which may have received the beacon request via active scanning. Using the beacons gathered in active scanning, the end device chooses the personal area network which it would like to associate with. The end device then sends an association request to the coordinator of that particular personal area network and the coordinator acknowledges it. Figure 4.5 illustrates both the active and passive scanning in synchronization of the ZigBee personal area network.

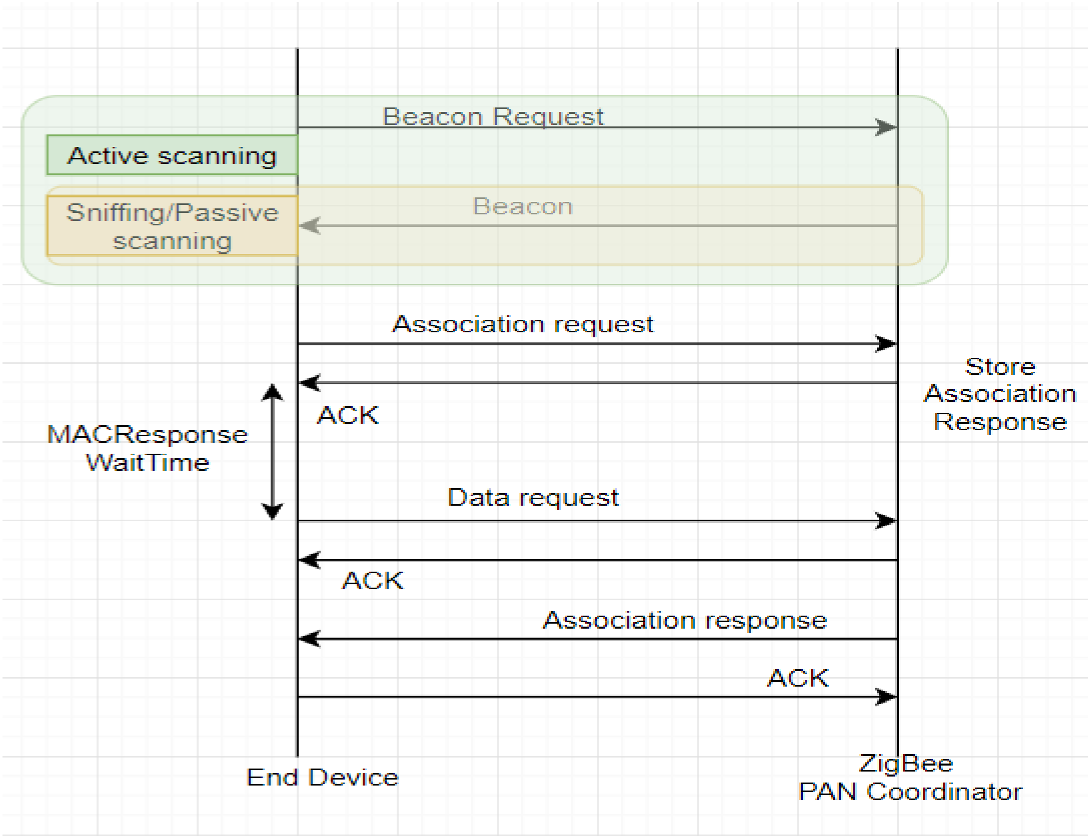

Figure 4.5: Synchronization in ZigBee Personal Area Network

In the IEEE 802.11 standard, the master node sends the association request acknowledgment by immediately transmitting an association response. On the contrary, the IEEE 802.15.4 standard allows the master node (coordinator) to store the association response locally and transmits only when the end device requests for the data and the coordinator acknowledges it. In the IEEE 802.15.4, this method of transmission is known as indirect transmission and this method is key for low energy devices used in low power low data rates networks such as the sensor networks. The devices in this type of network (sensor network) activate their radios periodically to check for any frames send to them rather than continuously having it up waiting to receive a frame immediately. The periods of activation may be configured differently depending on the type of services intended to be achieved. Some services require less sleep mode in the devices if signaling and synchronization are key while for other applications aiming for energy utilization, on-demand routing may be necessary. Upon receiving and acknowledging an association response, the end device is associated with the personal ara network. Eventually, the frames can be exchanged between the coordinator and the end device. The speed at which frames are acknowledged can also be influenced by their Acknowledgment Request indication.

Chapter 5: System Implementation and Testing

5.1. Introduction

This chapter focuses on how the proposed system design in the previous chapter was achieved. The chapter discussed the model components and the system setup that the researcher assembled during the implementation and testing phase. The chapter then moves to look at the actual implementation where design, implementation and testing the working of the system. The chapter further reports the outcome of the tests from the porotypes collected by the researcher gathered from the software tools and observation. The chapter concludes by giving the challenges the researcher experienced at the implementation phase as well as some of the approaches taken to solve these challenges.

5.2 Model Components

The software and hardware components that were used for the experiment to be realized in this research were informed by the nature of the tests to be conducted to achieve the objectives set out by the research.

Hardware components:

A personal computer with the following hardware components: 8 Gigabyte (GB) Random Access Memory (RAM), 1Terabyte (TB) hard drive disk storage and Intel Core i3 processor, Two (2) Arduino Uno microcontrollers, XBee 1mW wire antenna, and XBee 6.3 mW wire antenna (Mesh network) and the Jumper cables (male-male, male-female, female-female, male-male).

Software Components:

Windows 10 Host Operating System, XCTU v. 6.4.4, Windows x86/x64, Arduino 1.8.10 IDE and MatLab software R2018a.

5.3 Software Installations

At this point, the required software was installed to ensure that the proposed system functionalities are meeting its purpose as well as the interconnection of the components. The software installed is MatLab, Arduino/Genuino IDE, and the XCTU. Figure 5.1 illustrates the installed MatLab and its support package for Arduino hardware installation, installed XCTU and the Arduino/Genuino software tools.

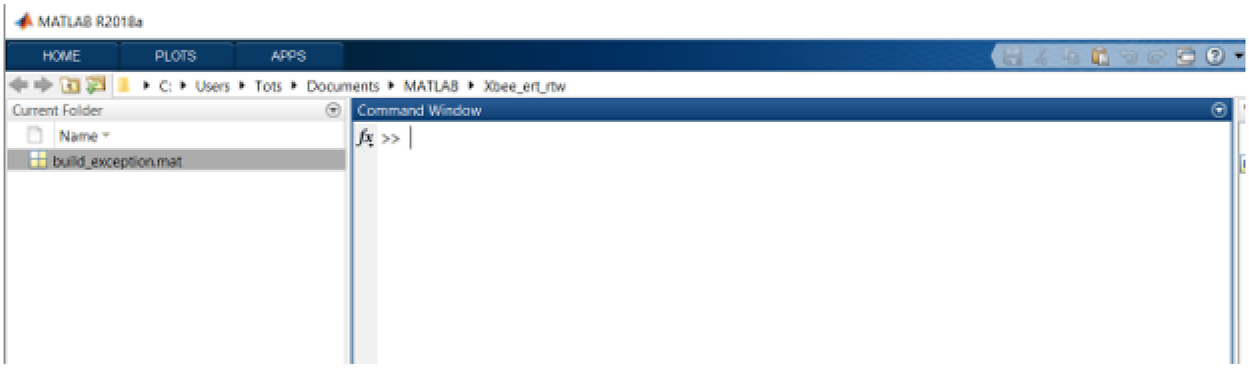

Figure 5.1: Installed MatLab Software

Figure 5.1 shows that the MatLab tool is installed and running. The version installed is MatLab R2018a. Version R2018a was chosen because it was easily available and could support the desired features of the research.

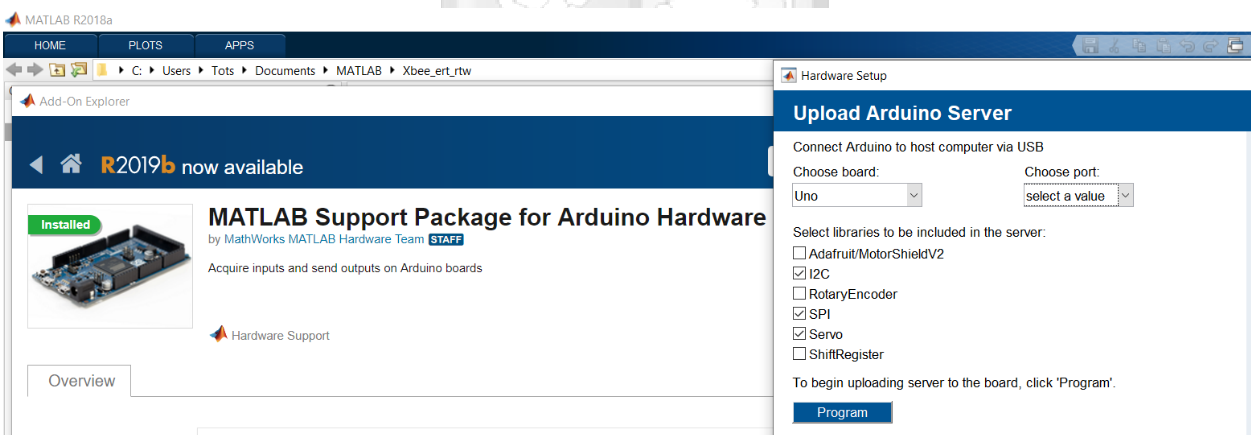

Figure 5.2: Support Hardware Packages Installation

Figure 5.2 illustrates the installation of the support package for Arduino hardware in the MatLab tool. MatLab supports several microcontrollers and hardware components and for effective integration, their respective MatLab support packages are required.

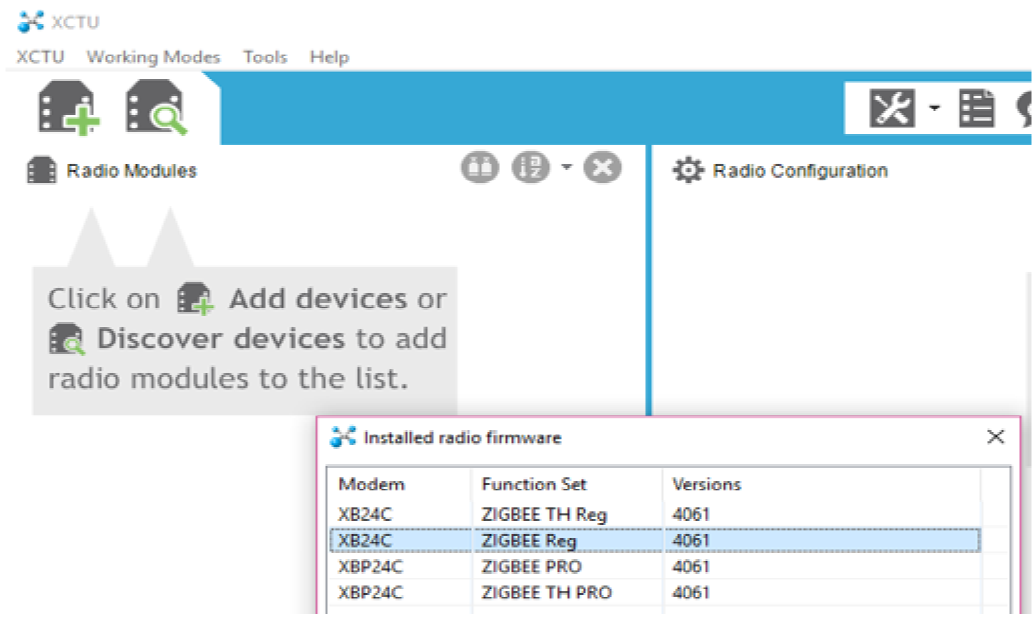

Figure 5.3: Installed XCTU Software Tool

Figure 5.3 shows the XCTU tool installed which is a configuration and test utility software. The software allows the developers to manage, set up, configure and test radio frequency (RF) modules such as the XBee radios. It also allows communication with several modules in a network as well as visualizing the network topology.

