Multi-Interface Cognitive Radio for Enhanced Routing Performance in Multi-Hop Cellular Networks

Ginu Thomas

A thesis submitted to

Auckland University of Technology

in partial fulfilment of the requirements for the degree

of

Master of Engineering (ME)

May 2012

School of Engineering

Primary Supervisor: Dr Boon-Chong Seet

Acknowledgements

Firstly, I would like to acknowledge and extend my heartfelt gratitude to my supervisor Dr Boon-Chong Seet for his supervision, advice, and guidance from the very early stage of this research as well as for giving me extraordinary experiences throughout the work. Above all and the most needed, he provided me unflinching encouragement and support in various ways. I could never have embarked upon and started all of this without his prior teachings which opened up unknown areas to me.

I would like to thank Dr. John Collins for his advice and guidance during the initial phase of this research. I am grateful in every possible way and hope to continue our collaboration in the future.

Many thanks go, in particular, to Mirza Mansoor Baig, PhD Student, for his timely support, guidance, and for providing technical expertise, and to Noreen Imran, PhD Student for her valuable advice and her precious times in helping to resolve some critical issues of this research.

I would also like to thank the OPNET Technical Support for their excellent support and suggestions throughout my research work.

Lastly, and most importantly, I wish to express my love and gratitude to my beloved parents and to my siblings who gave their moral support and care regardless of the distance apart.

Abstract

Multi-hop Cellular Network (MCN) is an emerging wireless communication architecture, which combines the benefits of conventional Single-Hop Cellular Networks (SCN) and Multi-hop Ad-hoc Relaying Networks. In MCN, communication can be either between a mobile station (MS) and the base station (BS) through intermediate relay stations (RS) over multiple hops or the MS can communicate directly with other MS through intermediate RSs over multiple hops without the involvement of a BS. Since all these stations are mobile, multi-hop routing in MCN is a major issue. The route selection depends on the availability of intermediate nodes, their neighbourhood connectivity and the channel availability.

Cognitive Radio (CR) is an emerging communication paradigm which exploits the available radio frequencies opportunistically for the effective utilisation of the radio frequency spectrum. Currently, CR is an enabling technology for users of non-licensed spectrum (e.g. ISM band) to sense and utilise unused radio frequencies in the licensed spectrum (e.g. TV white space) in a dynamic and non-interfering way.

The incorporation of CR and Mobile Ad-Hoc Network (MANET) routing protocols in MCN could potentially improve the spectrum utilisation and the routing performance of MCN. This research deals with the development and investigation of a multi-interface CR mobile node model in Optimized Network Engineering Tool (OPNET) to enhance the routing performance in MCN. The interfaces used in this CR mobile node are IEEE 802.11a (Wi-Fi), IEEE 802.11af (White-Fi) and IEEE 802.16 (WiMax). The 802.11a and 802.11af interfaces are used for multi-hop communication while WiMax interface is used for single-hop communication with BS when multi-hop communication is not possible with 802.11a and 802.11af interfaces. The 802.11af interface uses the CR feature to perform a channel selection. The multi-hop routing protocol used in this research is a modified version of AODV (Ad-hoc On-demand Distance Vector). The route selection by this routing protocol is not based on the minimum hops as in conventional AODV but based on the output interface available (802.11a or 802.11af) when multi-hop communication is possible. This research firstly develops the OPNET model for the multi-interface CR mobile and then investigates their opportunistic spectrum utilisation and routing performance in MCN.

Table of Contents

List of Figures

List of Tables

Statement of Originality

'I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been submitted for the award of any other degree or diploma of a university or other institution of higher learning. It contains results of my investigation, except where otherwise stated. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.'

I understand that my thesis may be made electronically available to the public.

Signed: Ginu Thomas ……………………

Date: 30 April 2012 …………………….

List of Abbreviations

ACK	Acknowledgement
AODV	Ad-hoc On-demand Distance Vector
API	Application Program Interface
BS	Base Station
CBR	Constant Bit Rate
CCC	Common Control Channel
CG	Cognitive Gateway
CR	Cognitive Radio
DSA	Dynamic Spectrum Access
DSDV	Destination Sequenced Distance Vector
DSR	Dynamic Source Routing
FCC	Federal Communications Commission
FSM	Finite State Machine
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
ISM	Industrial, Scientific and Medical (band)
MAC	Medium Access Control
MANET	Mobile Ad-hoc Network
MCN	Multi-Hop Cellular Network
MCRN	Multi-Hop Cognitive Radio Network
MPR	Multi-Point Relay
MS	Mobile Station
OLSR	Optimised Link State Routing
OPNET	Optimised Network Engineering Tool
PDR	Packet Delivery Ratio
PHY	Physical (layer)
PU	Primary User
QoS	Quality of Service
RERR	Route Error
RREP	Route Reply

RREQ	Route Request
RS	Relaying Station
SCN	Single-Hop Cellular Network
SDR	Software Defined Radio
SNR	Signal-to-Noise Ratio
SS	Subscriber Station
STD	State Transition Diagram
SU	Secondary User
TORA	Temporally Ordered Routing Algorithm
UDP	User Datagram Protocol
UHF	Ultra-High Frequency
UWB	Ultra-Wide Band
WCETT	Weighted Cumulative Expected Transmission Time
WiMax	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network

CHAPTER 1 Introduction

1.1 Background

The past decade has witnessed a significant growth in mobile communications. Today, the use of mobile communications has expanded from simple voice applications to streaming multimedia applications (such as video conferencing, mobile gaming) and remote accessing of large information databases [1]. These multimedia applications require a higher data rate and also a higher Quality of Service (QoS). The current cellular network, also known as the Single-Hop Cellular Network (SCN) is facing a great challenge in meeting both higher data rate demands and QoS in order to support these future mobile users' applications [2]. In SCN, the mobile station (MS) is directly communicating with the nearest base station (BS) through a single hop. This conventional cellular network is adequate for simple voice applications but requires increased network capacity to support high data rate multimedia applications. Since BS coverage typically reduces with increasing data rate [2], either the transmission power of the BS should be increased or more BSs should be installed to maintain coverage and support this high data rate demand. However, increasing transmission power can lead to greater co-channel interference which reduces the frequency reuse [2], while installing more BSs may not be practical due to their potential high installation cost.

Another important paradigm in mobile communication is the Multi-Hop Ad-Hoc Relaying Network [2]. These networks are infrastructure-less and can be easily deployed without any site planning. Multi-hop ad-hoc relaying network consists of mainly mobile users which can act as source, destination and intermediate relay stations. These intermediate relay stations which lie in the path of the source and the destination stations, forward the data packets in a hop-by-hop manner from the source to destinations. These multi-hop ad-hoc relaying networks are similar to Mobile Ad-hoc Networks (MANETs) [2] which operate on short-range wireless communication technologies (802.11, UWB, etc) and are capable of supporting high data rate multimedia applications.

1.2 Multi-Hop Cellular Networks (MCN)

MCN is a wireless communication architecture originally proposed by Lin and Hsu [4]. MCN combines the benefits of conventional SCN and Multi-hop Ad-hoc Relaying Networks. In conventional SCN, the source node reaches its destination node with a single hop through the nearest BS. For example as illustrated in Figure 1.2-1, if MS *A* wants to communicate with MS *B* and both stations are within the same cell *i, A* forwards the data to its nearest BS through a single-hop, which in turn forwards this data to *B* through another single-hop. In another example, if MS *C* wants to communicate with MS *D* and *C* is located within cell *i* and *D* is located within cell *j, C* forwards the data to its nearest BS located in cell *i* through a single hop, which in turn forwards this data to another BS located in cell *j* through a wired backbone. Finally, BS *j* forwards the data to *D* through another single hop. On the other hand, in MCN, communication can be either between a MS and the BS through intermediate RSs over multiple hops or the source MS communicates directly with its destination MS through intermediate RSs over multiple hops without the involvement of BS.

The intermediate RSs which lie in the path of the source and the destination MSs, forwards data packets towards the destination in a hop-by-hop manner. For example as illustrated in Figure 1.2-2, if MS *A* wants to communicate with MS *B* and both stations are within the same cell *i*. *A* may forward the data directly to *B* without the involvement of BS. As another example, if BS *j* wants to communicate with MS *D,* BS *j* forwards the data to an intermediate RS *E*, which then forwards it to MS *D*. The intermediate RSs can be either fixed network elements deployed by the service provider, known as Fixed Relays, or the MSs within the network, known as Mobile Relays or a combination of both Fixed and Mobile Relays, known as Hybrid Relays.

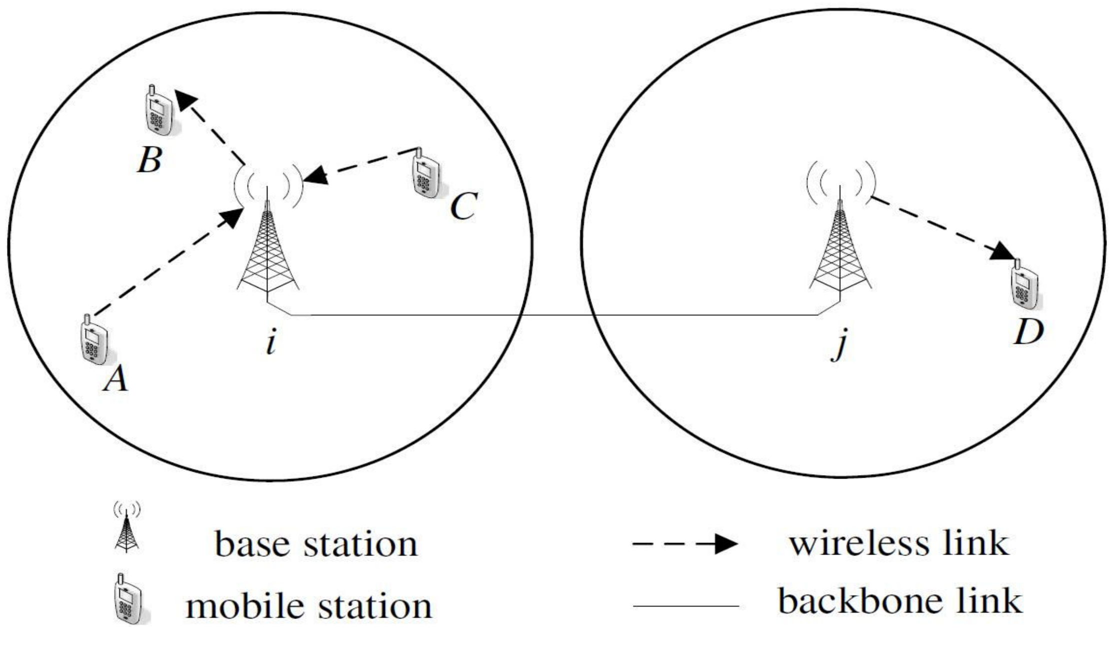


Figure 1.2-1: Single-hop Cellular Networks (SCNs) [2].

The fundamental idea behind the MCN is to break the long distance communication link into shorter links. This breaking of long distance link into shorter links reduces the required transmission power. This reduced transmission power in turn reduces the co-channel interference level and thus leads to greater frequency reuse. The shorter communication links in MCN also allows using a short-range and high data rate wireless communication technologies such as 802.11 and UWB, which provides high data transmission rate while also reducing the traffic load in the cellular network [2].

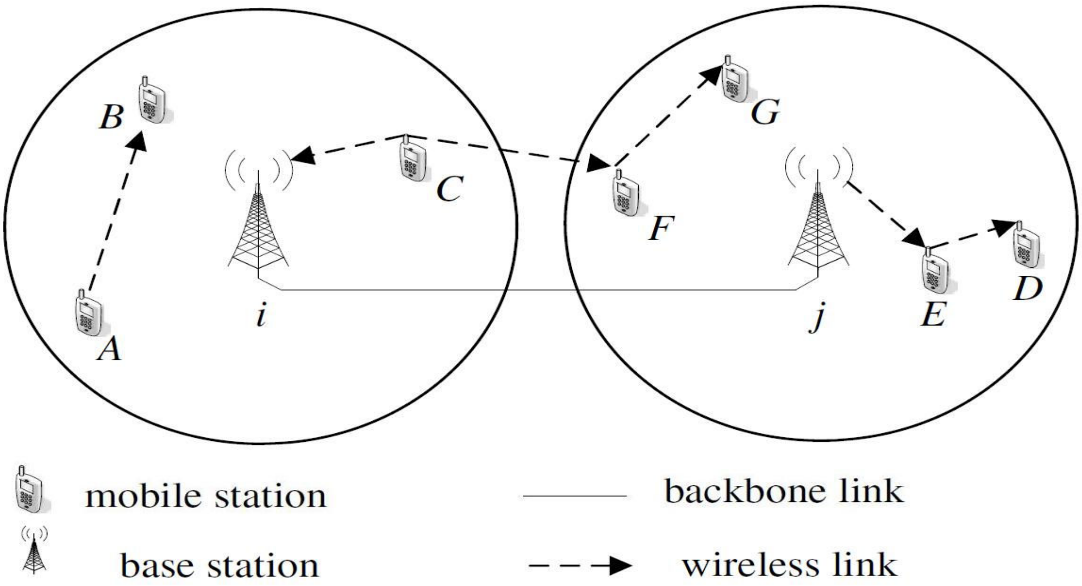


Figure 1.2-2: Multi-hop Cellular Networks (MCNs) [2].