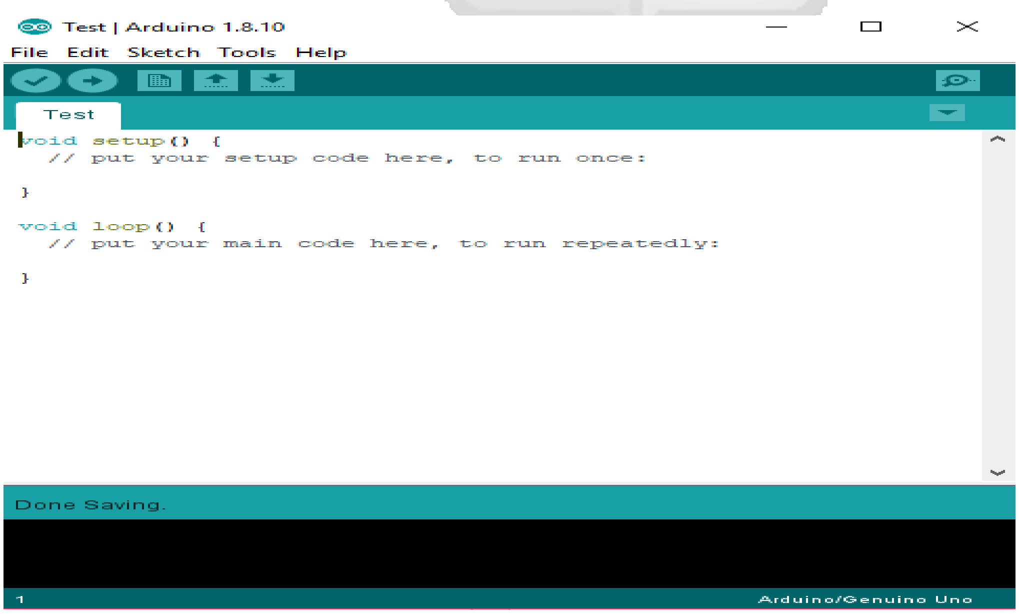

Figure 5.4: Arduino/Genuino Uno IDE

Figure 5.4 shows the version of Arduino/Genuino Uno IDE which is Arduino 1.8.10 used to read the data as well as configuring the microcontroller. The version of the IDE is very important to the version of the XCTU used as most times the latest XCTU firmware would require the latest version of the IDE.

5.3 Testbed Setup

The emulation of a wireless mobile network to enhance synchronization and awaking guarantee was done primarily by setting up a ZigBee network with two ZigBee radios. One radio being primarily the coordinator of a network and the other radio acting as a router and an end device. The radios were supported by the Arduino microcontrollers. The set up dependent on the other components and software like the XCTU and the Arduino to make it work effectively and minimize errors on coming up with the different prototypes. XCTU tool was used to configure, save and load the commands for the XBee radios while the Arduino/Genuino IDE was used to configure and load the code to the Arduino microcontroller. To test for the correct functionality of these components of choice by the researcher, each XBee radio was invoked and tested separately for the function of a coordinator, router and end device. Later a simple communication network topology was invoked and tested. The versions of the software tools played an important role in the implementation of the research objectives. Other components required to be integrated by the researcher to reduce human intervention once the simple network was established. The first phase of setting up the testbed was establishing a simple connection between the XBee and the Arduino microcontroller, then testing the XBee radios and finally testing the communication between the two XBee radios remotely.

5.3.1 Establishing Connection between Xbee Module and Arduino Microcontroller

The XBee modules acquired for the project work with 3.3v of power unlike other modules like Bluetooth which works with 5.0v. XBee modules would not work as they would be damaged on connecting to the 5.0v. To enable Arduino microcontroller, communicate with the XBee module, the following connections are used:

- Connect 3.3v pin of Arduino to pin 1 of the XBee module.
- Connect the ground(GND) pin of Arduino to pin 10 of the XBee module.
- Connect Tx (pin 1) of Arduino to Tx (pin 2) of XBee
- Connect Rx (pin 0) of Arduino to Rx (pin 3) of XBee

The Arduino is then connected to the power source using the Arduino USB cable to power both devices. Figure 5.5 illustrates the connections between the Arduino and the XBee module.

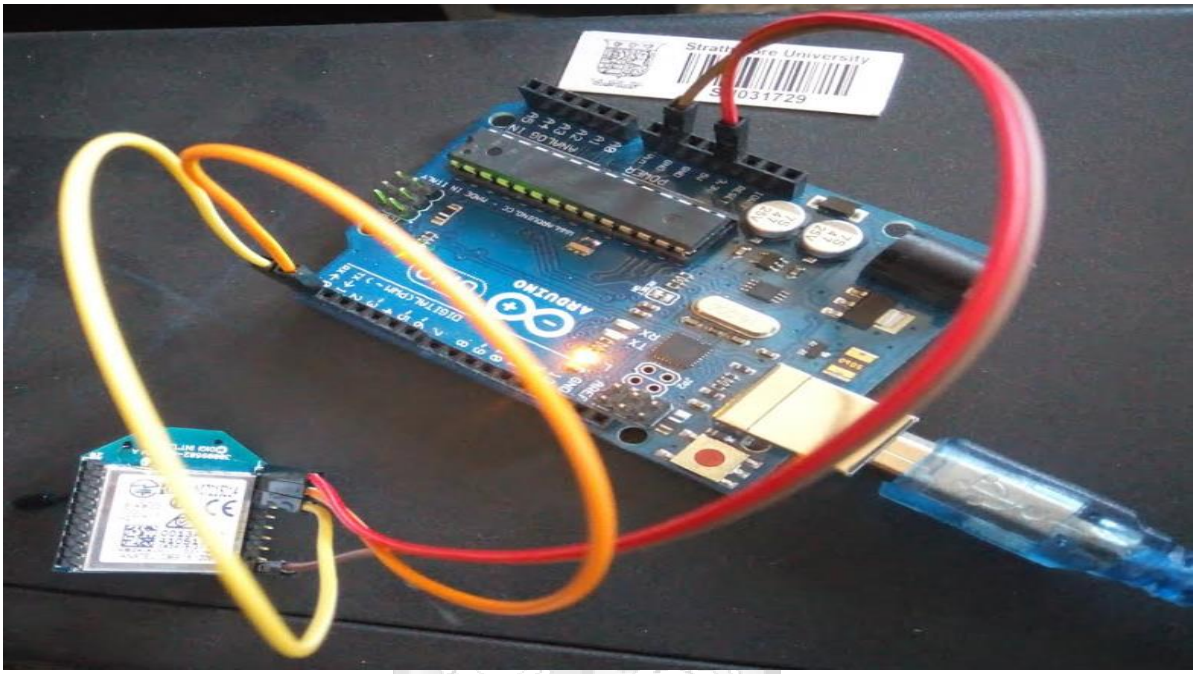

Figure 5.5: Arduino and the XBee Setup

5.3.2 Testing the XBee Modules

The testing phase of the XBee modules involved configuring the Modules; one as a coordinator and the other as an end device (full function device/router and the reduced function device), using attention (AT) commands (instructions used to control a modem and every command starts with AT/at) to test the working of the XBee, generating key pair for the module and configuring the XBee to set up a network automatically with the nearby XBee radios. A coordinator is a protocol node in a ZigBee network hence controls all the other nodes in the mobile wireless network. Figure 5.6 illustrates the configuration of the XBee module as a coordinator node in a mobile network.

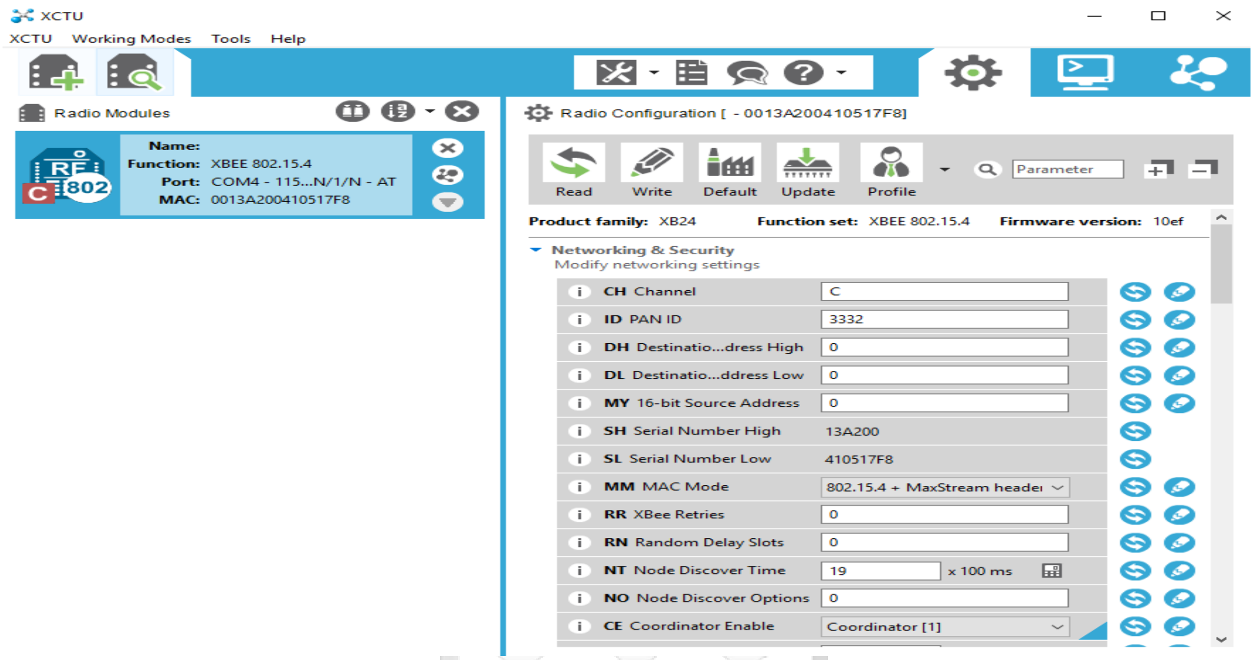

Figure 5.6: Configuring XBee 1Mw Wire Antenna as the Coordinator

Various modules like GPRS, GSM, and XBee use AT commands to get the required information in the module and for configurations as long as it supports modulation and demodulation of the signals. To test the working of various components in the XBee module, the same AT commands are used through the console configuration window thus considering communication packets in the test. Figure 5.7 shows the sending of test packets in the XBee module using the AT commands.

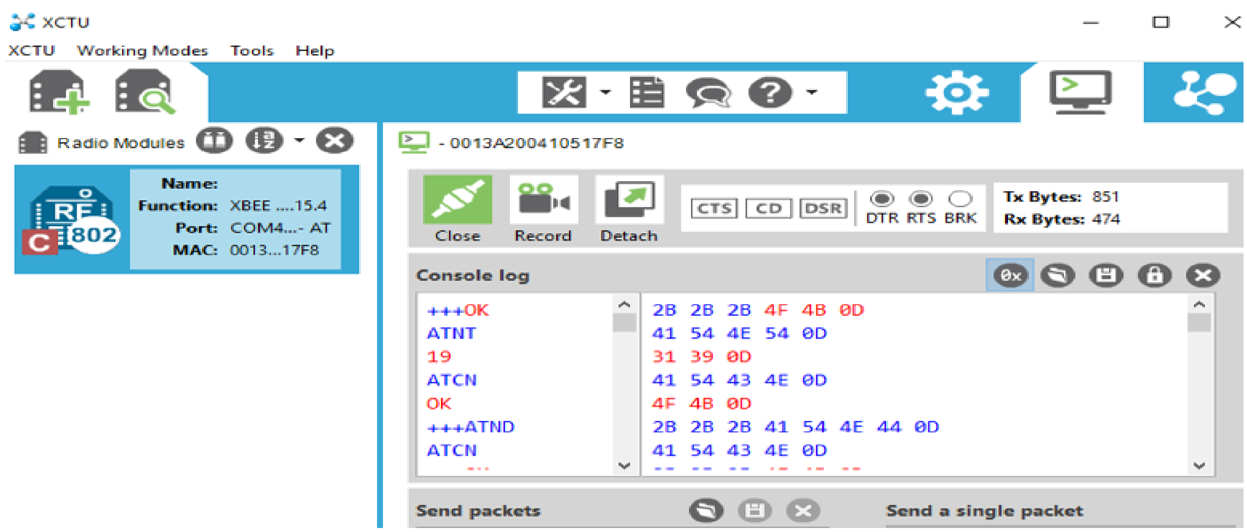

Figure 5.7: Testing the XBee Coordinator using AT Commands

Communication in the XBee network supports encryption and digital signature using the Elliptic Curve Digital Signature Algorithm (ECDSA). Each XBee radio generates a key pair (private and public keys) to secure the channel and file sharing as XBee supports it through File System Manager (FSM). FSM allows for easy and secure key distribution for the new nodes joining the network and alerting the network in the case of the nodes leaving the network. The protocol is supported by Advanced Encryption Standard (AES). Figure 5.8 illustrates the use of ECDSA for key generation.