The significant advantages of MCN over conventional SCN are described as follows:

i. **Transmission power:** In SCN, the communication between a MS and BS occurs over a single long hop. The required transmission power of the BS should be high enough to reach the MS. However, in MCN, the communication between a MS and BS is through intermediate RSs over multiple short hops. Thus, the required power per transmission can be reduced.

ii. **System capacity:** In SCN, since the required transmission power is much higher, the coverage of the BS will also be larger, which limits channel reuse. However, in MCN, since the required transmission power is lower, the BS coverage will also be smaller and thus offers a greater opportunity for channel reuse. This increases the system capacity in terms of the number of users that can be supported without increasing the number of unique frequency channels.

iii. **Data transmission rate:** Compared to MCN, the data transmission rate in SCN can deteriorate more severely as the single-hop distance between the MS and BS increases. In MCN, this deterioration is limited by the shorter hop distance between the stations. The shorter distance also allows the usage of short range and high data rate wireless technologies such as 802.11 and UWB in MCN.

iv. **Area coverage:** In SCN, the MS located in dead spots (regions near the cell border or inside a tunnel) will not be able to receive the BS signals. In MCN, this issue can be avoided by relaying the data around the dead spots over multiple hops.

The potential drawbacks of MCN are described as follows:

i. **System complexity:** Since MCN is a combination of SCN and MANETs, it inherits known issues from both types of systems, including handover, routing and resource management for peer to peer communication in MCN. This may cause the solutions required to be more complex than those for SCN or MANET itself.

ii. **Security:** Data transmission in MCN is through intermediate RSs over multiple hops in an open environment, and this may weaken the security of the system when the relay channels are in the ISM band, which can be freely accessed by anyone.

iii. **Delay:** Due to data transmission over multiple hops, the packets may be buffered at intermediate RSs when the nodes and/or channel are busy. This may result in higher end-to-end delay in MCN as compared to that in SCN.

iv. **User cooperation:** In cases where MSs are also serving as RSs, an incentive mechanism may be required to encourage cooperative relaying between individual MSs, which may further increase the system complexity.

1.3 Cognitive Radios (CR)

The increasing demand for new wireless communication applications has caused an increasing scarcity of available radio spectrum. Today's spectrum allocations are regulated mainly by governmental agencies based on fixed spectrum assignment policy [5]. Most of the prime radio frequency spectrum has been exclusively assigned for fixed licensed users on a long term basis. Due to this fixed frequency spectrum allocation, it is becoming difficult to find vacant radio frequency spectrum to allocate for these new wireless communication applications. However, recent study shows that spectrum scarcity is not due to the spectrum shortage but due to the inefficient fixed frequency spectrum allocations [6]. It also shows that a large portion of this allocated spectrum for licensed users are in low utilisation for most of the time. Figure 1.3-1 illustrates that the spectrum usage is heavy on certain portions of the radio frequency spectrum while a large portion remains unutilised. According to Federal Communications Commission (FCC) [5], the temporal and geographical utilisation variation of the fixed assigned spectrum ranges from 15% to 85%. For 90% of the time, certain parts of the licensed bands are unoccupied. This fixed allocation of frequency spectrum limits the flexibility of reusing any unused licensed frequencies, resulting in many spectrum holes known as “white spaces”.

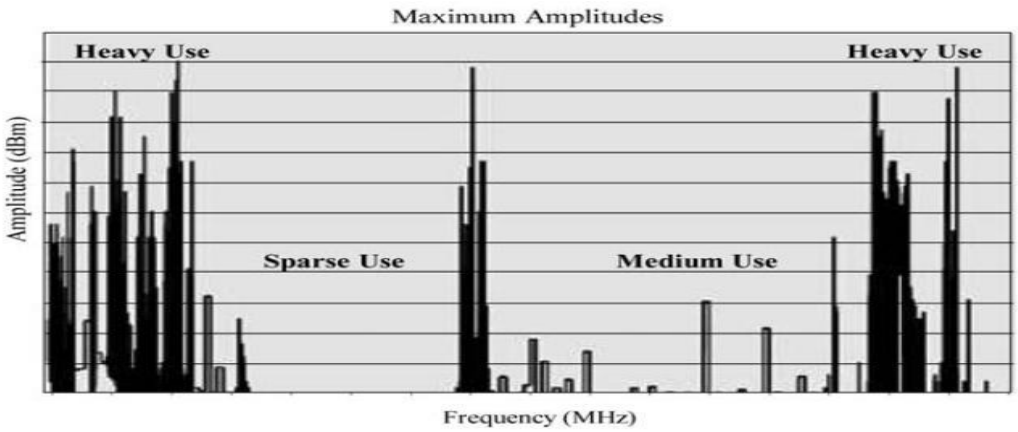


Figure 1.3-1: Spectrum utilisation graph [5].

The inefficiency of fixed spectrum allocation motivates a new communication paradigm that exploits such spectrum holes in licensed bands for opportunistic use by unlicensed users. Dynamic Spectrum Access (DSA) [6] is a new concept capable of solving the current fixed spectrum allocation inefficiencies. DSA is a technique in which unlicensed (secondary) users are allowed to use the unused licensed (primary) frequency bands opportunistically without causing any interference to the licensed users. For an efficient DSA operation, the unlicensed users should be able to determine the available frequency spectrum which is not being used by any licensed user in order to avoid interferences. The key technology which enables the unlicensed users to perform DSA operation is the Cognitive Radio technology.

CR is a communication paradigm originally defined by Mitola [5], [7], [8], [9] as an adaptive, multi-dimensionally aware, autonomous radio system that enables a network to use frequency spectrum in a dynamic manner. The CR, built on Software Defined Radio (SDR) [6] has the capability to sense its spectrum environment, tracks changes in spectrum usage, and reconfigures its air interface accordingly. The CR technology helps the secondary users to use or share licensed spectrum in an opportunistic manner by detecting the vacant portions of the frequency spectrum which are free from the primary users, selects the best available channel and releases this channel when a primary user is detected. Figure 1.3-2 depicts a flow process showing the different functional tasks and their relationships in a CR node.

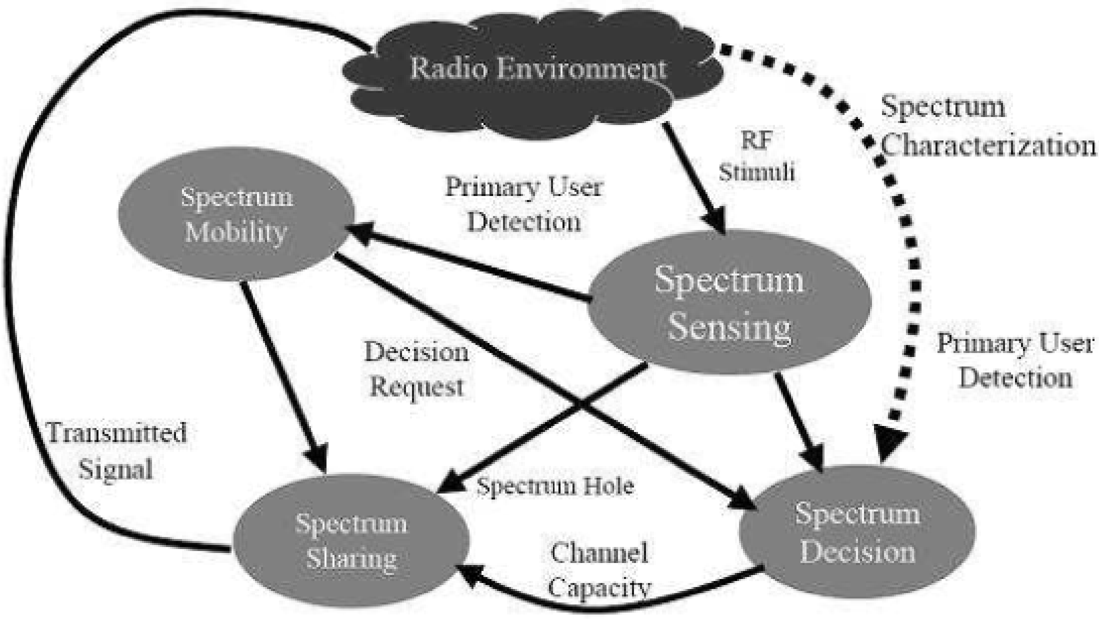


Figure 1.3-2: Functional flow process of Cognitive Radio.

The main functions of a CR node are as follows:

1.3.1 Spectrum sensing

Spectrum sensing is one of the main tasks in a CR node since it involves the spectrum discovery operation. The spectrum sensing task determines the available portion of the spectrum and detects the presence of any licensed users operating in that portion. This task has the most crucial role because it should detect the vacant licensed spectrum efficiently without causing any interference to the primary users. One of the most common spectrum sensing techniques used in CR is the transmitter detection [6], which is based on the detection of a weak signal from any primary user transmitter locally present in the detector's environment. In literature, there are three main types of transmitter detection techniques for spectrum sensing operation in CR.

a) *Matched Filter Detection*: This method is known as the optimum method for detection of primary users when the transmitted signal is known, since it maximizes the received signal-to-noise ratio (SNR) [6]. The main advantage of this technique is that it requires less time to achieve a certain probability of false alarm and miss detection. The main drawback of this technique is that the detection performance greatly depends on the accuracy of information about the primary users signal features such as bandwidth, operating frequency, modulation type and order, and pulse shaping, etc. If this information is not accurate, the performance of this detection method will be poor.

b) *Energy Detection*: This method is the most commonly used spectrum sensing techniques when the receiver does not have the signal information of the primary user [6]. In this technique, the received signal strength is compared with a threshold value which depends on the noise floor. The main drawbacks are that an improper selection of the threshold value will lead to misdetection of primary users, inability to differentiate interference from primary users and noise and poor detection performance under low SNR values.

c) *Cyclostationary feature detection*: This is a technique for detecting primary users by exploiting the cyclostationary features of the received signal [6]. The modulated signals are coupled with sinusoidal carriers, spreading codes or hopping sequences, which results in built-in periodicity. Cyclostationary features are caused by this periodicity in the signal or in its statistics such as mean or auto-correlation. This technique performs much better than the energy detection method as cyclostationary-based detection algorithms are more efficient in differentiating noise from primary user's signals.

1.3.2 Spectrum management or Spectrum decision

In CR networks, unused licensed and unlicensed spectrums are spread across a wide frequency range. The spectrum management task of the CR node characterizes the different available spectrum bands and selects the best available spectrum to meet the QoS requirements. It analyzes spectrum holes by considering their time-varying environment, primary user activities and the spectrum band information such as operating frequency and bandwidth. It also defines different parameters such as interference level, channel error rate, path loss and holding time for which a CR node can occupy the licensed spectrum without causing any interruption to the primary users. These parameters are useful in determining the quality of a particular spectrum.

1.3.3 Spectrum sharing

This task of the CR node provides a fair spectrum scheduling method to coexisting secondary users. Similar to medium access control, in CR networks, multiple CR nodes may attempt to access the same detected unused licensed spectrum at the same time, which calls for the spectrum sharing task to allocate and coordinate access to the unused licensed spectrum to avoid transmission collision.

1.3.4 Spectrum mobility

This task of the CR node maintains the seamless communication requirements of the secondary user while transiting from current spectrum band to another spectrum band. When a CR node needs to switch to another frequency due to bad channel conditions or the detection of primary user on the current channel, the spectrum mobility task of the CR node handles this transition by selecting another unused spectrum while maintaining its QoS requirements. This type of transition in CR networks is known as the *spectrum handoff*. The spectrum mobility task of the CR node handles this spectrum handoff.

1.4 White-Fi (IEEE 802.11af)

Research has shown that the major portion of the Ultra-High Frequency (UHF) spectrum remains unutilised or under-utilised for most of the time [10]. These unused or under-utilised spectrum portions are known as *white spaces*. These white spaces in UHF spectrum together constitutes more than 180 MHz of available bandwidth from channel 21 (512 MHz) to 51 (698 MHz) in the US (with the exception of channel 37). The prime incumbents of these channels are TV broadcasts and wireless microphone transmissions. The FCC had not allowed using these white spaces by unlicensed users in the past because it would interfere with the licensed users of these frequency spectrums. However, with the introduction of the DSA concept and CR technology, it is now possible for these white spaces to be used by other secondary users without interfering with the primary users. In 2008, FCC issued a ruling permitting the use of unlicensed devices in these white spaces [11]. Recent studies show that the use of these UHF white spaces in wireless networks especially in Wi-Fi systems, offers a substantial bandwidth and long transmission ranges [11].

White-Fi is a new concept which describes the use of Wi-Fi technology within TV white spaces using CR technology [11]. The IEEE has set up a new working group known as the IEEE 802.11af working group [12] to define a standard specification for White-Fi, similar to that of existing 802.11 standards. The main difference between 802.11a/b/g and 802.11af standards is that the latter will be based on CR technology and will be operating in TV white spaces instead of ISM bands. The CR technology is used in White-Fi because since 802.11af enabled devices are using TV white spaces, it is necessary to ensure that these devices does not create any undue interferences with the licensed TV transmissions.

The main advantages of White-Fi are as follows:

a) *Propagation characteristics:* The 802.11af White-Fi system uses TV white spaces with frequencies below 1 GHz. This allows long distance transmissions up to about 1 km. Current Wi-Fi systems use frequencies in the ISM band in 2.4 GHz and 5.8 GHz range, which allows only a transmission up to about 300 metres.

b) *Additional bandwidth:* A UHF channel is narrow (6 MHz wide in the US and 8 MHz in New Zealand) and prior research has shown that aggregating contiguous channels improves throughput [11]. A main feature of White-Fi is that it allows the aggregation of several unused contiguous TV channels to provide the same bandwidth of about 20 MHz that Wi-Fi uses on 2.4 GHz and 5.8 GHz and this in turn improves the rate of data throughput.

1.5 Multi-Hop Routing

Multi-hop wireless networks, such as MANETs, have gained world-wide research attentions in the past decade [13]. In multi-hop wireless networks, a source MS communicates with the destination MS through intermediate RSs over multiple hops. Here, each MS in the network acts as a potential mobile router capable of forwarding data packets to other MS. The route selection for forwarding the data packet is based on the multi-hop routing protocol used. There exist different classifications of multi-hop routing protocols used for MANETs, the most common being On-demand (Reactive) and Table driven (Proactive) routing protocols [14].

1.5.1 On-demand (Reactive) routing protocol

On-demand routing is a popular routing strategy for wireless ad-hoc networks [14]. In this routing technique, each node tries to reduce routing load by only sending routing packets only when a node has a data packet to transmit. On-demand routing protocols typically have two phases, route discovery and route maintenance. The route discovery phase determines all possible routes to some destination, only when a node wants to send a data packet and when there is no valid route already exists between the source and the destination node. The route maintenance phase maintains this discovered route temporally at the source node until this route expires or some events occur (e.g., a link failure) that requires another route discovery to be performed again. The most commonly used on-demand routing protocols are Dynamic Source Routing (DSR), and Ad-hoc On-demand Distance Vector (AODV) [14], [15].

The main difference between AODV and DSR is that the former uses a distributed routing table approach (hop-by-hop routing) whereas the latter uses a source routing approach [3]. In hop-by-hop routing, the source node is not required to maintain a complete sequence of intermediate hop nodes to reach the destination. Here, each node uses a routing table that stores only information about the next-hop (neighbouring node) to which packet should be forwarded for each destination. An important feature of AODV is the maintenance of timer based states in each node regarding the utilisation of individual routing table entries [15]. Routing tables are periodically checked for their validity. A routing table entry may expire after a specified time if it is not used within that time. A set of predecessor nodes is maintained for each routing table entry, indicating the set of neighbouring nodes which use that entry to route data packets. Whereas in source routing, the source node is required to maintain a complete sequence of intermediate hop nodes to reach the destination and it also determines the route that a data packet should follow to the destination. Among multiple routes, the sender of the packet selects a route used for the packet transmission. Like AODV, DSR does not use any periodic routing advertisement and neighbour detection packets.

1.5.2 Table driven (Proactive) routing protocol

Table driven routing protocols maintain consistent and up-to-date information of all possible routes to the destination at all times regardless of whether a route is needed for a data packet transmission. These protocols frequently broadcast messages to gather up-to-date information of all possible routes and maintains more than one route entry to store routing information. The most commonly used table driven routing protocols are Optimised Link State Routing (OLSR) and Destination Sequenced Distance Vector (DSDV) [15].

The main difference between OLSR and DSDV is that the former uses a link state approach whereas the latter uses a distance-vector approach. In link state approach, the nodes exchange information about their links (radio connectivity) with direct immediate neighbour nodes. OLSR uses Multi-Point Relays (MPR) for forwarding control traffic. MPRs advertise link state information periodically in their control messages. Whereas in distance-vector approach, all nodes in the network maintains a routing table which contains the list of all destination nodes along with the number of hops required to reach a particular node. Each entry in the routing table is marked with a sequence number assigned by the destination node. The sequence numbers are used to avoid the formation of duplicate routes.

1.6 Research Objective

The objective of this research is to firstly develop a model for describing the multi-interface CR mobile node and then investigate its opportunistic spectrum utilisation to enhance the routing performance in MCN. The interfaces used in this CR mobile node are IEEE 802.11a (Wi-Fi), IEEE 802.11af (White-Fi) and IEEE 802.16 (WiMax). The 802.11a and 802.11af interfaces are used for multi-hop communication while WiMax interface is used for single-hop communication with BS when multi-hop communication is not possible with 802.11a and 802.11af interfaces. The 802.11af interface uses the CR feature to perform channel selection. The multi-hop routing protocol used in this research is a modified version of AODV (Ad-hoc On-demand Distance Vector). The route selection by this routing protocol is not based on minimum hops as in conventional AODV but based on the output interface available (802.11a or 802.11af) when multi-hop communication is possible.

Network simulation is chosen as the research methodology over an analytical modelling or test bed approach. This is because the behaviour of the multi-interface CR mobile node and its interaction with other network entities is deemed too complex to be modelled mathematically and therefore not analytically tractable. Implementing a test bed of multi-interface CR mobile nodes is also not feasible given the limited time and budget available for the project. The widely used OPNET Modeller is adopted as the simulation tool for the development of the multi-interface CR mobile node model and for the evaluation of its impact on the routing performance in MCN. The following research questions are considered in this thesis: (a) How does the multi-interface CR mobile node model utilises the spectrum opportunistically? (b) What impact do multi-interface CR mobile nodes have on the routing performance in MCN? (c)What are the significant advantages of having multiple interfaces over single interface?(d) Are there any associated trade-offs in performance?

1.7 Thesis Structure

This thesis begins with a general introduction on MCN, CR, multi-hop routing, and White-Fi (IEEE 802.11af). Chapter 2 presents a detailed discussion on prior and related works carried out on MCRN, routing protocols for MCN, and CR based MCN. An overview of the OPNET Modeller simulation tool is presented in chapter 3. Chapter 4 and 5 collectively present the main contributions to this research: Chapter 4 describes the design and implementation of a multi-interface CR node model in OPNET and a modified AODV routing protocol based on this node model for enhanced routing performance in MCN; Chapter 5 presents the details of various experimental simulations carried out and the analysis of the simulation results. Finally, Chapter 6 concludes this thesis along with a discussion on possible directions for future work.

CHAPTER 2 Prior and Related Works

2.1 Multi-Hop Cognitive Radio Networks (MCRN)

CR is an emerging communication technology which allows the secondary users to use or share a licensed spectrum in an opportunistic manner by sensing unused portions of the licensed spectrum, selects the best available channel and releases this channel when a transmission by a primary user on this channel is detected [5]. Multi-Hop Cognitive Radio Networks (MCRN) are multi-hop self-organized and dynamic CR networks where CR users can communicate with each other through ad hoc connection [20]. When the destination MS is not in the range of the source MS, the data is forwarded to the destination MS through intermediate RSs over multiple hops. MCRN is different from most multi-hop networks in which all users operate in the same channel (or frequency band). In MCRN, multiple channels (or frequency bands) are used depending on the availability of the spectrum. Since MCRN is a multi-hop, multi-channel network with spectrum-awareness (enabled by CR), the MAC and Routing protocols of these networks involves much complex design and also faces many challenges. Some of them are as follows:

(i) The CR-MAC should be able to handle multiple channels.

(ii) The CR-MAC should be able to detect the presence of primary users in licensed spectrum bands.

(iii) The routing protocol should be able to exchange the local spectrum information between nodes and interact with multi-frequency scheduling in each node.

A number of research studies have been carried out on multi-channel CR-MAC and routing protocols for multi-hop CR networks. Some of the prominent works are outlined below.

2.1.1 Multi-channel CR-MAC for MCRN

This section briefly describes some of the prior research works carried out on multi-channel CR-MAC for MCRN. Timmers et al. [16] proposed a Distributed Multichannel MAC Protocol for MCRN which enables opportunistic spectrum sharing of licensed spectrum. Here, a Common Control Channel (CCC) is used for the transmission of control packets and this channel is assumed to be free from a primary user. This channel is also used to share information about the environment among the CR users. Kondareddy et al. [17] proposed a synchronised MAC protocol for MCRN which avoids the usage of a dedicated CCC. Here, the total time is divided into fixed time intervals and each interval represents one of the available channels. At the beginning of each time slot, all nodes within the network listen to a channel which the time slot represents for exchanging control signals. Choi et al. [18] proposed a full duplex multi-channel MAC protocol for MCRN which uses at least two transceivers. One transceiver is used for the transmission and reception of control packets while the second transceiver is used for the data packets.

2.1.2 Routing protocol for MCRN

This section briefly describes some of the prior research works carried out on routing protocol for MCRN. Ma et al. [19] proposed a spectrum aware routing for MCRN using only a single transceiver. Here, a modified version of AODV routing protocol is used. Since there is no dedicated CCC to share information about the available primary channels, the AODV Route Request (RREQ) packets are used to exchange this information among the CR nodes. Since all nodes may be tuned to different channels, RREQs are broadcasted on all available channels. Kamruzzaman et al. [20] proposed a spectrum and energy aware routing protocol for MCRN with channel timeslot allocation. This routing protocol selects an energy-efficient route and assigns channels and timeslots for a connection request. The main aim of this work was to improve the network throughput by distributing the traffic over different channels and timeslots. Chehata et al. [21] proposed an on-demand routing protocol for multi-hop multi-radio multi-channel CR networks. A multi-radio multi-channel on-demand approach is proposed here to effectively manage the transmission activities of primary and cognitive (secondary) users. Here, multiple channels are considered for data traffic while one dedicated channel is used for control traffic. Two radio interfaces are used: a fixed interface assigned to a fixed channel for long intervals for data traffic while the other is a switchable interface dynamically assigned to any data traffic channel for some short intervals. Here an on-demand routing protocol similar to AODV is employed which uses a Weighted Cumulative Expected Transmission Time (WCETT) metric. The source node broadcasts a RREQ packet which contains the measured WCETT metric and the channels to be used.

2.2 Routing Protocols for MCN

MCN combines the benefits of having fixed base stations and the flexibility of Multi-hop Ad-hoc Relaying Networks. In conventional SCNs, routing is fairly simple because communications between MSs are through BS over a single hop. However in MCN, the communications between MSs and also those between MSs and BS could be through intermediate RSs. This behaviour of communicating through intermediate RSs over multiple hops is much similar to multi-hop ad-hoc relaying networks. There are a number of routing protocols proposed for these multi-hop ad-hoc relaying networks. However, these protocols fail to provide a good solution for MCN because they are not capable of exploiting the presence of BS [1]. So the routing protocols designed for MCN should be able to work in an ad-hoc manner by establishing a route through the intermediate RSs and also in an infrastructure manner by detecting the presence of BS. Most existing research on routing protocols for MCN are based on modification of MANET routing protocols suitable to operate in a MCNs. Some of these works are outlined below.

Coll-Perales et al. [22] proposed an energy efficient routing protocol for MCN which uses mobile relays to forward packets from MS to the BS. The routing protocol used is a modified version of AODV which selects route using a linear cost function based on the number of hops, the state of channel congestion and node energy. Kannan et al. [23] proposed a cross layer routing protocol for MCN which considers multiple constraints for intermediate RS selection. The constraints imposed on RS are its willingness for co-operation and its neighbourhood connectivity. A fast neighbour detection scheme is proposed which adopts an explicit handshake mechanism to reduce neighbour detection latency instead of using periodic 'HELLO' messages as in traditional ad-hoc routing. Li et al. [24] proposed an efficient routing protocol to increase the communication reliability of MCN. Here, the routing protocol chooses a node with higher channel capacity to forward data to a BS, and the routing path length is adaptive to the channel condition of the forwarding nodes. When a node's channel quality to a BS is not sufficiently high or the buffer storage of the node is almost full, it chooses a relay node with higher capacity to forward the data to the BS.