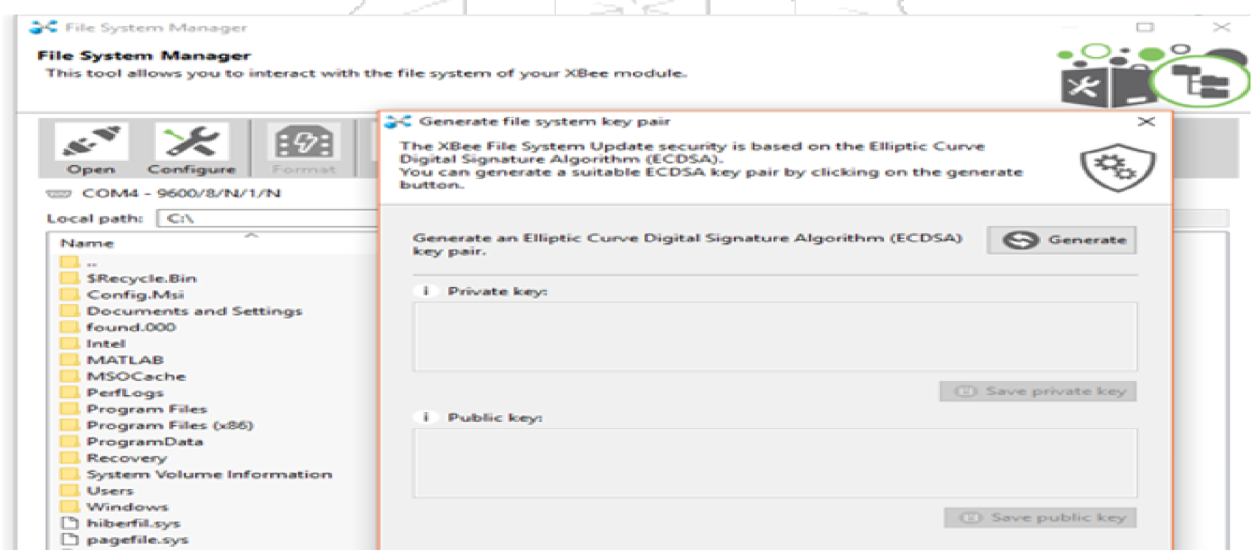

Figure 5.8: Generating Key Pair Using ECDSA by XBee

ZigBee network is by default a mesh topology due to the mobility of the nodes joining the network as other nodes leave the network. A new node has to scan the network for basic information like network identity (ID) and at the same time, nodes tend to discover each other through the neighbor discovery mode in a ZigBee network. A new neighbor node discovered and can only join the network if it is configured with the same details. Figure 5.9 illustrates the configuration of the ZigBee network using the XCTU tool.

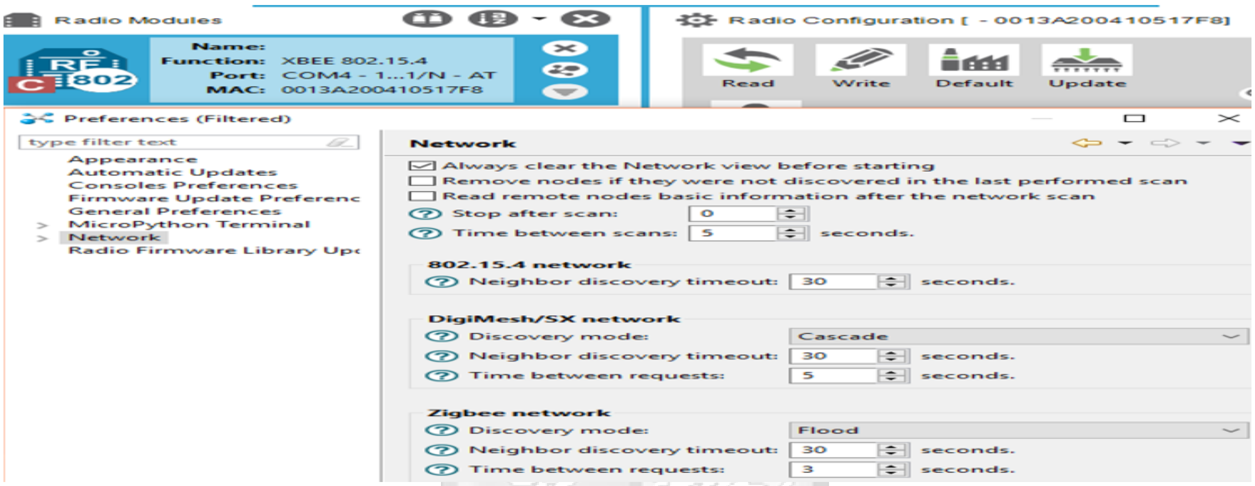

Figure 5.9: ZigBee Network Configuration

5.3.3 Communication Between the Two XBee Modules

This phase of setting up the testbed ensures that there is a communication between to XBee radios which one is the coordinator and the other being the router and the end device. The other radio is used interchangeably as the router and the end device. The ZigBee network has one coordinator and for this experiment, one radio is to be used without changing to an end device or a router. For routing and synchronization, capturing the signal strength is very important as it can be used to indicate the distance of the nodes from each other. Figure 5.10 shows how the X-CTU tool can be used to discover XBee radios. The figure also shows one radio as coordinator (labeled C) and the other as a router (labeled R).

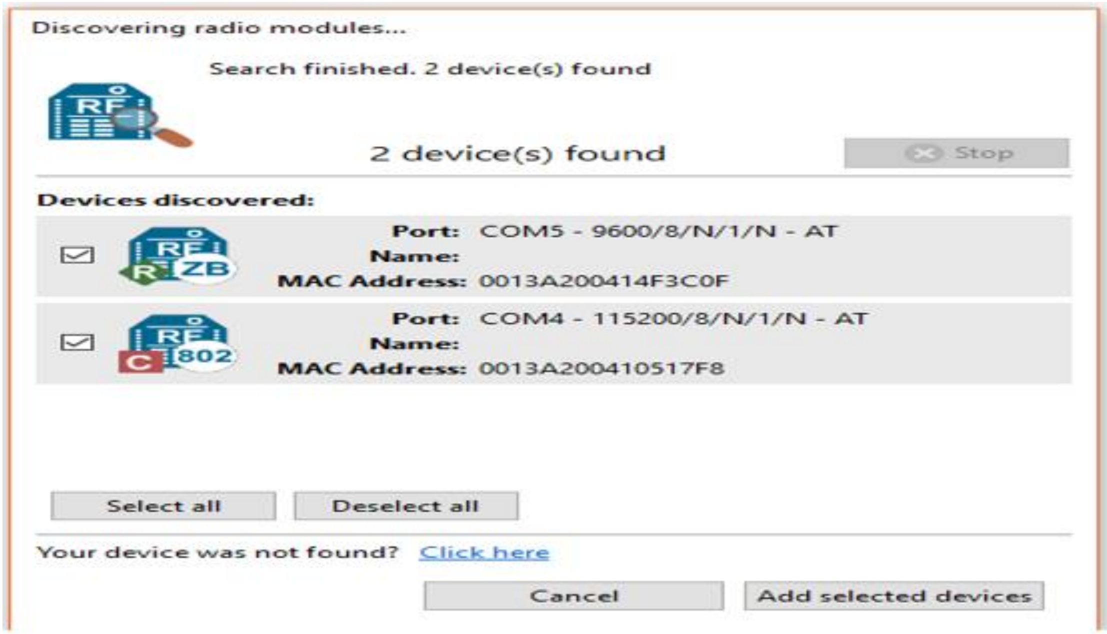

Figure 5 10: ZigBee Coordinator and Router in One Network

5.4 System Implementation

The system implementation is to be achieved through four phases; topology design, device, and service discovery to ensure connectivity between the nodes, matching the packets to the respective services, writing the source code for the random mobility models and finally checking synchronization between the devices in the ZigBee network based on the synchronized and asynchronous, CSMA-based IEEE 802.15.4 MAC.

5.4.1 Topology Design

The initial stage of this phase was to design a network topology that matches the diagram illustrated by Figure 4.4: ZigBee Network Topology in chapter 4 above. This representation was done using the network mode in the XCTU tool which offers custom and advanced network topology definitions. The topology was also written in python using the Micro Python packages in the X-CTU tool to support routing and synchronization features of the network. The python packages are built specifically for the radios (ZigBee) and can be used to enhanced low-level services and security of the radios and the network itself. The researcher utilized these packages for security, synchronization, awaking guarantee (very low sleep mode in the radios), and routing. The python files created by the researcher in defining the services mentioned above were saved in the folder with the other X-CTU configurations and the Arduino configurations.

5.4.2 Link Discovery to Ensure Connectivity Between the Nodes

In any ZigBee system, the coordinator is only one and it fully coordinates all the activities of the network as well as selecting the personal area network identifier (PANid) to be used by the network. It also chooses the physical radio channel on which the devices will communicate (Vasseur & Dunkels 2010). The coordinator functionalities are reduced to a router after the bootstrapping process. The coordinator is logically attached to the routers and it utilizes the routers to generate the mesh topology in the network since the routers provide network range extension (Vasseur & Dunkels 2010). Even though both the ZigBee routers and the coordinator facilitate new nodes to joining the network and route frames, the routers build a special network amongst themselves from their paths through the exchanged frames.

ZigBee coordinator once acting as a ZigBee router, remain active always just like the other routers in the network. The coordinator function will be off at this time hence would not receive traffic forwarded to it. However, it would resume its function periodically to check the messages at the ZigBee router attached to as the associated ZigBee router buffers the data and sent when the coordinator function is awake or on request from the ZigBee end device. The wake-up schedule for the ZigBee device is specified by the application developer. In this experiment, the wake-up schedule was set to 15 seconds for the ZigBee coordinator only. The ZigBee router and the end device were not allowed to go into sleep mode for effective awaking and synchronization features. The researcher had to confirmed that a custom topology would work and therefore, the first thing was ensuring that there was connectivity between the nodes in the topology. This was achieved in two phases: setting up one ZigBee module as a coordinator and the other as the router where the router did the self-discovery. The researcher configured the router to use the same PANid and the physical radio channel as the coordinator. Secondly, the researcher configured one device as a coordinator and the other as an end device where the end device did the self-discovery. The researcher configured both devices to use the same PANid and the physical radio channel. Self-discovery allowed the devices to advertise their identities. The next phase after the discovery of the devices was to ensure that the devices could communicate by sending packets to each other based on any network changes like such as the range, sleep mode, and the path. The diagram in Figure 5.11 shows the discovery process in a ZigBee module.