2.3 Cognitive Radio based MCN

The multi-hop relaying capability of MCN potentially enhances the cell capacity or coverage by reducing the link path loss. Spectrum overlay could also be achieved in MCN based on the spatial isolation of the links [25]. CR technology could be used for this spectrum overlay technique which provides flexible and dynamic spectrum utilisation [5]. CRs allow Secondary Users (SUs) to access licensed bands that are under-utilised or unused by licensed or Primary Users (PUs). Such a spectrum is found in TV broadcast bands which possess better spectrum characteristics (e.g. lower path loss) [26]. CRs can thus be used with existing cellular system to form a CR-enabled cellular network which could offer greater bandwidth to mobile end users by tapping unused radio spectrum such in TV bands [26]. However, the existing cellular network is based on single-hop communication, which limits the potential benefits brought by CR to cellular networks. Thus, this research further investigates the use of CR for multi-hop communications in MCN. To the best of our knowledge, there are still very limited works in literature on CR based MCN. Some of them are outlined below.

Venkataraman et al. [27] proposed a multi-hop multi-band cognitive radio architecture for cellular network which makes use of a number of Cognitive Gateways (CG) in order to enable simultaneous usage of spectrum resources within the same cell. This CGs serves as a fixed relay for forwarding data packets from MSs to BS or to other CGs, as well as sensing vacant TV channels for dynamic spectrum access. Zhuang et al. [25] proposed a hierarchical and adaptive spectrum sensing approach in CR-based MCN based on cooperative sensing and soften hard detection fusion mechanisms with one-bit overhead. Here, the SUs that are able to make local detection decisions either directly report their one-bit decision to the CR BS or they forward their decision to other favourable SUs based on the states of their reporting channels. Xin et al. [28] proposed a joint admission control, channel assignment and QoS routing solution that maximizes the coverage of SUs in a CR cellular network which supports multi-hop secondary transmissions.

2.4 Further research

Section 2.1.1 and 2.1.2 describes the representative works carried out on multi-channel CR-MAC protocols, and routing protocols, respectively, for MCRN. The former are only focused on the MAC issues, while the latter are carried out in the context of infrastructure-less ad hoc networks rather than MCN. Section 2.2 describes a number of recent works carried out on routing protocols for MCN. The routing protocols proposed in these works are mainly based on single interface non-CR nodes. Section 2.3 describes some of the prominent works carried out on CR based MCN. However, none of these works proposed a multi-interface CR mobile node model and explored its opportunistic spectrum utilisation to enhance routing performance in MCN. For example, [25, 26] are focused on spectrum sensing issues; [27] proposed the use of fixed and dedicated CR-based relays rather than CR mobile nodes; and [28] is based on single-interface CR nodes without explicit consideration of node mobility.

The purpose of this research is to develop a multi-interface CR mobile node model and then investigate its opportunistic spectrum utilisation to enhance routing performance in MCN. This research also investigates and compares the performance improvement of multiple interfaces over single interface in MCN.

CHAPTER 3 OPNET MODELLER Overview

3.1 OPNET Modeller definition and features

OPNET Modeller is an object oriented, discrete event and general purpose network simulator with an extensive set of features to provide a modelling and simulation environment for designing network equipment and evaluating network architectures. The main features and capabilities of the OPNET Modeller are as follows [29]:

- Object orientation – OPNET Modeller consists of objects with configurable sets of attributes. Each object belongs to a particular class, which defines its characteristics in terms of behaviour and capability.
- Specialised in communication networks and information systems – OPNET Modeller provides many constructs relating to communications and information processing thus providing high leverage for modelling of networks and distributed systems.
- Hierarchical models – OPNET models are hierarchical and reflecting the structure of actual communication networks.
- Graphical specification – OPNET Modeller supports graphical specification where the models are entered through graphical editors. These editors provide an intuitive mapping from the modelled system to the OPNET model specification.
- Flexibility to develop detailed custom models – OPNET Modeller provides a flexible, high-level programming language with extensive support for developing custom models. This environment allows realistic modelling of all communication protocols and transmission technologies.
- Automatic generation of simulations – Model specifications are compiled automatically into executable, efficient, discrete-event simulations implemented in the C programming language. Advanced simulation construction and configuration techniques minimise compilation requirements.
- Application-specific statistics – OPNET Modeller provides built-in performance statistics that can be collected automatically during simulations.
- Interactive analysis – OPNET simulation incorporates support for analysis with the help of an interactive debugger.

- Animation – OPNET simulation runs can be configured to generate animations automatically of the modelled system and can also include animations of changing statistics over time.
- Co-simulation – OPNET Modeller is capable of interfacing with one or more other simulators such as Matlab to enable harnessing the strengths of different simulators.
- Application Program Interface (API) – OPNET models and data files can also be specified through a programmatic interface. This ability provides an alternative to graphical specification. This is useful for automatic generation of models.

3.2 OPNET Modeller Editors

System models in OPNET Modeller are specified with the help of a number of tools, known as *Editors*. These editors capture the characteristics of a modelled system's behaviour and also handle the required modelling information in a manner that is parallel to the structure of real network systems. Therefore, these model specification editors are organised hierarchically. There are different types of editors used in OPNET modeller. They are as follows [29]:

1) Project Editor
2) Node Editor
3) Process Editor
4) External System Editor
5) Link Model Editor
6) Packet Format Editor
7) ICI Editor
8) PDF Editor

Since the model specification editors are organised hierarchically, models built in the project editor rely on elements specified in the node editor. The node editor in turn, relies on the models defined in the process editor and external system editor. The Project, Node and Process editors are briefly explained below.

3.2.1 Project editor

The Project Editor is used to construct and edit the topology of a communication network model. It also provides basic simulation and analysis capabilities. The Network Domain in which the Project Editor works is the highest modelling level which encompasses objects that are defined in the other modelling domains. The Project Editor contains three fundamental object types namely, subnetwork, node, and link. There are several varieties of nodes and links which offer different capabilities. Each node or link is further specialised by its model, which determines its functions and behaviour. Table 3.2-1 lists the types of objects used in project editors.

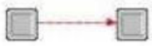
Table 3.2-1: Project Editor Objects [29].

Object Type	Definition	Default Representation
Fixed subnetwork	A container for additional network objects, including other subnetworks. Subnets can be nested to any depth to model complex hierarchical networks.	
Fixed node	A network terminal or device, usually with the ability to communicate to other nodes. Can have a wide range of capabilities, as determined by its model. Cannot change geographical positions over time.	

3.2.2 Node editor

The Node Editor is used to specify the structure of device models. These device models can be instantiated as node objects in the Network domains. The Node Editor is also used to define the interface of a node model which determines the aspects of the node model that should be visible to its user. This includes the attributes and statistics of the node model. The Node Editor contains different types of objects known as, *modules.* These modules are black boxes with attributes that can be configured to control the node's behaviour. Each of these modules represents particular functions of the node's operation and they can be active concurrently. Several types of connections support flow of data between the modules within a node. Table 3.2-2 lists the types of modules used in Node Editors.

Table 3.2-2: Node Editor Objects [29].

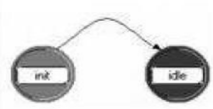
Object Type	Definition	Default Representation
Processor	General purpose, programmable object whose behavior is specified by a process model.	
Queue	General-purpose and programmable like a processor, but also provides internal packet queuing facilities consisting of a bank of subqueues. Subqueues are ordered lists of packets.	
Transmitter	Allows packets to be sent outside of the node's boundary via attached links. Two types of transmitters correspond to supported link types: point-to-point and bus.	
Receiver	Allows packets to be received from other nodes via attached links. Two types of receivers correspond to supported link types: point-to-point and bus.	
Packet Stream	Connects an output stream of a source module to the input stream of a destination module, allowing packets to be communicated and buffered between them.	
Statistic Wire	Connects an output statistic of a source module to the input statistic of a destination module, allowing numerical data to be communicated. Optional active notification of value changes via interrupts at the destination module.	

3.2.3 Process editor

The Process Editor is used to specify the behaviour of process models. Process models are instantiated as processes in the Node Domain and exist within processor, queue and esys modules. Processes can be independently executing threads of control that do general communications and data processing functions. They can represent functionality that would be implemented both in hardware and in software. The Process Editor is also used to define the interface of the model which determines the aspects of the process model that should be visible to its user. This includes the attributes and statistics of the process model. The language which the OPNET Modeller supports for developing process models is called *Proto-C*.

Process models use a Finite State Machine (FSM) paradigm to express behaviour that depends on current state and new stimuli. FSMs are represented using a State Transition Diagram (STD) notation. The states of the process and the transitions between them are depicted as graphical objects. Table 3.2-3 lists the types of objects used in Process Editors.

Table 3.2-3: Process Editor Objects [29].

Object Type	Definition	Representation
State	Represents a mode of the process which has been attained due to previous stimuli and corresponding decisions. States contain code expressing processing that is performed immediately after they are entered, or immediately before they are exited. A state can be forced or unforced. A process blocks immediately upon executing the enter code of an unforced state, at which point it waits for a new interrupt before continuing.	
Transition	Indicates a possible path that a process can take from a source state to a destination state. Each state can be the source and destination of any number of transitions. A transition has a condition statement which specifies the requirements for the process to follow the transition. An executive statement specifies actions that are to be taken when the process does follow the transition.	
Model-level information "blocks"	Several blocks of text specify additional components of the process, including: declaration of state (persistent), and temporary (scratch) variables; user-defined functions that can be called by the process' states and transitions; code to be executed upon process termination; and declaration of globally-scoped variables, data structures, etc.	

3.2.3.1 State Transition Diagram (STD)

Proto-C process models consist of two basic component types: States and Transitions, hence known as State Transition Diagram. States are generally used to represent the top-level modes that a process can enter. Transitions specify the changes in state that are possible for the process. STDs are graphically depicted in the Process Editor. There are also other important aspects of Proto-C process models that are not represented graphically. These include actions associated with each state, variable and attribute declarations, and common definitions of expressions and functions.

3.2.3.1.1 States

States in STD refers to the information that a process may have accumulated over the time that it has existed. The process's state may not include all information to which the process has had access, but only that which it has chosen to retain. State information may be continually updated as new events occur and data becomes available. Proto-C defines two types of states, *forced* and *unforced*. In Proto-C diagrams, forced states are graphically represented as green circles and unforced states are represented as red circles, as illustrated in Figure 3.2-1. On monochrome displays or hard copy, forced states are dark and unforced states are light. Specifications of actions may be associated with each Proto-C state. In Proto-C, actions are called *executives.* The executives of a state are of two types, *enter executives* and *exit executive.* The State's enter executives are executed when a process enters a state and exit executives are executed when the process leaves the state by following one of the outgoing transitions.

Figure 3.2-1: Graphical representation of Forced and Unforced states [29].

a) *Unforced states:* These states allow a pause between the enter executives and exit executives, and thus can model true states of a system. After a process has completed the enter executives of an unforced state, it blocks and returns control to the previous context that invoked it. This previous context may be another process that invoked this context or if the process was invoked by the Simulation Kernel. This blocking signifies the end of the current event and the kernel may select a new event to begin its execution. At this point, the process remains suspended until a new invocation causes it to progress into exit executives of its current state. Figure 3.2-2 depicts the flow of execution through the unforced states of an STD as interrupts cause the process to advance.

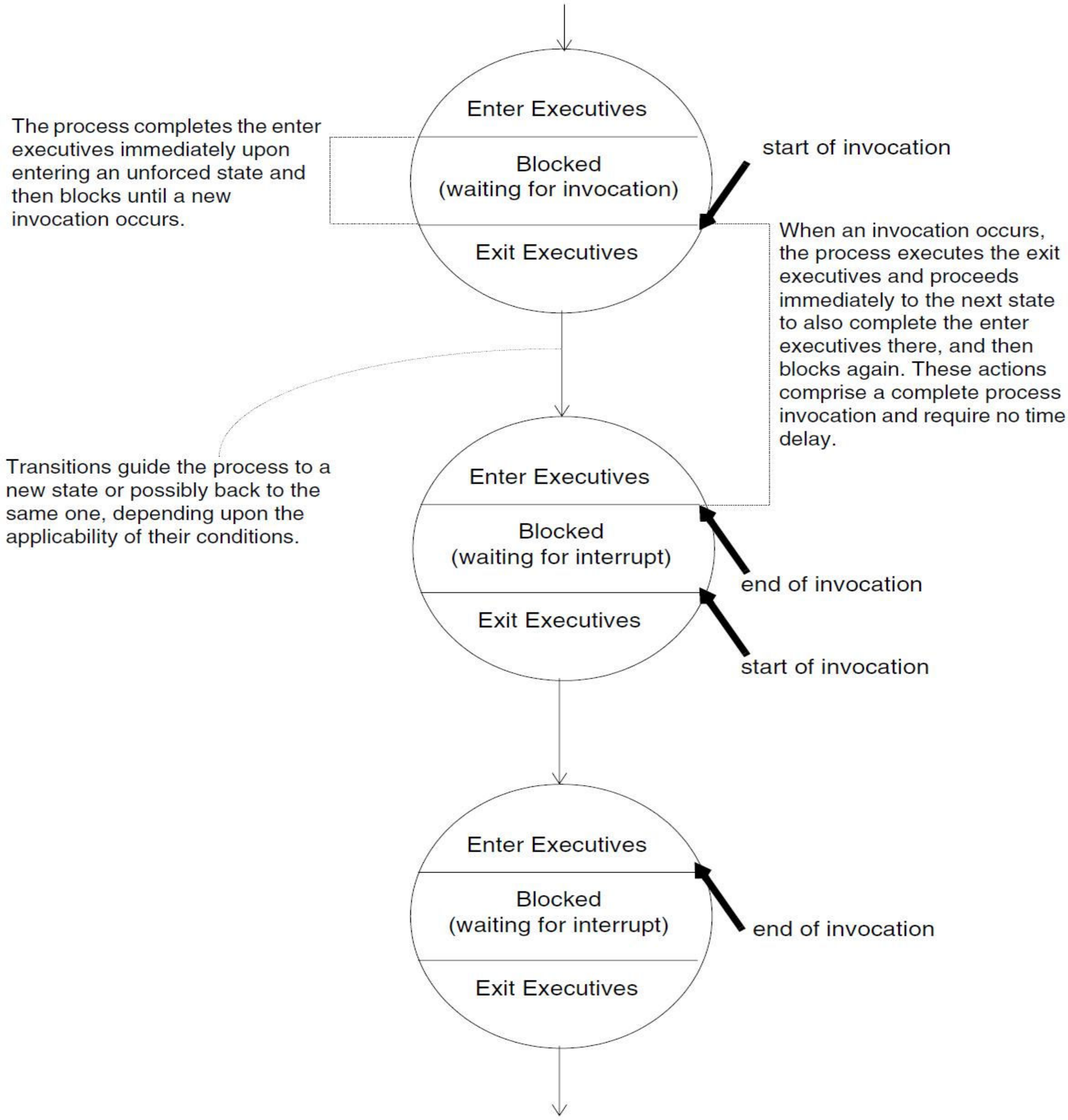


Figure 3.2-2: Execution flow through Unforced States [29].

b) *Forced states:* These states do not allow the process to wait. Therefore, these states cannot be used to represent modes of the system that persist for any duration. In this state, the exit executives are executed by a process immediately upon the completion of the enter executives. Therefore, the exit executives of a forced state are generally left blank because they are equivalent to the same statements placed at the end of the enter executives. These states are used less as compared to unforced states because these states cannot represent the actual system states.

3.2.3.1.2 Transitions

Transitions describe the possible movement of a process from state to state and the conditions under which such changes may occur. There are four components to a transition's specification. They are a source state, a destination state, a condition expression and an executive expression. When the control is in a source state and if any of the transition condition expression is true, then it implements the executive expression and transfer the control to the destination state.

In the Process Editor, transitions are specified graphically. Each state may have any number of outgoing and incoming transitions depicted as directed arcs with the arrow pointing toward the destination state. The condition and executive expressions appear in a combined label next to the arc. The condition expression is always placed in a parenthesis and precedes the executive expression by a forward slash ('/') to separate the two components. Transitions that have non-empty condition expression are depicted as dashed arcs and those without conditions are depicted as solid arcs, as shown in Figure 3.2-3.

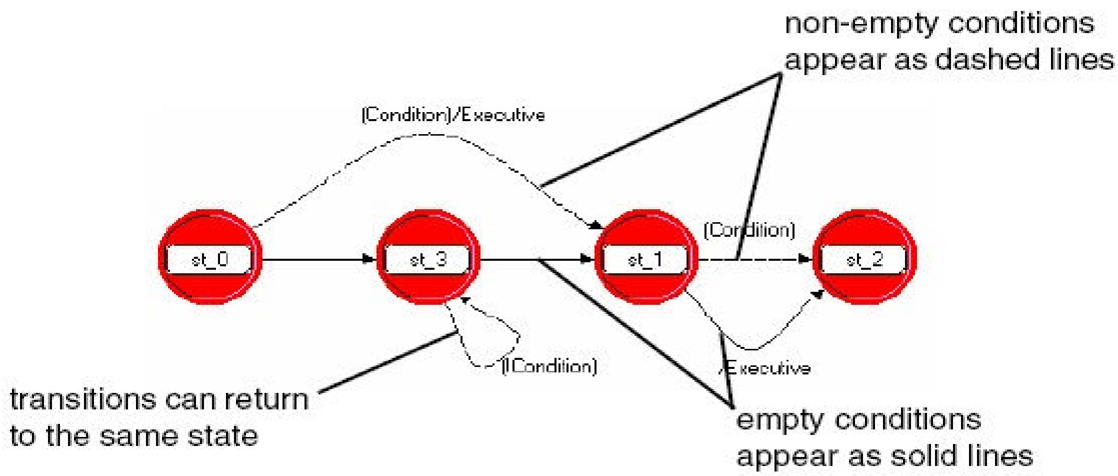


Figure 3.2-3: Representation of Transitions [29].

A transition's condition is evaluated as a Boolean expression to decide whether or not the process should enter the transition's destination state. A process evaluates outgoing transitions after the exit executive statements of the source state have completed. For forced states, enter and exit executives are executed in immediate succession but for unforced states, a pause occurs between enter and exit executives, and so only the latter can be grouped with the transitions.

3.3 OPNET Modeller Simulation Design

Simulation design in OPNET Modeller allows running or executing simulations of the network, node or process models created using the different editors and also allows collecting information from these models. The simulation design involves the following steps.

3.3.1 Specifying data collection

This step presents the various types of supported data that can be generated by a simulation and the mechanisms that enable its collection. After the construction of a network model, the simulations are run to examine the behaviour of the network model by considering different information generated from the model. For this, the OPNET Modeller requires additional specifications to cause that information to be generated. These additional specifications are specified prior to the simulation execution.

Discrete event simulations can generate different types of output information based on the different build-in statistics. It is also possible to define and compute new types of simulation output which can be reported during simulations or afterwards for post-processing.

Table 3.3-1 describes some of the common forms of simulation output and indicates the type of support that OPNET Modeller provides for generating it. Automatic support means that OPNET Modeller computes and records the information with no programming required by the modeller. Programmatic support means that OPNET Modeller provides programmatic interfaces that allow the Modeller to record customised output.

Table 3.3-1: Forms of Simulation Output [29].

Form of Output	Details	Support
Output Vectors	The history of a system variable can be captured as it varies over time in the form of an *output vector*. (OV) OVs provide insight into the manner in which the system evolves and its response to particular incidents during a simulation. Each OV consists of a series of values and associated times. A simulation can be configured to simultaneously collect multiple independent OVs.	Automatic for built-in statistics included in standard models; programmatic for customized statistics computed by user-defined code.
Output Scalars	Certain metrics of interest do not vary with time. Instead, they are one value that is representative of the system's performance over the course of the simulation (e.g., the throughput of the network). Each of these statistics is called an output scalar (OS). Each output scalar is typically recorded only once per simulation. Output scalars from multiple simulations are then combined to analyze their dependency on simulation inputs, also recorded as scalars.	Automatic for built-in statistics included in standard models; programmatic for customized statistics computed by user-defined code.
Animation	Visualization of system activity can be a valuable tool for gaining insight into behavior and interactions between components. Simulations can generate animations as they run or save them for playback after they complete.	Automatic for standard forms of animation included in standard models; programmatic for customized graphics computed by user-defined code.

3.3.2 Simulation construction

This step describes the various components of a simulation program and how a model specification is used to produce an executable simulation. Discrete event simulations are obtained by executing a simulation program, which is an executable file in the host computer's file system. A simulation program is a compiled program and takes the form of "object code". Object code consists of machine instructions that the host computer is capable of executing directly, without any further translation. Each discrete event simulation consists of many separate pieces of object code that are packaged together to form a complete program. The different pieces of object code play a variety of different roles in the simulation, as listed in the Table 3.3-2. Some of the object files are OPNET-provided, whereas others result from user specifications.

Table 3.3-2: Types of object files included in a simulation program [29].

Object File Type	Simulation Program
Simulation Kernel	Provides the framework for all simulations including basic services such as model loading, event scheduling and delivery, statistic collection, memory allocations, etc. The Simulation Kernel contains all Kernel Procedures (KPs) that are called by user-developed models.
Process Models	Each process model that is included in a simulation is compiled into a C language file with suffix ".pr.c"; this file is then compiled with the host computer's C compiler to generate an object file with suffix ".pr.o".
Pipeline Stages	Relied on by link models to implement modular computations and make decisions relating to the transfer of packets between transmitters and receivers. Each pipeline stage is a C language procedure within one C file with suffix ".ps.c". It is compiled using the host computer's C compiler into an object file with suffix ".ps.o".
External Object Files	Contain functions that play a supporting role for the process models and pipeline stages in a simulation. This software may be developed in C or in other languages that are callable from C. They can be compiled independently of OPNET Modeler and must have the suffix ".ex.o". External files can be made known to OPNET Modeler in two ways: 1) a process model's dependency on an external file can be expressed by declaring the external file in that model's specification; 2) a network model can express the dependency of any of its included models on an external file, via a similar declaration.
External Archives	Similar to external object files, but are packaged in the form of an archive containing multiple object files. This is sometimes a more convenient way of managing groups of object files that belong together as part of a "package".

3.3.3 Executing simulations

This step describes OPNET modeller's facilities for running simulations and interacting with them. Simulation execution is the final step in an iteration of a modelling experiment. Here, based on the results observed during this step, changes are made to the model's specification and additional simulations are executed. OPNET Modeller provides a number of options for running simulations, including internal and external execution, and the ability to configure attributes that affect the simulation's behaviour.

CHAPTER 4 Multi-Interface CR for Enhanced Routing Performance in MCN

This chapter deals with the development of a multi-interface CR mobile node model in OPNET Modeller. The multiple interfaces used in this node model are WiMax and WLAN. The WLAN interface is designed to operate either in IEEE 802.11a (Wi-Fi) mode or in IEEE 802.11af (White-Fi) mode. This chapter also describes the implementation of CR features such as spectrum sensing and spectrum management for use in White-Fi mode. Finally, this chapter also describes the development of a multi-hop routing protocol to enhance the routing performance of this CR mobile node in MCN.

4.1 Multi-interface (WiMax/WLAN) Node Model

The multi-interface CR node model developed in this project is based on OPNET's Wireless Local Area Network (WLAN) workstation node model and WiMax (IEEE 802.16) Subscriber Station (SS) node model. Figure 4.1-1 shows the OPNET's WLAN workstation node model and Figure 4.1-2 shows the WiMax SS node model.

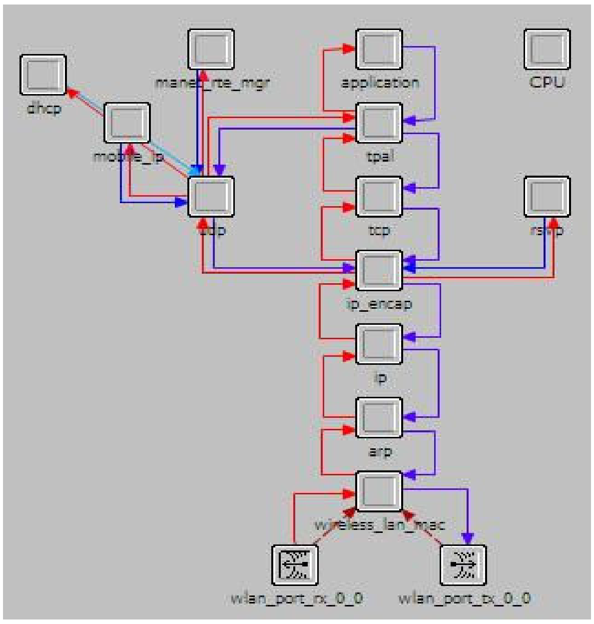


Figure 4.1-1: WLAN workstation node model.

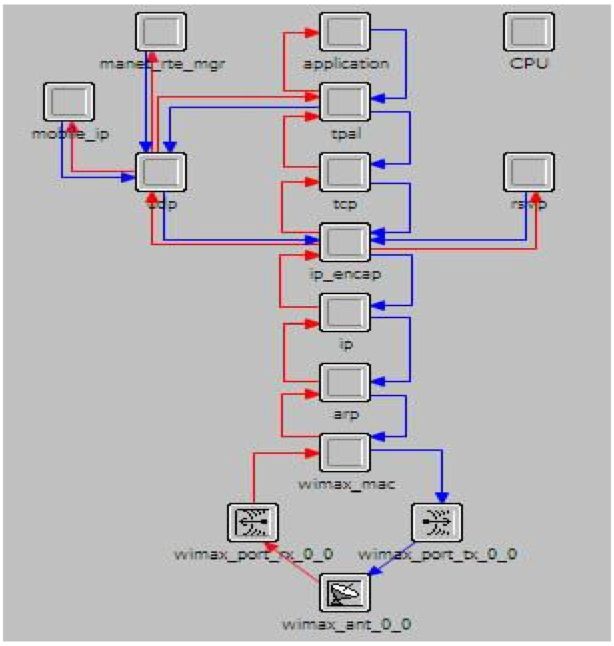


Figure 4.1-2: WiMax SS node model.