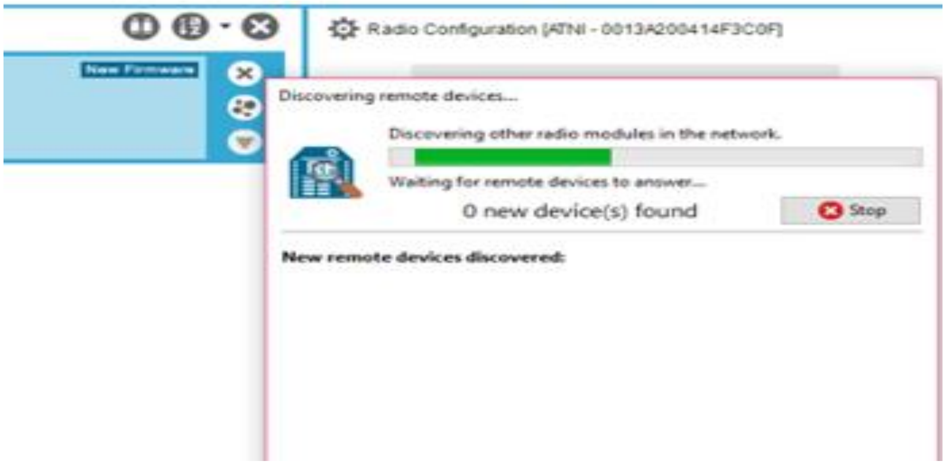

Figure 5 11: Discovery Process in ZigBee Module

5.4.3 Matching Packets to Services.

The several packets exchanged by the ZigBee modules were for different services. However, some of the services which were of key interest to the researcher were the synchronization, active connection (awaking guarantee), routing, and security. Some packets can be meant for more than one service due to a high level of service integration and interdependence such as synchronization, routing, and security. One feature influences how the other features would be implemented. AT commands and packets exchanges by the ZigBee modules provides information on the routing protocol which is AODV. The packets also provided some information about synchronization as the very little delay mean a higher level of synchronization between devices thus higher security measure against replay attacks. The packets to the coordinator show the level at which sleep mode can affect synchronization and communication between the devices. Setting low sleep mode in the coordinator device and no sleep mode in the end device shows a low synchronization complexity in the ZigBee network which can then be most appropriate to networks requiring awaking guarantee of nodes and active synchronization all the time. For example, a network of drones used in Agriculture and military operations or a network of autonomous cars to improve signaling and crash avoidance.

5.4.4 Building Random Mobility Model

Since the purpose of the research is testing on the connectedness of the wireless mobile nodes, the researcher decided to focus on the Random Mobility Models. The models are intended to mimic the movement of driverless automotive and drones thus more concentration on the Random Waypoint (RWP) and the Random Walk (RW) mobility models. The source code required to generate the Random Waypoint Mobility Model has several parts and Figure 5.12 shows the source code used to initialize the model. Figure 5.13 shows some parts of the source code used to generate the Random Walk Mobility Model. Unlike Random Waypoint, Random Walk shows the connection of the nodes in the form of a line. Random Waypoint assumes a mesh connection of nodes which is the typical environment build by highly mobile nodes.

```
Editor - C:\Users\Tots\Documents\MATLAB\Random_Waypoint\Generate_Mobility.m
Generate_Mobility.m

function s_mobility = Generate_Mobility(s_input)
    %The Random Waypoint mobility model.
    global s_mobility_tmp;
    global nodeIndex_tmp;

    s_mobility.NB_NODES = s_input.NB_NODES;
    s_mobility.SIMULATION_TIME = s_input.SIMULATION_TIME;
    for nodeIndex_tmp = 1:s_mobility.NB_NODES
        %%%%%%%%%%%%%%%%%%%%%%%%%%%Initialize:
        s_mobility_tmp.VS_NODE(nodeIndex_tmp).V_TIME = [];
        s_mobility_tmp.VS_NODE(nodeIndex_tmp).V_POSITION_X = [];
        s_mobility_tmp.VS_NODE(nodeIndex_tmp).V_POSITION_Y = [];
        s_mobility_tmp.VS_NODE(nodeIndex_tmp).V_DIRECTION = [];
        s_mobility_tmp.VS_NODE(nodeIndex_tmp).V_SPEED_MAGNITUDE = [];
        s_mobility_tmp.VS_NODE(nodeIndex_tmp).V_IS_MOVING = [];
        s_mobility_tmp.VS_NODE(nodeIndex_tmp).V_DURATION = [];

        previousX = unifrnd(s_input.V_POSITION_X_INTERVAL(1),s_input.V_POSITION_X_INTERVAL(2));
        previousY = unifrnd(s_input.V_POSITION_Y_INTERVAL(1),s_input.V_POSITION_Y_INTERVAL(2));
        previousDuration = 0;
        previousTime = 0;
        Out_setRestrictedWalk_random_waypoint(previousX,previousY,previousDuration,previousTime,s_input);
```

Figure 5.12: Random Waypoint Mobility Model Source Code

```
Editor - C:\Users\Tots\Documents\MATLAB\random_walk\random_walk.m
Generate_Mobility.m    random_walk.m

    step=1;
    Xi=0;
    Yi=0;
    Zi=0;
    X(1:N)=0;
    Y=X;
    Z=Y;
    X(1)=Xi;
    Y(1)=Yi;
    Z(1)=Zi;

    %-------------------------------------------------------

        figure(1)

for i=2:N
        if(rand(1,1)>=0.5)
            X(i)=X(i-1)+step;
        else
            X(i)=X(i-1)-step;
        end
        if(rand(1,1)<=0.5)
            Y(i)=Y(i-1)+step;
        else
            Y(i)=Y(i-1)-step;
        end
        if(rand(1,1)>=0.5)
            Z(i)=Z(i-1)+step;
        else
```

Figure 5.13: Random Walk Mobility Model Source Code

5.4.5 Buiding CSMA-based IEEE 802.15.4 MAC Synchronization

The researcher in this section creates a source code matching the physical network of the ZigBee network (IEEE 802.15.4) of 2 nodes: a Personal Area Network (PAN) coordinator and an end device. Since the X-CTU network simulator was configured in Figure 5.6 to process all devices at increments of a single backoff duration (20 symbols, 0.32 ms), the source code should also match the configuration mode on the devices. Figure 5.14 illustrates the association of the first device with the network.

```
symbolsPerStep = 20;
chipsPerSymbol = 32;
samplesPerChip = 4;
symbolRate = 65.5e3; % symbols/sec
time = 0;
stopTime = 5; % sec

% Create PAN Coordinator
panCoordinator = lrwpan.MACFullFunctionDevice('PANCoordinator', true, 'SamplesPerChip', 4, ....
  'PANIdentifier', 'Coordinator', 'ExtendedAddress', [repmat('0', 1, 8) repmat('7', 1, 8)], ...
  'ShortAddress', 'XB1A');

% Create first end-device:
endDevice1 = lrwpan.MACReducedFunctionDevice('SamplesPerChip', 4, ...
  'ShortAddress', 'XB2B', 'ExtendedAddress', [repmat('0', 1, 8) repmat('3', 1, 8)]);

% Initialize device inputs
received1 = zeros(samplesPerChip * chipsPerSymbol * symbolsPerStep/2, 1);
received2 = zeros(samplesPerChip * chipsPerSymbol * symbolsPerStep/2, 1);

while time < stopTime
  % Pass the received signals to the nodes for processing. Also, fetch what
  % they have to transmit:
  transmitted1 = panCoordinator(received1);
  transmitted2 = endDevice1(received2);

  % Ideal wireless channel, where both nodes are within range:
  received1 = transmitted2; % half-duplex radios, none receiving while transmitting
  received2 = transmitted1;

  time = time + symbolsPerStep/symbolRate; % update clock
end
```

Figure 5.14: PAN Coordinator and First End Device Association

Once the first end device has been associated, data frames are randomly pushed into the link between the end device and the PAN Coordinator. To create a detail view in the case more devices are added into the network, the researcher created a logical third ZigBee end device in the network. On adding this device as part of the network, it joins the Personal Area Network and packets are subsequently exchanged between the coordinator and both end devices as the end devices only transmit frames to coordinators. Figure 5.15 shows the source code used to simulate and associate the third device.

```
% Create second end-device:
endDevice2 = lrwpan.MACReducedFunctionDevice('SamplesPerChip', 4, ...
  'ShortAddress', '0002', 'ExtendedAddress', [repmat('0', 1, 8) repmat('4',
1, 8)], 'Verbosity', false);
% Suppress detailed output:
endDevice1.Verbosity = false;
panCoordinator.Verbosity = false;

% Initialize input
received3 = zeros(samplesPerChip * chipsPerSymbol * symbolsPerStep/2, 1);

stopTime = 10; % sec
while time < stopTime
  % Pass the received signals to the nodes for processing. Also, fetch what
  % they have to transmit:.,.
  transmitted1 = panCoordinator(received1);
  transmitted2 = endDevice1(received2);
  transmitted3 = endDevice2(received3);

  % Ideal wireless channel, where all nodes are within range:
  received1 = transmitted2 + transmitted3; % half-duplex radios, none
receiving while transmitting
  received2 = transmitted1 + transmitted3;
  received3 = transmitted1 + transmitted2;

  time = time + symbolsPerStep/symbolRate; % update clock
end
```

Figure 5.15: PAN Coordinator and the Second End Device

5.5 System Testing

The experiments conducted by this research were aimed at testing whether the synchronization of the nodes in the network can be achieved with the minimal sleep mode in the devices. The method of synchronization selected for the connectedness of the nodes matched the intended aim. The personal area network (PAN) coordinator was responsible for managing the network and the end devices could communicate between themselves. The configurations and the source code created were integrated for the full functioning of the system. Besides, the underlying routing protocol selected for the network was Ad hoc On-Demand Distance Vector (AODV) and all the devices were configured to work with the protocol including the simulated nodes used to generate the mobility models.

5.5.1 Xbee Communications

Figure 5.16 shows serial communication between a ZigBee Coordinator and a ZigBee device in a network. The figure below shows the message “hello coordinator” in blue text send by the end device (on the left) and the same message is received by the coordinator (to the right). The message is in red text. The coordinator (to the right) responds by sending a “ hello end device” message (in blue text) and the same message is received by the end device (to the left). The Xbee devices encode and decode any character text into the equivalent hexadecimal bits as illustrated in Figure 5.16. The end device acts as a router in this experiment.

Figure 5.16: ZigBee Coordinator and the ZigBee Router

5.5.2 Random Waypoint mobility model

To generate the Random Waypoint mobility model, the data from the two devices were not enough hence a need to simulate more end devices with the same characteristics. These devices were simulated with the features to change position every half a second in a random direction at a constant speed. A boundary limit set was a twenty (20) meters square. Figure 5.17 shows the Random Waypoint mobility model animated on the MatLab tool using the same dataset. Table 5.1 shows the input parameters for simulating the model.

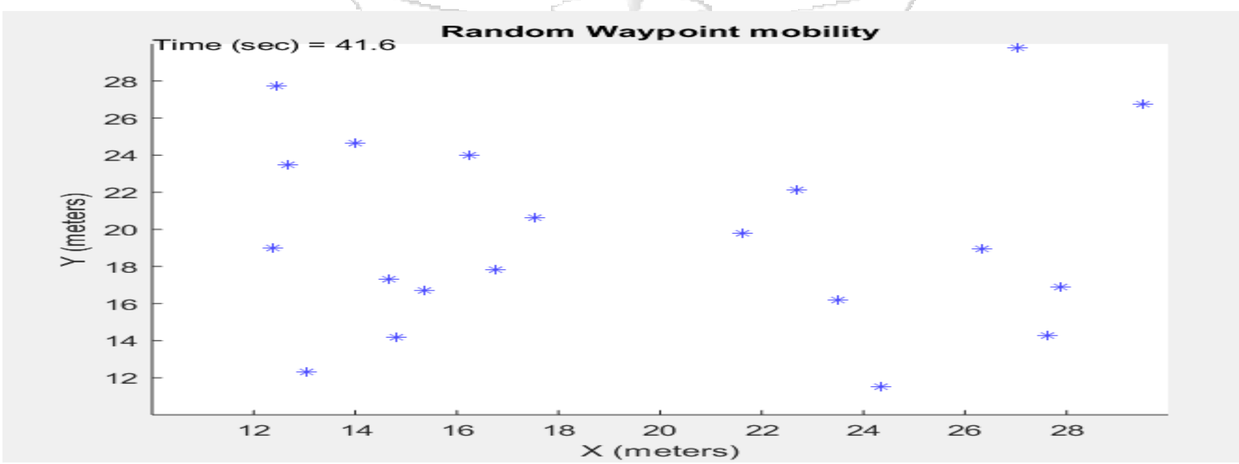

Figure 5.17: Animated Random Waypoint (RWP) Mobility Model

Table 5.1: Simulation Parameters for RWP Mobility Model

Simulation Parameters	**Values**
Position x interval	10 to 30m
Position y interval	10 to 30m
Speed interval	0.2 to 2.2 m/s
Pause interval	0 to 1 sec
Walk interval	2 to 6 sec
Direction interval	-180 to 180 degrees
Simulation time	500 sec
Number of nodes	20
Mobility model	RWP
Network connectivity time	0.1 sec
Simulation area length	20 m

5.5.3 Random Walk mobility model

The same dataset generated in figure 5.17 was used to generate a Random Walk mobility model on a 3D coverage to capture the connectivity interval of the nodes as they move on the set range. Figure 5.18 shows a 3D displayed of the Random Walk mobility model.