The proposed multi-interface node model has both WiMax and WLAN interface, as shown in Figure 4.1-3. The wimax_mac and the wimax_port transceivers together comprise the WiMax interface whereas the wlan_mac and the wlan_port transceivers together comprise the WLAN interface. Both of these interfaces are connected to a common Internet Protocol (IP) layer, thus sharing a common stack from the Application layer to the IP layer. This design of sharing a common stack enables the node model to function both as a Router and a Host.

The WLAN interface of this multi-interface node model is desired to operate in two different modes: IEEE 802.11a (Wi-Fi) and IEEE 802.11af (White-Fi). Since the original WLAN interface is designed to operate only in IEEE 802.11a mode, this interface is modified in-order to also incorporate the IEEE 802.11af mode. Since IEEE 802.11af is designed to operate in multiple channels, multiple transceivers are required for this multi-channel operation. Therefore, two pairs of transceivers and an additional receiver is used in the WLAN interface for this multi-channel operation. The first transceiver pair (wlan_port_rx_1_0 and wlan_port_tx_1_0) operates in the 5170 MHz frequency and is used for the transmission and reception of control packets only, while the second transceiver pair (wlan_port_rx_1_1 and wlan_port_tx_1_1) operates in multiple channels (5190 MHz for IEEE 802.11a or UHF TV channels for IEEE 802.11af) and is used for the transmission and reception of data packets only. The additional receiver (wlan_port_rx_1_2) is used for CR spectrum sensing operation.

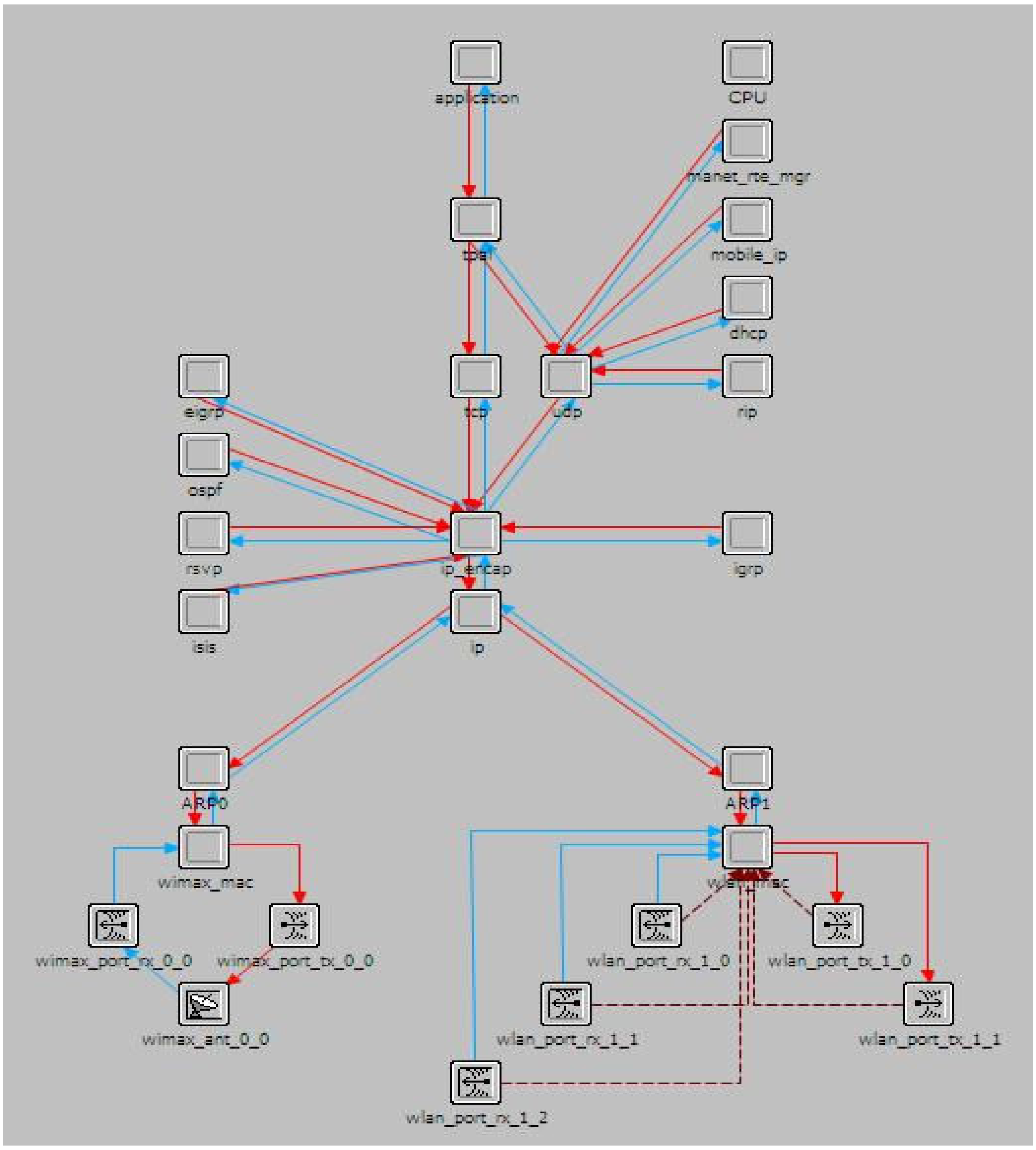


Figure 4.1-3: Multi-interface node model.

4.2 CR-MAC Model for WLAN (Wi-Fi/White-Fi) Interface

The CR-MAC model for WLAN interface in the multi-interface node model is a modified version of the original WLAN MAC layer model in OPNET. The original WLAN MAC model is modified in-order to incorporate the CR spectrum sensing and CR spectrum management techniques, and transceiver synchronisation techniques for multi-channel operations. Since the WLAN interface of this multi-interface node model consists of a second transceiver pair (wlan_port_rx_1_1 and wlan_port_tx_1_1) and an additional receiver (wlan_port_rx_1_2), it requires a registration of these additional transceivers in the CR-MAC model. The CR-MAC model for WLAN interface is shown in Figure 4.2-1.

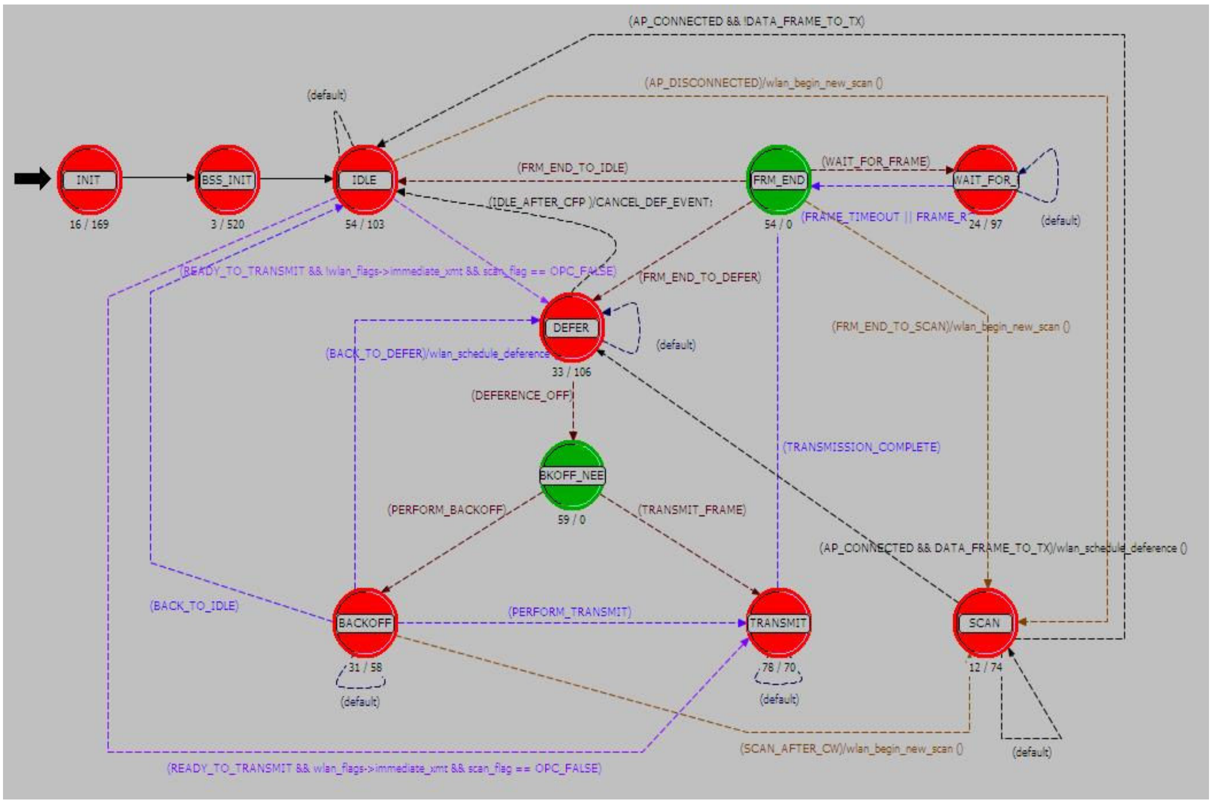


Figure 4.2-1: CR-MAC Process model.

The major steps involved in the development of the CR-MAC model are as follows:

4.2.1 Registration of additional transceivers in CR-MAC model

The original WLAN MAC model of the WLAN interface is designed to operate on a single channel using a single transceiver pair. The additional transceivers that require registration in the CR-MAC model of the WLAN interface are *wlan_port_rx_1_1*, *wlan_port_tx_1_1* and *wlan_port_rx_1_2*. The default transceiver pair *wlan_port_rx_1_0* and *wlan_port_tx_1_0* is configured to operate on a fixed channel frequency of 5170 MHz (based on the IEEE 802.11a standard) and used for the transmission and reception of control packets. The transceiver pair *wlan_port_rx_1_1* and *wlan_port_tx_1_1* is configured to operate on multiple channels and used for the transmission and reception of data packets. A separate receiver *wlan_port_rx_1_2* is used for the spectrum sensing operations. Figure 4.2-2 shows the code snippet for the registration of the additional transceivers in the CR-MAC model.

```
tx_objid1 = op_id_from_name (my_node_objid, OPC_OBJTYPE_RATX, "wlan_port_tx_1_1");
op_ima_obj_attr_get (tx_objid1, "channel", &channel_objid1);
txch_objid1 = op_topo_child (channel_objid1, OPC_OBJTYPE_RATXCH, 0);
op_ima_obj_attr_set (txch_objid1, "power", tx_power);
op_ima_obj_state_set (txch_objid1, OPC_NIL);

rx_objid1 = op_id_from_name (my_node_objid, OPC_OBJTYPE_RARX, "wlan_port_rx_1_1");
op_ima_obj_attr_get (rx_objid1, "channel", &channel_objid1);
rxch_objid1 = op_topo_child (channel_objid1, OPC_OBJTYPE_RARXCH, 0);
op_ima_obj_state_set (rxch_objid1, OPC_NIL);

rx_objid2 = op_id_from_name (my_node_objid, OPC_OBJTYPE_RARX, "wlan_port_rx_1_2");
op_ima_obj_attr_get (rx_objid2, "channel", &channel_objid2);
rxch_objid2 = op_topo_child (channel_objid2, OPC_OBJTYPE_RARXCH, 0);
op_ima_obj_state_set (rxch_objid2, OPC_NIL);
```

Figure 4.2-2: Registration of additional transceivers.

4.2.2 Spectrum sensing in CR-MAC model

Spectrum sensing is a CR technique used by secondary users to identify the vacant portion of the primary (licensed) spectrum. The proposed multi-interface node model which acts as a CR mobile station in a WiMax cellular network, opportunistically uses the vacant portions of the licensed TV spectrum, also known as *primary channels*. This CR mobile station senses for transmitting *primary users* of these spectrums such as TV broadcast stations, as well as for *secondary users* (i.e. other CR mobile stations) that are opportunistically transmitting on any of the available primary channels.

The original WLAN MAC model is modified to incorporate the CR spectrum sensing techniques for use by the IEEE 802.11af mode of the WLAN interface. The IEEE 802.11af mode is designed to operate on different available primary channels which are free from primary usage. When the mobile station is operating on 802.11af mode and has data packets to transmit, it first determines the vacant primary channels through spectrum sensing as mentioned above. The primary channels considered in this project are 10 New Zealand UHF TV channels with frequency limits [30] as shown in Table 4.2-1.

Table 4.2-1: UHF TV channels and their frequency limits.

Channel	Frequency Limits (MHz)
46	670 - 678
47	678 - 686
48	686 - 694
49	694 - 702
50	702 - 710
51	710 - 718
52	718 - 726
53	726 - 734
54	734 - 742
55	742 -750

4.2.2.1 Spectrum sensing for Primary Users

Spectrum sensing for the availability of these TV channels for secondary usage is considered on a probability basis. A uniform distribution generates a set of 10 primary channels. Table 4.2-2 shows the primary channel number and their minimum frequencies.

Table 4.2-2: TV channels and their minimum frequencies.

Channel Number	Minimum Frequency (MHz)
1	670
2	678
3	686
4	694
5	702
6	710
7	718
8	726
9	734
10	742

Figure 4.2-3 presents the pseudo code for determining the primary channel availability for secondary usage. Each simulation set generates an array of 10 elements which represents the 10 TV channels. The array elements can be either '0' or '1'. '0' represents a TV channel is free from primary usage and is available for secondary users whereas '1' represents a TV channel is being occupied by the primary user and it is not available for secondary users. All the mobile nodes within the network will have the same copy of the information about the status of the TV channels. Figure 4.2-4 shows an example simulation output console displaying the TV channel frequencies and its availability information. In this figure, the status of the TV channel frequencies 678.0, 686.0, 702.0, 710.0 and 742.0 MHz is '0', which means that these channels are available for secondary users while others are being occupied by primary users thus not available for secondary users.

```
Let channel_prob be an array of 10 values in steps of 0.1 from 0 to 1 generated from a uniform distribution
Let prob be any one value in steps of 0.1 from 0 to 1, representing  the desired probability of primary channel availability
Let i be the index of the array
Let channel_stat be an array of 10 binary values of either 0 or 1
For i starts at 0, i < 10, increment i by 1
If (channel_prob[i] < prob)
    channel_stat[i] = 0
 Else
       channel_stat[i] = 1
Endfor
```

Figure 4.2-3: Pseudo code for determining the primary channel availability.

```
Primary channel frequency 670.000000's status is 1
Primary channel frequency 678.000000's status is 0
Primary channel frequency 686.000000's status is 0
Primary channel frequency 694.000000's status is 1
Primary channel frequency 702.000000's status is 0
Primary channel frequency 710.000000's status is 0
Primary channel frequency 718.000000's status is 1
Primary channel frequency 726.000000's status is 1
Primary channel frequency 734.000000's status is 1
Primary channel frequency 742.000000's status is 0
```

Figure 4.2-4: Simulation output console on primary users scanning information.

4.2.2.2 Spectrum sensing for Secondary Users

This type of spectrum sensing is performed on each available primary channel individually by each mobile node in the simulation when there is a data packet to be transmitted. Before a data packet transmission, each node performs a frequency scanning operation to detect any secondary users already operating on those TV channels which are available for secondary usage. When a mobile node detects another secondary user operating on a particular channel, it skips that channel and move forward to the next available TV channel. Finally, it uses those TV channels which are free from other secondary users.

Spectrum sensing operation for secondary users is incorporated in the CR-MAC model. This spectrum sensing operation is performed by those nodes which have a data packet to be transmitted. When a data packet arrives at the MAC layer of this node from its higher layer and just before forwarding this data packet to the physical layer, the spectrum sensing operation is performed. Spectrum sensing is not performed for all data packets that arrive at the MAC layer. Data packets that arrive within some time interval since the last spectrum sensing operation can be transmitted immediately using the same available TV channel. This is to reduce the time delay experienced by data packets caused by the spectrum sensing operation. The time interval is calculated based on the time difference between current simulation time and the time at which the last spectrum sensing operation was performed. If this time difference is greater than 1 second, then a spectrum sensing operation is performed. For all data packets that arrive at the MAC layer, it checks the time interval conditions for spectrum sensing operation. If this condition is satisfied for a data packet, then a function 'wlan_channel_scan ()' is invoked which performs the spectrum sensing operation for secondary users. Figure 4.2-5 presents the pseudo code of the spectrum sensing operation for secondary users.

The spectrum sensing technique for detecting secondary users used in this project is based on *Energy Detection* technique which detects the presence of secondary users by determining the amount of signal strength in these channels. When the 'wlan_channel_scan ()' function is invoked, it checks the status of these TV channels one by one to determine whether it is available for secondary usage. If the status of a TV channel is '0', then this TV channel is considered to be vacant and it is available for secondary usage. Then to detect the presence of secondary users on this channel, the receiver *wlan_port_rx_1_2*'s channel frequency is set to this TV channel frequency, after which a self interrupt is scheduled to trigger after a certain amount of time. This amount of time is calculated based on the packet reception time and 44 μs is considered in this project. During this time, the receiver channel is set to this TV channel frequency and the receiver remains in this channel till the self interrupt is triggered. After this time duration of 44 μs, the self interrupt triggers and invokes an interrupt handling function. This function then compares the received power on this TV channel with the received power threshold (receiver sensitivity) value of the receiver, which in this project is −95 dBm. If the received power on this TV channel is greater than the received power threshold value, then this channel is considered to be used by another secondary user. If the received power on this channel is less than the received power threshold value, then this channel free from other secondary users. Likewise, all the available TV channels are scanned for secondary users. These vacant channels are then stored in an array for spectrum management operations.

```
Let channel_status be an array of 10 binary values of either 0 or 1, obtained after spectrum sensing for primary users
Let min_frequency represents the minimum frequency of each primary channel
Let CR_count be the index of the array representing the primary channels
Let t be the time required to scan each primary channel which is 44μs
//checks the status of the TV channel
out: If (channel_status[CR_count] == 0)
        //if the current primary channel is free from primary users
        Set  min_frequency = channel frequency of the current vacant primary channel represented by CR_count
        Set wlan_port_rx_1_2 receiver's channel frequency = min_frequency
        Schedule a self interrupt to occur after t seconds
      Else
        //if the current primary channel is occupied by a primary user, move on the next primary channel
        Increment CR_count by 1
        If (CR_count < 10)
          Goto out
        Else
          // No primary channel left for scanning
          Proceed to function call for final selection of channel frequency and total bandwidth
```

Figure 4.2-5: Pseudo code of CR spectrum sensing operation.

Figure 4.2-6 depicts a simulation output console displaying the status of the available TV channels after performing spectrum sensing for secondary users on these channels. The console shows that channel 1 is being occupied by other secondary users while channel 2 to channel 10 is being identified as free from other secondary users.

```
Secondary users present in channel 1
No Secondary users present in channel 2
No Secondary users present in channel 3
No Secondary users present in channel 4
No Secondary users present in channel 5
No Secondary users present in channel 6
No Secondary users present in channel 7
No Secondary users present in channel 8
No Secondary users present in channel 9
No Secondary users present in channel 10
```

Figure 4.2-6: Simulation output console on secondary users scanning information.

4.2.3 Spectrum management in CR-MAC model

Spectrum management is a CR technique used by secondary users to characterise the different available spectrum bands obtained after the spectrum sensing operation and selects the best available spectrum to meet the QoS requirements. Since White-Fi is an implementation of a Wi-Fi like protocol on top of the UHF white spaces [11], White-Fi should be able to provide a bandwidth equivalent to Wi-Fi which is normally 20 MHz for IEEE 802.11a. In-order to achieve a total bandwidth of 20 MHz, multiple vacant TV channels each with a bandwidth of 5 MHz should be used. In this project, depending on the number of available contiguous TV channels, three channel bandwidths of 5 MHz, 10 MHz and 20 MHz are considered in accordance to the draft specifications of IEEE 802.11af [12].

After performing the spectrum sensing operations for all available TV channels, the function 'wlan_channel_set ()' is invoked which performs the spectrum management operations. The 'wlan_channel_set ()' function has an array generated which holds the TV channels free from other secondary users. The main task of this spectrum management operation is to select the best available channel and also to provide the maximum possible bandwidth. Since in this project only the available contiguous TV channels are considered, the spectrum management operation has to first check whether the channels are contiguous. Figure 4.2-7 presents the pseudo code of the spectrum management operation. If it could find 3 contiguous channels, priority is given to these 3 contiguous channels so that it can provide a maximum bandwidth of 20 MHz. Otherwise, it will search for the next maximum number of contiguous channels (i.e. 2 contiguous channels) to provide the next preferred bandwidth of 10MHz. If it is again not available, it will use the first available TV channel to provide the minimum bandwidth of 5 MHz. In the event of no TV channel available after spectrum sensing operation, the node switches to operate in the IEEE 802.11a mode instead of IEEE 802.11af mode and uses a channel frequency of 5190 MHz and a bandwidth of 20 MHz. After determining the channel frequency and the total bandwidth to be used for data transmission, 'wlan_channel_set ()' function sets this channel frequency and bandwidth to the transceiver '*wlan_port_tx_1_1*' and '*wlan_port_rx_1_1*' respectively.

```
Let avail_channels be an array of TV (or primary) channels which are free from secondary users
Let CR_num be the total number of channels which are free from secondary users
Let avail_bandwidth be an array of channel bandwidths calculated based on channel contiguity
Let channel_frequency be the final chosen channel frequency for data transmission
Let total_bandwidth be the total bandwidth available from contiguous channels
Let j, k ,l and y  be the index of the array
For j starts at 0, j < CR_num, increment j by 1
     //consider j and j+1 channels in the avail_channels array
     //check whether they are contiguous by taking the difference between their channel numbers
     If difference between the channel numbers is 1
        //These two channels are contiguous
        avail_bandwidth[j] is 10.0
        For k starts at j+1, k < CR_num, increment k by 1
           //consider k and k+1 channels in the avail_channels array
           //check whether they are contiguous by taking the difference between their channel numbers
           If difference between the channel numbers is 1
             //These two channels are contiguous
             avail_bandwidth[j] is 20.0
                break
             Endfor
           break
        Endfor
Endfor

//checking the array avail_bandwidth whether any channel is providing a bandwidth of 20.0 MHz
For y starts at 0, y < CR_num, increment y by 1
      If (avail_bandwidth[y] = 20.0)
          Set channel_frequency = frequency of the TV channel with channel number y
          Set total_bandwidth = 20.0
          Goto in
Endfor

//checking the array avail_bandwidth whether any channel is providing a bandwidth of 10.0 MHz
For y starts at 0, y < CR_num, increment y by 1
      If (avail_bandwidth[y] = 10.0)
          Set channel_frequency = frequency of the TV channel with channel number y
          Set total_bandwidth = 10.0
          Goto in
Endfor

//checking the array avail_bandwidth whether any channel is providing a bandwidth of 5.0 MHz
For y starts at 0, y < CR_num, increment y by 1
      If (avail_bandwidth[y] = 5.0)
          Set channel_frequency = frequency of the TV channel with channel number y
          Set total_bandwidth = 5.0
          Goto in
Endfor

//If no TV channels are available, then it will operate in IEEE 802.11a mode
If (channel_frequency = 0.0)
    Set channel_frequency = 5190.0
    Set total_bandwidth = 20.0


//setting the selected channel frequency and bandwidth to the transceiver
in: Set the frequency of transmitter 'wlan_port_tx_1_1' to the channel_frequency
    Set the frequency of receiver 'wlan_port_rx_1_1' to the channel_frequency
    Set the bandwidth of transmitter 'wlan_port_tx_1_1' to the total_bandwidth
    Set the bandwidth of receiver 'wlan_port_rx_1_1' to the total_bandwidth
```

Figure 4.2-7: Pseudo code of the CR spectrum management operation.

4.2.4 Transceiver synchronisation in CR-MAC model

Spectrum sensing and spectrum management operations are performed only by those CR mobile nodes which has a data packet to be transmitted. The spectrum management operation sets the selected channel frequency and bandwidth only on the transceiver of these transmitting nodes. The destination nodes of these nodes are not aware of this channel frequency and bandwidth selection and cannot set its transceiver accordingly. Hence, it is required to perform transceiver synchronisation between the source and the destination nodes so that the destination node is aware of this change in channel frequency and bandwidth and can set its transceiver accordingly.

Transceiver synchronisation operations considered in this project are based on the OPNET modeller's Process Registry and are initiated at the source nodes. The Process registry defines a group of procedures that allow defined processes to record, access, and share information in a model-wide (or global) registry. This process registry information is shared by all the nodes within the network. Hence, the source node can set the transceiver of the destination node to the channel frequency and bandwidth selected by the spectrum management operations.