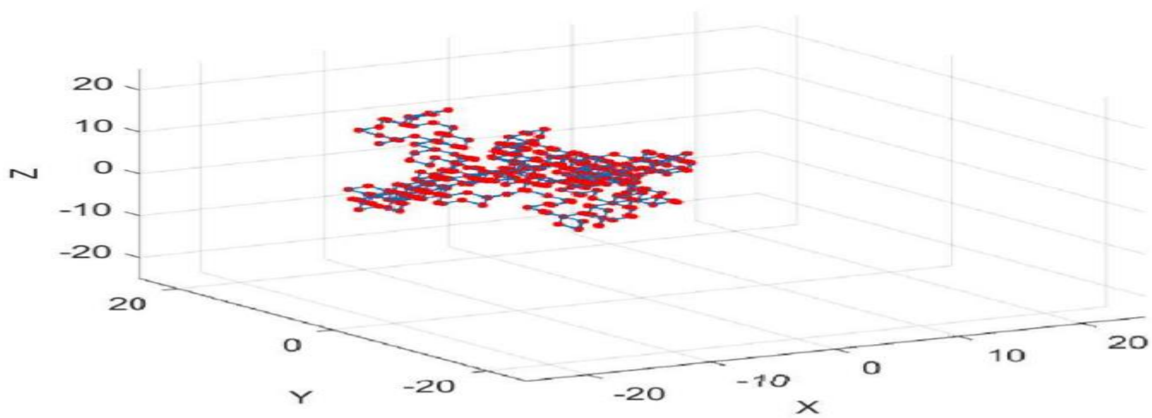

Figure 5.18: Random Walk Mobility Model

5.5.4 Synchronization in the Established Network

The IEEE 802.15.4 MAC and PHY layers are the foundation of the higher-layer standards, such as ZigBee and thus, the standard provided the best environment for testing the Low-Rate Wireless Personal Area Network (LR-WPANs) setup such as the one for this research. This experiment provided a detailed simulation of the non-beacon-enabled, CSMA-based IEEE 802.15.4 MAC. As illustrated by Figure 5.19, the nodes that would like to join a network carry out inactive (passive) scanning by sniffing to collect beacon frames from PAN coordinators. On gathering the beacons during the scan, the end device chooses the ZigBee PAN coordinator to associate and then transmits an association request to it. The coordinator then acknowledges it if the PAN identifier matches.

```
Command Window
>> Xbee_devices
FFFE: ********* Adding Beacon Request frame to the queue
FFFE: Passive scanning for 1584 steps
FFFE: Processing next frame from the queue
FFFE: Initializing transmission; backoff delay = 1 steps
FFFE: Backoff delay = 1 steps -> 0 steps
FFFE: Carrier sensing: Medium is idle.
FFFE: Clear to transmit
FFFE: Transmitting Beacon Request
FFFE: IFS offset = 0 samples
FFFE: Transmitting 1-1280 of 2050
Found preamble of OQPSK PHY.
Found start-of-frame delimiter (SFD) of OQPSK PHY.
FFFE: IFS offset = 0 samples
FFFE: Transmitting 1281-2050 of 2050
FFFE: Finished transmission
FFFE: Need to wait for SIFS (12) symbols. Offset = 12, next IFS = 4
FFFE: Entering passive scanning
1234: PHY decoded IEEE 802.15.4 frame
1234: Need to wait for SIFS (12) symbols. Offset = 12, next IFS = 4
CRC check passed for the MAC frame.
1234: *********** Received frame type = MAC command
1234: *********** Received MAC Command type = Beacon request
1234: ********* Adding Beacon frame to the queue
1234: next IFS = 4
1234: Processing next frame from the queue
```

Figure 5.19: Passive Scanning Between Coordinator and the One End Device

Figure 5.20 illustrates the communications between the third device and the PAN coordinator on the PHY and MAC layers of the IEEE 802.15.4. The Offset quadrature phase-shift keying (OQPSK) was used as the modulation and the demodulation technique among the devices.

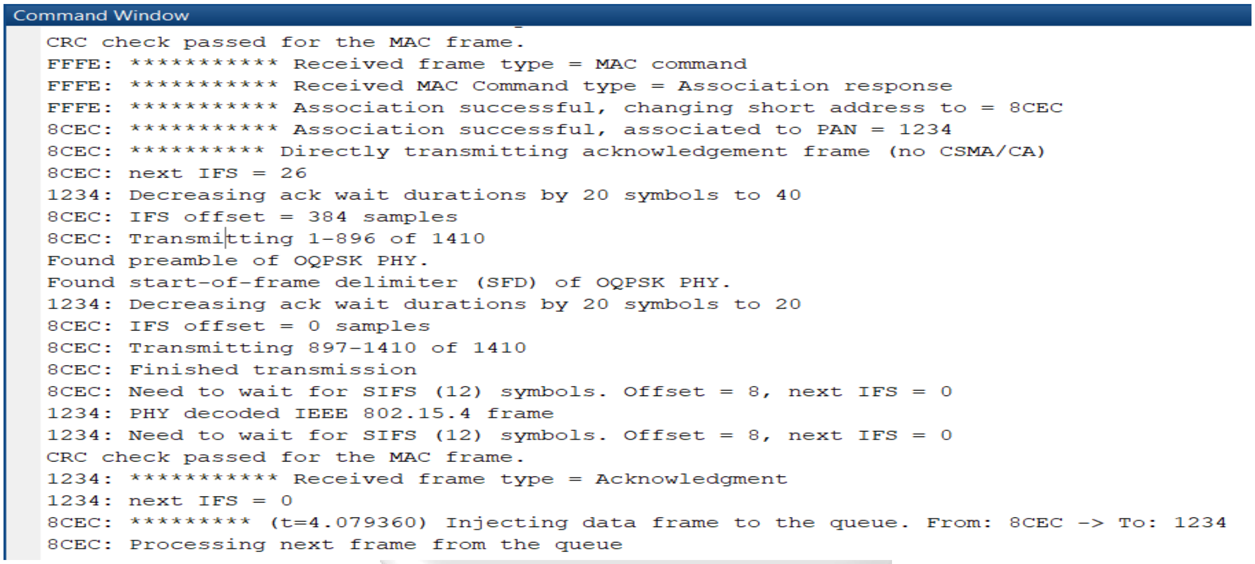

Figure 5.20: ZigBee Coordinator and the Third End Device Synchronization

5.5.5 Nodes Connectivity

Connectivity of the nodes in the network is affected by several factors such as the distance, speed, change in position, and the set sleep mode in the nodes. This, therefore, can take linear, quadratic, cubic, and more polynomial expressions. Figure 5.21 shows an ideal connectivity level in a fixed network considering constant bandwidth. The connectivity assumes a linear expression which is a line of best fit. Figure 5.22 illustrates the level of connectivity in a dynamic network of very mobile nodes based on a simulation on a Randon Waypoint algorithm. The fit curves illustrate the effect of connectivity as the number of mobile nodes increased. The fit curve labeled 1 shows the connectivity of at most seven (7) nodes, fit curve labeled 2 shows the connectivity of almost fourteen (14) nodes, and the fit curve labeled 3 shows the connectivity of nineteen (19) and more mobile nodes. The following formulas were used for the fit curves in Figure 5.22:

For the fit curve labeled 1;

$$y = p1 * x + pn$$

$$Coefficients:$$

$$p1 = 1.2645, \qquad pn = 0$$

$$Norm\ of\ residuals = 0$$

Equation 5.1: Linear Fit Curve

For the fit curve labelled 2;

$$y = p1 * x^2 + p2 * x + pn$$

$$Coefficients:$$

$$p1 = 0.060251, \quad p2 = 0, \quad pn = 0$$

$$Norm\ of\ residuals = 0$$

Equation 5.2: Quadratic Fit Curve

For the fit curve labelled 3;

$$y = p1 * x^3 + p2 * x^2 + p3 * x + pn$$

$$Coefficients:$$

$$p1 = 0.0028707, \quad p2 = 0, \quad p3 = 0, \quad pn = 0$$

$$Norm\ of\ residuals = 0$$

Equation 5.3: Cubic Fit Curve

Where:

y is the level of connectivity represented on the y-axis.

x is the simulation time represented on the x-axis.

p is the particular position of the different nodes.

n is the unspecified number of the node in the network.

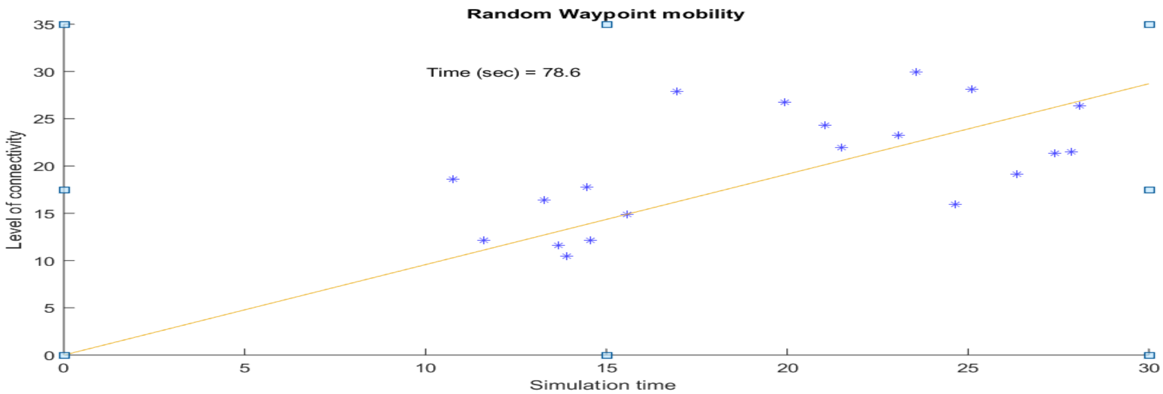

Figure 5 21: Ideal Fixed Network Connectivity

Figure 5.22: Effect of Mobile Nodes on the Level of Connectivity

5.5.6 System Testing Classes

Table 5.2: System Testing Classes

Test Class	**Difficulty**	**Comment**	**Test Results**
Functional	Medium	Connectivity between all the hardware components of the system was achieved.	Pass
Functional	High	All the system components were able to detect the commands being generated and forwarded.	Pass

Functional	High	Components' interconnections allowed the various algorithms to be analyzed and tested based on their characteristics.	Pass
Non-functional	Low	The interoperability of the hardware components and communication standards in mobile networks during system implementation was demonstrated.	Pass
Non-functional	Low	Integration and adherence to software standards were demonstrated during system implementation.	Pass

5.6 Challenges Faced in Implementation

To implement the research's objective, the researcher experienced some challenges that increased the timeline of the research implementation and it also affected the functionality of the system. The challenges the researcher came across areas discussed in the next sections starting from 5.6.1 through section 5.6.4 below.

5.6.1 Complexity

The Xbee radios are very fragile and very easy to get spoiled. The first Xbee radio failed as a result of exceeding the required voltage of 3.3v to 5.0v. Also failing to ground the Xbee results in the failure of the devices. The Xbee had to be recovered most of the time during the experiments since any change in their functionality resulted in the devices resetting their firmware. The support hardware in MatLab for the Arduino Uno led to a constant reset of the Xbee radios as the radios required the Arduino support for forwarding the packets. This resulted in the system crashing whenever the models were running simultaneously with the radios exchanging data and synchronization happening. This also resulted in the system being slow.

5.6.2 Conflicting Version of the Software Components

The researcher had proposed to used MatLab version R2015a, however, certain functionalities and features could be achieved with this version. There were some libraries for programming the Xbee radios this version did not have. The researcher eventually settled on working with MatLab version R2018a which had most of these libraries. Selecting the correct version of the X-CTU especially the one to configure the S2C module required trial and error as some version of the software tool lacks the list of all the firmware of the different functionalities an Xbee radio can have.

5.6.3 Devices Interconnectivity

There are different types of Xbees radios in which connecting the different radios is not that straight forward while others are incompatible. Xbee S1 and S2 are incompatible while S1 and S2C compatible but need some configurations. S2 and S2C are also compatible with some configurations. The researcher bought one S2C and two S1 Xbee types but one S1 Xbee got spoiled thus an option of working with an S2C and an S1.

5.6.4 Cost

All types of Xbee cost more than Ksh 3500 and therefore, getting around twenty (20) Xbee radios to generate the several mobility models were expensive. The researcher had to simulate the other devices to get adequate data for the generation of the models.

Chapter 6: Discussion

6.1 Introduction

This chapter brings the test results collected in chapter five into perspective. The discussion in this chapter examines how the research was able to implement the synchronization and connectedness through the simulation of a personal area network based on ZigBee IEEE 802.15.4 standard. It further sheds more light on the test results obtained by the research. Moreover, the chapter concludes the research and offers insights for future research.

6.2 System Accuracy

The system accuracy outlines the level at which the results of the research and particularly the built model met the expected reliability of the system in a real-world environment in terms of connectedness which is both synchronization and routing.

6.2.1 Synchronization Accuracy

Synchronization is one of the challenges in mobile wireless networks as the mobile nodes are expected to be coordinated together without affecting the throughput and communication in a network very much. This has not been so successful in mobile networks like WiFi, Zwave, and Bluetooth, however, certain techniques and algorithms have been considered more stable and better. In a ZigBee network, the tuning of beacon intervals, superframe duration, and time offsets are key parameters in achieving synchronization The beacon interval Bi, the superframe duration Sd, and the beacon time offset Bto (originate from coordinator node to end devices) can be configured in a ZigBee network especially for chain topologies. The recommended configurations are that the Bi > 3 and Bto and Sd < 2 (Giarré, Pesenti, & Tinnirello, 2010). These conditions, however, are affected by hidden nodes as they do not guarantee active connections in a generic topology. The hidden nodes would not be synchronized to other nodes immediately thus the network takes more time to update its routing table and topology. Consider, for example, the network topology shown in Figure 6.1 assuming node 1 is the coordinator. Node 1 has two hidden nodes (node 3 and node 6) and when it tries to transmit frames to nodes 3 and 6, the nodes would be idle (inactive). These hidden nodes would only transmit requests towards node 1 (coordinator) or to other devices only after when the other devices are not transmitting. The approach adds some complexities to the network.

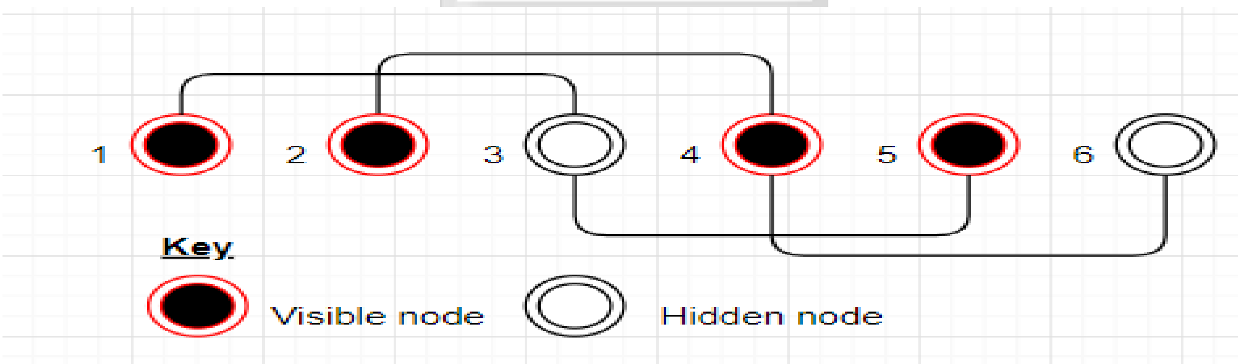

Figure 6.1: Decentralized Synchronization for Zigbee wireless sensor networks **(Tinnirello et al. 2010)**

This research, however, solved these complexities by ensuring that for every new node which most a times are hidden, the beacon intervals and the superframe duration are attached to the routing procure thus every time a routing table is updated, the hidden nodes are included in the network topology. This approach resulted in the outcomes shown in Figure 6.2.

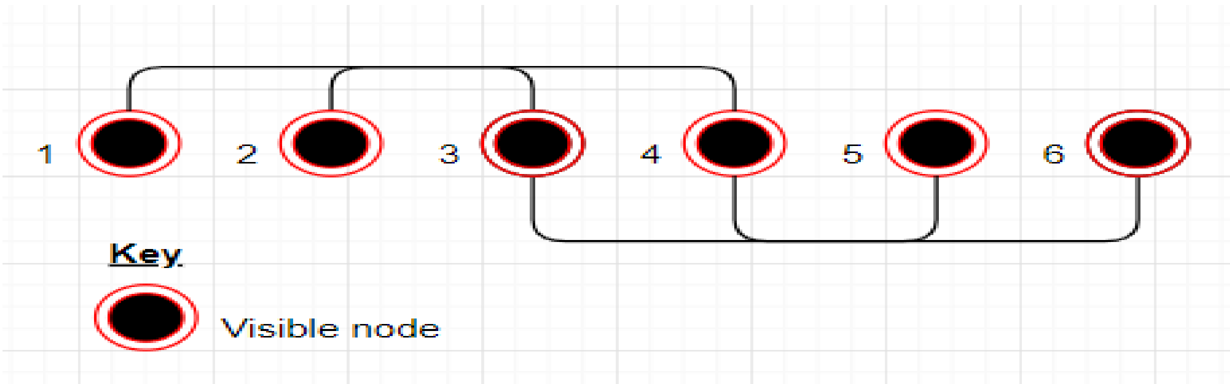

Figure 6.2: Hybrid Decentralized Synchronization for Zigbee Wireless Sensor Networks

6.2.2 Routing Accuracy

The reactive AODV routing protocol is the most used in mobile wireless networks such as the sensor networks due to its simplicity and it can effectively reduce routing overhead (Shabbir & Hassan, 2017). This research also implemented AODV as the routing protocol and coupled it with the synchronization algorithm to further reduce overheads and enhanced security. As a result, the hybrid of the algorithm proved to be more efficient because of the modulation of the signal done during the synchronization procedure. Also, normally if errors and variations arise in nominated links, the AODV protocol makes a fresh new route for the rest of communication and therefore, coupling it with synchronization makes the protocol more intelligent and fast in establishing new and fresh routes. In this research, the researcher used the Zigbee networks to illustrate the network topic of study and the parameters used were based on the XBee API frame type 0xA1 utilizing the frames labeled "route record".

6.3 Discussion of the Results

The topological design outlined by the diagrams in figure 4.3 and figure 4.4 and the results captured in Figures 5.16, 5.19, 5.20 and 5.22 are the results that answer the achievements of the research's objectives. Figure 4.3 illustrates the several messages exchange between the several components in the system to achieve synchronization in the whole network. Moreover, appendix B in the appendices chapter shows the final results of the setup. Figure 4.5 illustrates the synchronization process and how the association and data requests are achieved. Figure 5.19 and Figure 5.20 demonstrate how the design in Figures 4.3 and 4.5 was achieved. Furthermore, appendix A indicates the source code used to actualize the design mentn above. Figures (5.19 and 5.20) show the messages produced and exchanged during synchronization between the Zigbee coordinator and the two end devices. Figure 5.16 illustrates how the coordinator and an end device can communicate via a serial console and the conversion of the text messages to frames for transmission. The underlying routing protocol for forwarding such frames is the ad hoc on-demand distance routing protocol (AODV). The hardware devices chosen for the experiment played an important role in the synchronization and forwarding of frames in the personal area network. Any other microcontrollers apart from Arduino Uno could have achieved the same purpose though with some complexities and challenges. Arduino is highly compatible with Matlab and easy to use with the Xbee radios. This, therefore, was the basis of the researcher's decision to use the Arduino Uno microcontrollers with the Xbee modules and Matlab as the simulation tool.

Moreover, the mention tools allow simulation of more virtual Xbee modules with the same characteristics thus the researcher was able to simulate the random walk and random waypoint mobility models using the data generated. The models illustrate the mobility aspect of the Xbee modules in a specified network. The models are illustrated in Figures 5.17 and 5.18 in chapter 5. The Random Walk and Random Waypoint mobility models produced the same results for the same inputs. However, the difference in performance rose with the increase in pause time. The figures illustrate that the motion of the nodes, direction, and speed, as well as the angle of direction, are similar under both mobility models. The random walk assumed many connections compared to the random waypoint. However, the research proved that it is easier and better analyzing connectivity in a highly dynamic network using the Random Waypoint mobility model than the Random Walk mobility model. Random Walk mobility model shows how the number of mobile nodes affects connectivity in a network as illustrated by Figure 5.22. Appendix C gives a clear definition of the input parameters required to simulate Random Waypoint mobility model and appendix D illustrates the source code used to simulate the Random Waypoint mobility model.

Finally, the research has demonstrated that an awaking guarantee (synchronization and connectedness) in mobile wireless networks is possible. This is achievable as long as the mobile nodes go into sleep modes in an active network for very little time like 15 milliseconds or zero time. Routing protocol plays an important role in this purpose as well since efficient routing determines the transmission time of frames between the mobile nodes. The results, therefore, influence the level of connectedness.

6.4 Validity of the Proposed Solution

The results of the research illustrate that routing and synchronization can be done at the same time in the ZigBee network and other ad hoc networks such as sensor networks to improve low-level services and security for the dynamic nodes through a hybrid decentralized synchronization. A major problem in the current ad hoc networks is active connectedness complexities and therefore, the research provided an insight on how the issue can be addressed especially in the novel applications such as drones technology and driverless cars. Moreover, the research illustrates how the low power low data rates technology can be used to provide services in real-world applications and especially those which require few signaling and coordination services. This research also solves the complexities of distributed synchronization in a ZigBee wireless network as explain by Tinnirello et al., 2010. The research reduced the risks of beacons collisions by ensuring that the synchronization is enabled with association and data requests and responses thus limiting the number of beacons send while ensuring the devices remain active. Furthermore, the research illustrates how modulation in the channel is useful for the low power low data rates networks in achieving the best connectivity and low-level services provision.

Chapter 7: Conclusion and Recommendation

7.1 Conclusion

To improve low-level services such as communications and coordination for high mobile nodes in mobile wireless networks, improvement of the awaking guarantee (synchronization and connectedness) is very essential. The level of synchronization has a great impact on the connectivity among the mobile nodes and it also influences the choice of the routing procedure implemented. Certain applications of the mobile wireless network such as in autonomous cars and MANETS mean that the aim would be avoiding accidents. Therefore, the messages should not be delayed and all the nodes should be connected and the updates should be constant. All these require a high level of synchronization. In such kinds of applications, the synchronization mechanism has a direct effect on the choice of a routing algorithm. The capability of the nodes in a ZigBee technology to have zero sleep time mode, allows the technology to be used in highly critical applications. Certain applications are not very critical but can benefit sufficiently from this technology due to communication on low bandwidth and low power with improvements in low-level services provision through better synchronization and connectivity of the mobile nodes. Moreover, the technology is affordable compared to some of the low power low bandwidth technologies like LoRa thus a high potential of implementation to solve the current problems in society.

7.2 Recommendations

ZigBee technology being easy to learn, manage, customize and easily compatible with several standards like Zwave, Bluetooth, Thread, LoRa, 6LoWPAN, Sigfox, Near Field Communications (NFC) and WiFi, it has practical applications as far as the Internet of Things (IoT) is a concern. The aspect that it uses low bandwidth and low power in communication, it suits well with current developments such as signaling in autonomous cars to avoiding a collision or even in drones technology. The technology allows utilization of hardware especially microcontrollers from different vendors and the several simulations open source software thus reducing some component complexities. The research, however, recommends the following:

i. Information technology companies dealing with the internet of things applications should evaluate the several low bandwidth technologies like ZigBee and their potential use in the industry to solve the current challenges in society.
ii. Synchronization and routing in mobile networks are quite a challenge in the internet of things (IoT) and therefore, organizations can utilize the features in the ZigBee network to improve low-level services in the IoT environment.
iii. The mobility aspect in the mobile network can be utilized in real-world applications to solve the problems by mimicking mobility and using mobility solutions in networks to solve it.
iv. Synchronization in mobile networks plays a key role in services provision. Therefore, more research on synchronization algorithms and techniques is required for more improvements.
v. Embedded systems are one of the rapid developments in building smart cities and therefore, ZigBee being one of the best technologies with wide compatibility with other technologies, the organizations should embrace this purpose and support its development especially in aspects of software and hardware interoperability.

7.3 Suggestions for Future Research

The researcher managed to demonstrate how routing, connectedness, and synchronization in the mobile wireless networks can be achieved using the Zigbee based wireless network. The researcher focused on the synchronization and the connectedness aspect of the network to demonstrate awaking in mobile networks and has the suggestions for future work:

i. The researcher recommends using several (more than 10) physical Xbee modules to set up the Zigbee mobile wireless network. These Xbee modules could then be connected to an external powered source for high mobility.

Analysis of this network could then be carried out as the topology of the network change unpredictable.

ii. Moreover, the researcher suggests that future work involves using an algorithm that can provide both synchronizations and routing as a single function. The routing and synchronization by this algorithm should be efficient and effective thus eliminating any overheads.

iii. The researcher also suggests that in future work, a Zigbee network be incorporated into any other wireless communication standard like Z-Wave. Two or more technologies should communicate and be synchronized together seamlessly.

7.4 Contributions

Wireless and mobile communication networks are more practical now than before due to a wide range of applications. Therefore, there is a higher requirement for the rapid deployment of independent mobile entities and users such as in dynamic communications for emergency operations, disaster management, crime management, drones in agriculture and autonomous cars networks. All these communications systems are based on mobile ad-hoc networks that do not have a centralized control or management point. These networks are, therefore, characterized by a collection of mobile nodes that are in motion thus the topology of the entire network is unpredictable due to rapid changes over time. Therefore, routing and synchronization in these types of networks is the main issue due to the constant change of the infrastructure and paths. The communication of these nodes is over limited bandwidth and low energy consumption and the nodes themselves managed the network by forwarding the traffic since there is no special device or node for this purpose. This research on awaking guarantee (synchronization and connectedness) in mobile wireless networks contribute significantly in evaluating the best algorithm for routing which can automatically recognize any topological changes and limit the extra overhead of control messages in the low bandwidth communications and the best synchronization mechanisms with little overhead in these highly dynamic networks.

References

Alex. (2016). Google’s Self-Driving Car Caused its First Crash. Retrieved from Wired: https://www.wired.com/2016/02/googles-self-driving-car-may-caused-first-crash/

Aslam, M. Z., & Rashid, A. (2011). Comparison of random waypoint & random walk mobility model under DSR, AODV & DSDV MANET Routing Protocols. *arXiv preprint arXiv:1104.2368*.

Attia, R., Rizk, R., & Ali, H. A. (2015). Internet connectivity for mobile ad hoc network: a survey-based study. *Wireless networks*, *21*(7), 2369-2394.

Bandyopadhyay, S., Coyle, E. J., & Falck, T. (2007). Stochastic properties of mobility models in mobile ad hoc networks. *IEEE Transactions on Mobile Computing*, *6*(11), 1218-1229.

Bernard. (2018). *Key Milestones Of Waymo - Google's Self-Driving Cars.* New York: Forbes.

Cheng, X., Zhang, R., Chen, S., Li, J., Yang, L., & Zhang, H. (2018). 5G enabled vehicular communications and networking. *China Communications*, *15*(7), iii-vi.

Chung, J. M., & Go, D. C. (2012). Stochastic vector mobility model for mobile and vehicular ad hoc network simulation. *IEEE Transactions on Mobile Computing*, *11*(10), 1494-1507.

Cui, H., Liang, Y., Zhou, C., & Cao, N. (2018). Localization of Large-Scale Wireless Sensor Networks Using Niching Particle Swarm Optimization and Reliable Anchor Selection. *Wireless Communications and Mobile Computing*, *2018*.

Danda, B., Bhed, R., & Gongjun, B. (2015). Information Resources Management Association. *Mobile Computing and Wireless Networks: Concepts, Methodologies, Tools, and Applications*. IGI Global.

Emmanuel, T. & Sanders, J. (2013). *Routing in Mobile Networks.* Muizenberg: African Institute for Mathematical Sciences, South Africa.

Eyisi, D. (2016). The Usefulness of Qualitative and Quantitative Approaches and Methods in Researching Problem-Solving Ability in Science Education Curriculum. Journal of Education and Practice, 7(15), 91-100.

Foley & Lardner LLP. (2017). 2017 Connected Cars & Autonomous Vehicles Survey. Milwaukee: Foley.

Függer, M., Nowak, T., & Charron-Bost, B. (2015, March). Diffusive clock synchronization in highly dynamic networks. In *2015 49th Annual Conference on Information Sciences and Systems (CISS)* (pp. 1-6). IEEE.

Griffin, L. (2018). *Wireless & Mobile Networking.* Mountain View: Study.com.

Ibañez-Guzman, J., Laugier, C., Yoder, J. D., & Thrun, S. (2012). Autonomous driving: Context and state-of-the-art. *Handbook of Intelligent Vehicles*, 1271-1310.

IEEE 802.15.4-2011 - IEEE Standard for Local and metropolitan area networks--Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs).

Intellias. (2018). *How Machine Learning Algorithms Make Self-Driving Cars a Reality*. Intellias - Intelligent Software Engineering. Berlin: Intellias.

Java T Point. (2018). *Mobile Communications and Routing.* Noida: Java T Point.

Joarder. (2011). *A Comprehensive Study on Multi-hop Ad hoc Networking and Applications: MANET and VANET.* Stafford: Staffordshire University. Retrieved from SlideShare: https://www.slideshare.net/kamal005/a-comprehensive-study-on-multihop-ad-hoc-networking-and-applications-manet-and-vanet

Karen. (2019). *Self-driving cars may be more likely to hit you if you have dark skin.* Massachusetts: MIT Technology Review.

Kimberlin, C. L., & Winterstein, A. G. (2008). Validity and reliability of measurement instruments used in research. American journal of health-system pharmacy, 65(23), 2276-2284.

Kursus. (2016). *WIFI Technology.* Johor Bahru: University Technology Malaysia.

Li, Y., & Bartos, R. (2019). Connectedness-aware copy-adaptive routing protocol in intermittently connected networks. *International Journal of Wireless Information Networks*, *26*(3), 230-242.

Lin, X., Wiren, R., Euler, S., Sadam, A., Maattanen, H. L., Muruganathan, S., & Yajnanarayana, V. (2019). Mobile Network-Connected Drones: Field Trials, Simulations, and Design Insights. *IEEE Vehicular Technology Magazine*, *14*(3), 115-125.

Marzak, B., Toumi, H., El Guemmat, K., Benlahmar, A., & Talea, M. (2016). A survey on routing protocols for vehicular ad-hoc networks. *Indian Journal of Science and Technology*, *9*(S1).

Marzak, B., Toumi, H., Talea, M., & Benlahmar, E. (2015, June). Cluster head selection algorithm in vehicular Ad Hoc networks. In *2015 International Conference on Cloud Technologies and Applications (CloudTech)* (pp. 1-4). IEEE.

Michael. (2018, December). Nine months after deadly crash, Uber is testing self-driving cars again in Pittsburgh. Retrieved from the Washing Post: https://www.washingtonpost.com/transportation/2018/12/20/nine-months-after-deadly-crash-uber-is-testing-self-driving-cars-again-pittsburgh-starting today/?noredirect=on&utm_term=.cdc3a2f7ad56

Morales, M. M. C. (2011). An Adaptable Mobility-Aware Clustering Algorithm in vehicular networks. 13th Asia-Pacific Network Operations and Management Symposium (APNOMS). pp. 1- 6.

Nath, S. K., Aznabi, S., Islam, N. T., Faridi, A., & Qarony, W. (2017). Investigation and performance analysis of some implemented features of the ZigBee protocol and IEEE 802.15. 4 Mac specification. *International Journal of Online and Biomedical Engineering (iJOE)*, *13*(01), 14-32.

Pawlik, A., Bonfort, T., John, N., Karl, S., Klein, T., Oser, L., & Abthoff, T. (2018). Big data for assisted and autonomous driving. In *18. Internationales Stuttgarter Symposium* (pp. 467-475). Springer Vieweg, Wiesbaden.

Pfeuffer, K., & Li, Y. (2018, April). Analysis and Modelling of Grid Performance on Touchscreen Mobile Devices. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (p. 288). ACM.

Pramanik, A., Choudhury, B., Choudhury, T. S., Arif, W., & Mehedi, J. (2015, April). Simulative study of random waypoint mobility model for mobile ad hoc networks. In *2015 Global Conference on Communication Technologies (GCCT)* (pp. 112-116). IEEE.

Pype, P., Daalderop, G., Schulz-Kamm, E., Walters, E., & von Grafenstein, M. (2017). Privacy and security in autonomous vehicles. In *Automated Driving* (pp. 17-27). Springer, Cham.

Qiu, B. (2018). Design of Mobile 4G Gateway Based on ZigBee Wireless Sensor Network. *International Journal of Online Engineering (iJOE)*, *14*(11), 117-132.

Rabbit Technologies. (2016). *An Introduction to Wi-Fi.* USA: Digi International Inc.

Rajasekhar, M. V., & Jaswal, A. K. (2015, August). Autonomous vehicles: the future of automobiles. In *2015 IEEE International Transportation Electrification Conference (ITEC)* (pp. 1-6). IEEE.

Ramasamy, V. (2017). Mobile Wireless Sensor Networks: An Overview. *Wireless Sensor Networks: Insights and Innovations*, 1.

Santi, P., & Blough, D. M. (2002). An evaluation of connectivity in mobile wireless ad hoc networks. In *Proceedings International Conference on Dependable Systems and Networks* (pp. 89-98). IEEE.

Singh, V., & Dadhich, R. (2015). Efficient routing by minimizing end to end delay in delay tolerant enabled VANETs. *International Bulletin of Mathematical Research*, *2*(1), 241-5.

Shabbir, N., & Hassan, S. R. (2017). Routing protocols for wireless sensor networks (WSNs). In *Wireless Sensor Networks-Insights and Innovations*. IntechOpen.

Shah, S. A. A., Ahmed, E., Imran, M., & Zeadally, S. (2018). 5G for vehicular communications. *IEEE Communications Magazine*, *56*(1), 111-117.

Swathi, G., & Saravanan, R. (2014). Secure routing in MANET using synchronization and data authenticity. *Cybernetics and Information Technologies*, *14*(2), 24-30.

Threlfall, R. (2019). KPMG International. 2019 Autonomous Vehicles Readiness Index. *Assessing countries' preparedness for autonomous vehicles*: Retrieved from KPMG International website:https://assets.kpmg/content/dam/kpmg/xx/pdf/2019/02/2019-autonomous-vehicles-readiness-index.pdf

Tinnirello, I., Giarré, L., & Pesenti, R. (2010). Decentralized synchronization for ZigBee wireless sensor networks in Multi-Hop Topology. *IFAC Proceedings Volumes*, *43*(19), 257-262.

Vasseur, J. P., & Dunkels, A. (2010). *Interconnecting smart objects with IP: The next internet*. Morgan Kaufmann.

Wen, X. X., & Lin, J. Y. (2016). Time synchronization algorithm based on the mobility model for underwater sensor networks. *Journal on Communications*, *37*(1), 2016001.

Zigbee Wireless Mesh Networking. (2019, January). Retrieved from Digi.com: https://www.digi.com/resources/standards-and-technologies/zigbee-wireless-standard

Appendices

Appendix A: Source Code for XBee Devices Synchronization and Connection

Editor - C:\Users\Tots\Documents\MATLAB\Xbee_devices.m

test_Animate.m | Xbee_devices.m | Untitled*

```matlab
symbolsPerStep = 20;
chipsPerSymbol = 32;
samplesPerChip = 4;
symbolRate = 65.5e3; % symbols/sec
time = 0;
stopTime = 5; % sec

% Create PAN Coordinator
panCoordinator = lrwpan.MACFullFunctionDevice('PANCoordinator', true, 'SamplesPerChip', 4, ....
  'PANIdentifier', '1234', 'ExtendedAddress', [repmat('0', 1, 8) repmat('7', 1, 8)], ...
  'ShortAddress', '1234');

% Create first end-device:
endDevice1 = lrwpan.MACReducedFunctionDevice('SamplesPerChip', 4, ...
  'ShortAddress', 'FFFE', 'ExtendedAddress', [repmat('0', 1, 8) repmat('3', 1, 8)]);

% Initialize device inputs
received1 = zeros(samplesPerChip * chipsPerSymbol * symbolsPerStep/2, 1);
received2 = zeros(samplesPerChip * chipsPerSymbol * symbolsPerStep/2, 1);

while time < stopTime
  % Pass the received signals to the nodes for processing. Also, fetch what
  % they have to transmit:
  transmitted1 = panCoordinator(received1);
  transmitted2 = endDevice1(received2);

  % Ideal wireless channel, where both nodes are within range:
  received1 = transmitted2; % half-duplex radios, none receiving while transmitting
  received2 = transmitted1;

  time = time + symbolsPerStep/symbolRate; % update clock
end
% Create second end-device:
endDevice2 = lrwpan.MACReducedFunctionDevice('SamplesPerChip', 4, ...
  'ShortAddress', '0002', 'ExtendedAddress', [repmat('0', 1, 8) repmat('4', 1, 8)], 'Verbosity', false);
% Suppress detailed output:
endDevice1.Verbosity = false;
panCoordinator.Verbosity = false;

% Initialize input
received3 = zeros(samplesPerChip * chipsPerSymbol * symbolsPerStep/2, 1);

stopTime = 10; % sec
while time < stopTime
  % Pass the received signals to the nodes for processing. Also, fetch what
  % they have to transmit:.,.
  transmitted1 = panCoordinator(received1);
  transmitted2 = endDevice1(received2);
  transmitted3 = endDevice2(received3);

  % Ideal wireless channel, where all nodes are within range:
  received1 = transmitted2 + transmitted3; % half-duplex radios, none receiving while transmitting
  received2 = transmitted1 + transmitted3;
  received3 = transmitted1 + transmitted2;

  time = time + symbolsPerStep/symbolRate; % update clock
end
```

Appendix B: Synchronization and Connectivity Results in XBee Devices

Command Window

```
>> Xbee_devices
FFFE: ********* Adding Beacon Request frame to the queue
FFFE: Passive scanning for 1584 steps
FFFE: Processing next frame from the queue
FFFE: Initializing transmission; backoff delay = 1 steps
FFFE: Backoff delay = 1 steps -> 0 steps
FFFE: Carrier sensing: Medium is idle.
FFFE: Clear to transmit
FFFE: Transmitting Beacon Request
FFFE: IFS offset = 0 samples
FFFE: Transmitting 1-1280 of 2050
Found preamble of OQPSK PHY.
Found start-of-frame delimiter (SFD) of OQPSK PHY.
FFFE: IFS offset = 0 samples
FFFE: Transmitting 1281-2050 of 2050
FFFE: Finished transmission
FFFE: Need to wait for SIFS (12) symbols. Offset = 12, next IFS = 4
FFFE: Entering passive scanning
1234: PHY decoded IEEE 802.15.4 frame
1234: Need to wait for SIFS (12) symbols. Offset = 12, next IFS = 4
CRC check passed for the MAC frame.
1234: *********** Received frame type = MAC command
1234: *********** Received MAC Command type = Beacon request
1234: ********* Adding Beacon frame to the queue
1234: next IFS = 4
1234: Processing next frame from the queue
1234: Initializing transmission; backoff delay = 7 steps
1234: Backoff delay = 7 steps -> 6 steps
1234: Backoff delay = 6 steps -> 5 steps
1234: Backoff delay = 5 steps -> 4 steps
```

```
1234: Backoff delay = 4 steps -> 3 steps
1234: Backoff delay = 3 steps -> 2 steps
1234: Backoff delay = 2 steps -> 1 steps
1234: Backoff delay = 1 steps -> 0 steps
1234: Carrier sensing: Medium is idle.
1234: Clear to transmit
1234: IFS offset = 256 samples
1234: Transmitting 1-1024 of 2562
1234: IFS offset = 0 samples
1234: Transmitting 1025-2304 of 2562
Found preamble of OQPSK PHY.
Found start-of-frame delimiter (SFD) of OQPSK PHY.
1234: IFS offset = 0 samples
1234: Transmitting 2305-2562 of 2562
1234: Finished transmission
1234: Need to wait for LIFS (40) symbols. Offset = 4, next IFS = 24
1234: Decreased wait time by 20 symbols to 4
FFFE: PHY decoded IEEE 802.15.4 frame
FFFE: Need to wait for SIFS (12) symbols. Offset = 4, next IFS = -4
CRC check passed for the MAC frame.
FFFE: *********** Received frame type = Beacon
FFFE: next IFS = 0
FFFE: Scanning finished
FFFE: ********* Adding Association request frame to the queue
FFFE: Processing next frame from the queue
FFFE: Initializing transmission; backoff delay = 0 steps
FFFE: Carrier sensing: Medium is idle.
FFFE: Clear to transmit
FFFE: IFS offset = 0 samples
FFFE: Transmitting 1-1280 of 3458
```

```
1234: IFS offset = 0 samples
1234: Transmitting 6401-7680 of 8578
1234: IFS offset = 0 samples
1234: Transmitting 7681-8578 of 8578
1234: Finished transmission
1234: Need to wait for LIFS (40) symbols. Offset = 14, next IFS = 34
1234: will wait for ack for 54 symbols additional to IFS = 34
1234: Decreasing ack wait durations by 20 symbols to 68
8CEC: PHY decoded IEEE 802.15.4 frame
8CEC: Need to wait for LIFS (40) symbols. Offset = 14, next IFS = 34
CRC check passed for the MAC frame.
8CEC: *********** Received frame type = Data
8CEC: *********** Directly transmitting acknowledgement frame (no CSMA/CA)
8CEC: next IFS = 34
1234: Decreasing ack wait durations by 20 symbols to 48
8CEC: IFS offset = 896 samples
8CEC: Transmitting 1-384 of 1410
1234: Decreasing ack wait durations by 20 symbols to 28
8CEC: IFS offset = 0 samples
8CEC: Transmitting 385-1410 of 1410
8CEC: Finished transmission
8CEC: Need to wait for SIFS (12) symbols. Offset = 16, next IFS = 8
Found preamble of OQPSK PHY.
Found start-of-frame delimiter (SFD) of OQPSK PHY.
1234: PHY decoded IEEE 802.15.4 frame
1234: Need to wait for SIFS (12) symbols. Offset = 16, next IFS = 8
CRC check passed for the MAC frame.
1234: *********** Received frame type = Acknowledgment
1234: next IFS = 8
```

Appendix C: Source Code for the Random Waypoint Mobility Model Input Parameters

Editor - C:\Users\Tots\Documents\MATLAB\Random_Waypoint\test_Execute.m

test_Animate.m | Xbee_devices.m | test_Execute.m | +

```
%Testing Random Waypoint mobility model.
clear all;clc;close all;

s_input = struct('V_POSITION_X_INTERVAL',[10 30],...%(m)
                 'V_POSITION_Y_INTERVAL',[10 30],...%(m)
                 'V_SPEED_INTERVAL',[0.2 2.2],...%(m/s)
                 'V_PAUSE_INTERVAL',[0 1],...%pause time (s)
                 'V_WALK_INTERVAL',[2.00 6.00],...%walk time (s)
                 'V_DIRECTION_INTERVAL',[-180 180],...%(degrees)
                 'SIMULATION_TIME',500,...%(s)
                 'NB_NODES',20);
s_mobility = Generate_Mobility(s_input);

timeStep = 0.1;%(s)
test_Animate(s_mobility,s_input,timeStep);
```

Appendix D: Source Code for the Animation of the Random Waypoint Mobility Model

Editor - C:\Users\Tots\Documents\MATLAB\Random_Waypoint\test_Animate.m

test_Animate.m | Xbee_devices.m | test_Execute.m

```matlab
function test_Animate(s_mobility,s_input,time_step)

    v_t = 0:time_step:s_input.SIMULATION_TIME;

    for nodeIndex = 1:s_mobility.NB_NODES
        %Simple interpolation (linear) to get the position, anytime.
        %Remember that "interp1" is the matlab function to use in order to
        %get nodes' position at any continuous time.
        vs_node(nodeIndex).v_x = interp1(s_mobility.VS_NODE(nodeIndex).V_TIME,s_mobility.VS_NODE(nodeIndex).V_POSITION_X,v_t);
        vs_node(nodeIndex).v_y = interp1(s_mobility.VS_NODE(nodeIndex).V_TIME,s_mobility.VS_NODE(nodeIndex).V_POSITION_Y,v_t);
    end
    figure;
    hold on;
    for nodeIndex = 1:s_mobility.NB_NODES
        vh_node_pos(nodeIndex) = plot(vs_node(nodeIndex).v_x(1),vs_node(nodeIndex).v_y(1),'*','color',[0.3 0.3 1]);
    end
    title(cat(2,'Simulation time (sec): ',num2str(s_mobility.SIMULATION_TIME)));
    xlabel('X (meters)');
    ylabel('Y (meters)');
    title('Random Waypoint mobility');
    ht = text(min(vs_node(1).v_x),max(vs_node(1).v_y),cat(2,'Time (sec) = 0'));
    axis([min(vs_node(1).v_x) max(vs_node(1).v_x) min(vs_node(1).v_y) max(vs_node(1).v_y)]);
    hold off;
    for timeIndex = 1:length(v_t);
        t = v_t(timeIndex);
        set(ht,'String',cat(2,'Time (sec) = ',num2str(t,4)));
        for nodeIndex = 1:s_mobility.NB_NODES
            set(vh_node_pos(nodeIndex),'XData',vs_node(nodeIndex).v_x(timeIndex),'YData',vs_node(nodeIndex).v_y(timeIndex));
        end
        drawnow;
```

Appendix E: Ethical Review Report from Strathmore University – Institutional Ethics Review Committee (SU-IERC)

Strathmore
UNIVERSITY

31st October 2019

Mr Lonyeiye Bruce,
bruce.totona@strathmore.edu

Dear Mr Lonyeiye,

RE: Awaking Guarantee in Mobile Wireless Networks for Autonomous Vehicles Based on Stochastic Mobility Model.

This is to inform you that SU-IERC has reviewed and **approved** your above research proposal. Your application approval number is **SU-IERC0560/19**. The approval period is **31st October, 2019 to 30th October, 2020.**

This approval is subject to compliance with the following requirements:

i. Only approved documents including (informed consents, study instruments, MTA) will be used
ii. All changes including (amendments, deviations, and violations) are submitted for review and approval by SU-IERC.
iii. Death and life threatening problems and serious adverse events or unexpected adverse events whether related or unrelated to the study must be reported to SU-IERC within 72 hours of notification
iv. Any changes, anticipated or otherwise that may increase the risks or affected safety or welfare of study participants and others or affect the integrity of the research must be reported to SU-IERC within 72 hours
v. Clearance for export of biological specimens must be obtained from relevant institutions.
vi. Submission of a request for renewal of approval at least 60 days prior to expiry of the approval period. Attach a comprehensive progress report to support the renewal.
vii. Submission of an executive summary report within 90 days upon completion of the study to SU-IERC.

Prior to commencing your study, you will be expected to obtain a research license from National Commission for Science, Technology and Innovation (NACOSTI) https://oris.nacosti.go.ke and also obtain other clearances needed.

Yours sincerely,

for: Dr Virginia Gichuru,
Secretary; SU-IERC

Cc: Prof Fred Were,
Chairperson; SU-IERC

STRATHMORE UNIVERSITY INSTITUTIONAL ETHICS REVIEW COMMITTEE (SU-IERC)
31 OCT 2019
TEL: +254 (0)703 034 000
P. O. Box 59857 - 00200
NAIROBI - KENYA

Ole Sangale Rd, Madaraka Estate. PO Box 59857-00200, Nairobi, Kenya. Tel +254 (0)703 034000
Email info@strathmore.edu www.strathmore.edu

Appendix F: Originality Report