The source node obtains a handle to the destination node from the global process registry. This handle is obtained with help of the destination node address which is extracted from the data packet field. This process registry handle enables the source node to have an access to the destination node's *'wlan_port_rx_1_1'* receiver and can set this receiver's frequency and bandwidth to the selected channel frequency and bandwidth. Before setting this receiver to the selected channel frequency and bandwidth, the receiver status should be checked to identify whether the receiver is idle. The status of the receiver is obtained by comparing the packet reception end time of this receiver with the current simulation time. If the receiver is busy with the reception of data packets from other nodes, the scanning operation performed by the source node continues till the receiver becomes idle. Once the receiver becomes idle, the frequency and bandwidth of this receiver is set to the selected channel frequency and bandwidth. Hence, the transmitting and the receiving nodes are synchronised. The code snippet for transceiver synchronisation operation is shown in Figure 4.2-8.

```
proc_record_handle_list_ptr = op_prg_list_create ();
oms_pr_process_discover (OPC_OBJID_INVALID, proc_record_handle_list_ptr, "address", OMSC_PR_INT64, dest_node_addr, OPC_NIL);
process_record_handle = (OmsT_Pr_Handle) op_prg_list_access (proc_record_handle_list_ptr, OPC_LISTPOS_HEAD);
oms_pr_attr_get (process_record_handle, "node objid", OMSC_PR_OBJID, &dest_node_objid);

dest_rx_objid = op_id_from_name (dest_node_objid, OPC_OBJTYPE_RARX, "wlan_port_rx_1_1");
op_ima_obj_attr_get (dest_rx_objid, "channel", &channel_objid);
dest_rxch_objid = op_topo_child (channel_objid, OPC_OBJTYPE_RARXCH, 0);
rx_state_info_ptr1 = (WlanT_Rx_State_Info *) op_ima_obj_state_get (dest_rxch_objid);

if(rx_state_info_ptr1->wlan_pk_rx_end_time >= current_time)
    {
    wlan_channel_scan ();
    }
else
    {
    op_ima_obj_attr_set (dest_rxch_objid, "min frequency", channel_frequency);
    op_ima_obj_attr_set (dest_rxch_objid, "bandwidth", total_bandwidth * 1000.0);
    }
```

Figure 4.2-8: Transceiver synchronisation operation.

4.3 Routing Protocol for Multi-Interface CR Node Model

The routing protocol proposed in this project is a modified version of AODV routing protocol. The proposed routing protocol is not based on the minimum hops as in conventional AODV routing protocol but based on the available output interface used for data transmission. The multi-interface CR node model proposed in this project has mainly two interfaces: WiMax and WLAN. The routing protocol for this multi-interface node model should be able to make a proper selection among these two interfaces. Hence, this routing protocol is designed to make a selection among these two interfaces based on the following conditions:

(i) If the routing protocol discovers a multi-hop path and if there is any vacant TV channel available, then WLAN interface with White-Fi (IEEE 802.11af) mode is used for data transmission.

(ii) If the routing protocol discovers a multi-hop path and if there is no vacant TV channel available, then WLAN interface with Wi-Fi (IEEE 802.11a) mode is used for data transmission.

(iii) If the routing protocol could not discover a multi-hop path, then the conventional single-hop communication using WiMax (IEEE 802.16) interface is used for data transmission.

The conventional AODV routing protocol is a reactive routing protocol which has mainly two phases, route discovery and route maintenance [3]. In the route discovery phase, when the source node needs to send data to the destination node, it first looks up its routing table to check if a valid route already exists to the destination node. If no route exists, then the source node broadcasts a Route Request (RREQ) with packet fields containing information about the source address, source sequence number, destination address, destination sequence number and hop count. The RREQ is flooded through the network until it reaches an intermediate node that has a route to the destination or until it reaches the destination node. Each intermediate node that forwards the RREQ creates a reverse route for itself back to the source node. When the RREQ reaches an intermediate node which has a valid route to the destination node, generates a Route Reply (RREP) containing information about the number of hops necessary to reach the destination node and the sequence number for the destination node. If there is no intermediate node which has a valid route to the destination node, the destination node itself generates a RREP. Each node that forwards this RREP towards the source node creates a forward route to the destination node.

The Route Maintenance phase maintains the pre-determined routes. This is performed using three different types of messages: Route Error (RERR), 'Hello' (keep-alive), and Route Time-out message. The Route Time-out message is generated when there is no activity on a route for certain amount of time and the link will be deleted from the route table. The periodic 'Hello' messages between the neighbour nodes are required to prevent the forward and reverse route from expiration. If one of the links in a route fails, a RERR is generated by the intermediate node and forwards that to the source node. Then the source node will initiate a new route discovery process.

In the modified AODV routing protocol proposed for the multi-interface CR node model, when a source node has a data packet to transmit, it first broadcast a RREQ packet through both WiMax and WLAN interfaces. The RREQs received by the WLAN interface of intermediate relay nodes re-broadcast these RREQs till it reaches the destination node whereas the RREQ broadcasted through the WiMax interface of the source node is received only by the WiMax interface of the WiMax base station (BS). The BS then re-broadcast this RREQ and it is being received by the WiMax interface of all mobile nodes. However, these mobile nodes are not allowed to re-broadcast the RREQ through their WiMax interface, hence all mobile nodes except the destination node will destroy the RREQ packet received through their WiMax interface. When the destination node receives the RREQ through WLAN interface, it sends a RREP packet back to the source node. The destination node also sends a RREP through WiMax interface and it is being received by the WiMax interface of the BS. The BS then forwards this RREP to the source node. Thus, under an ideal fully connected topology, the source node should receive RREP through both its interfaces.

When the source node receives a RREP, it first checks its route table to determine if any route to this destination already exists. If there is no existing route in the route table, the source node sends it buffered data packets through the interface from which it received the RREP. This can be either WiMax or WLAN interface. If there is an existing route in the routing table, the routing protocol checks for the interface type of the route entry. If the interface type is WiMax, then the route table is updated when a multi-hop path through WLAN interface is discovered. If the interface type is WLAN, then the route table is updated only when a multi-hop path with less number of hops is discovered. Thus, this proposed routing protocol always prefers the WLAN interface for data transmission if a valid multi-hop path to the destination exists, and if no multi-hop path exists, it uses the single-hop path through the WiMax interface. Figure 4.3-1 presents the pseudo code of the proposed routing protocol for multi-interface CR node model.

After discovering a multi-hop path through the WLAN interface, the source node is required to select the mode of the WLAN interface (IEEE 802.11af or IEEE 802.11a) for data packet transmission. This selection is made by the MAC layer of the source node after performing the spectrum sensing and spectrum management operations. After these operations, if vacant TV channels are available, the WLAN interface will operate in IEEE 802.11af mode and if there is no vacant TV channel, it will operate in IEEE 802.11a mode.

```
Let R be a multi-interface CR node receiving a data packet p for destination D
Let N be the set of one-hop neighbours of R
Let IF0 be the WiMax interface of R
Let IF1 be the WLAN (White-Fi/Wi-Fi) interface of R
Let BS be the WiMax base station
Let RREQ denotes the route request packet
Let RREP denotes the route reply packet
// Broadcasting Route Request packet
If (R = source of p and no route exists to D)
   Broadcast RREQ for p through IF0 and IF1interfaces of R
If (n ∈ N receives RREQ through IF1)
   If (n = D)
      Send a RREP through IF1 of n
   Else
      Re-broadcast RREQ through IF1 of n
Else if (n = BS receives RREQ through IF0)
   Re-broadcast RREQ through IF0 of n
Else if (n ∈ N receives RREQ through IF0)
   If (n = D)
      Send a RREP through IF0 of n
    Else
      Discard RREQ

//Receiving Route Reply packet
If (R = destination of RREP)
  //Checks the route table, if a route already exists to D from R
  If (no route exists)
     Create a new route table entry irrespective of the interface (IF0 or IF1) through which the RREP is received
     Send all buffered data packets through this interface
  Else
     //A route already exists to D from R, implying this is a second RREP received through an interface different
     // from the first for the same RREQ
      If (RREP is received through IF1 of R)
        // Give preference to routing through WLAN (White-Fi/Wi-Fi) interface
        Update the route table entry
        Send all buffered data packets through IF1
```
Figure 4.3-1: Pseudo code of the routing protocol for multi-interface CR node model.

CHAPTER 5 Simulation and Result Analysis

This chapter deals with the simulations performed to investigate the performance of the proposed multi-interface CR mobile node in a WiMax cellular network. The basic network setup in OPNET, simulation parameter settings, simulation scenarios and the performance metrics used are explained in this chapter. This is followed by the presentation and analysis of the results obtained from these simulations.

5.1 Network Setup in OPNET

A WiMax (IEEE 802.16) cellular network is setup as shown in Figure 5.1-1. The network consists of a single WiMax cell of radius 2 km, a WiMax BS, and 150 multi-interface CR mobile nodes distributed randomly within the network. A *mobility config* and *WiMax* node objects are placed within the simulation environment, which define the mobility configuration and WiMax attributes respectively, for the cellular network.

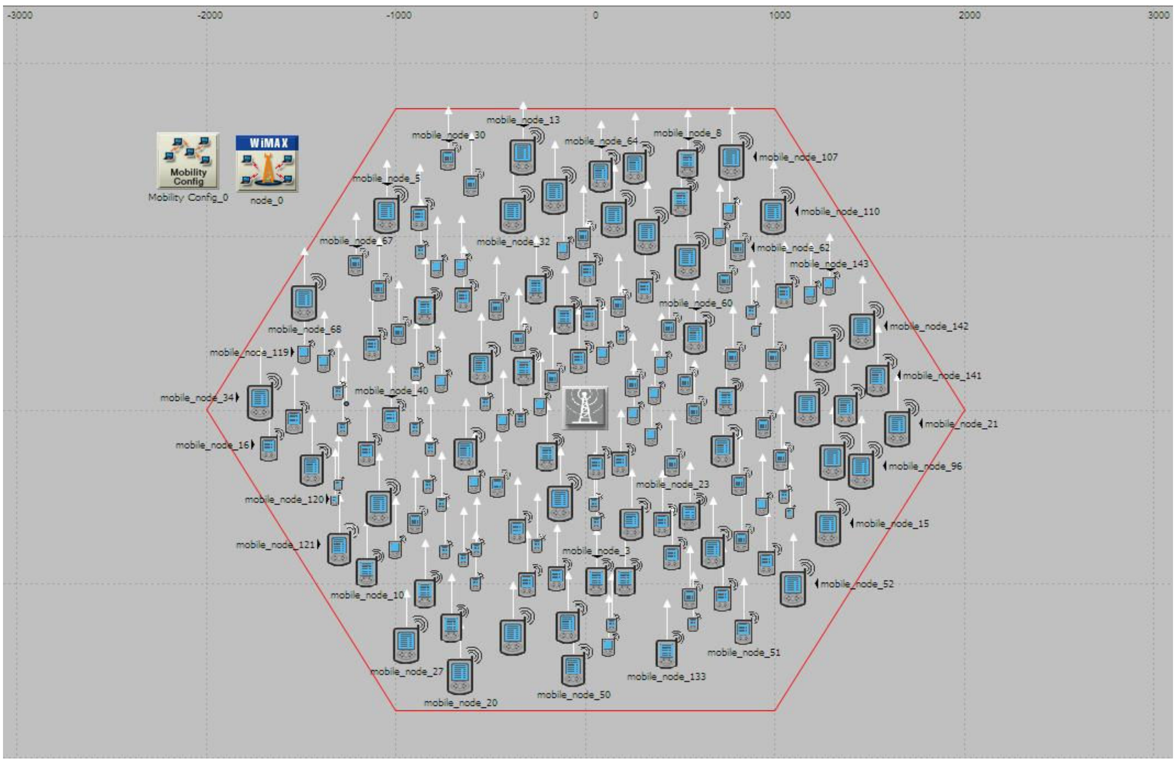


Figure 5.1-1: WiMax cellular network.

5.2 Simulation Parameters

Unless otherwise shown in the following tables, the simulation parameters use their default values as defined by the OPNET modeller. More detailed information about the simulation parameters and their values are provided in the appendix.

Table 5.2-1: General parameters information.

Parameters	Values
Network Size (nodes)	150
Mobility Model	Random Waypoint
Application Model	Application Demand
Data Traffic Type	Constant Bit Rate (CBR)
Traffic Start Time (seconds)	Uniform(20,50)
Traffic End Time (seconds)	Uniform(50,80)
Packet Size (bytes)	512
Packet Rate (packets/second)	4
WLAN Data Rate (Mbps)	24
WLAN Transmission Power (Watt)	0.01
Simulation Time (seconds)	100

Table 5.2-2: WiMax configuration information.

Parameters	Values
WiMax cell radius (km)	2
Transmission power (Watt)	0.05
Multipath Channel Model	ITU Pedestrian A
Pathloss Model	Suburban Fixed (Erceg)
Terrain Type (Suburban Fixed)	Terrain Type B
Shadow Fading Standard Deviation	2.0
Service Class Name	Bronze

Table 5.2-3: AODV routing protocol information.

Parameters	Values
Route Request Retries	5
Destination Only Flag	Enabled
Active Route Timeout (seconds)	3
Hello Interval (seconds)	Uniform (1, 1.1)
Allowed Hello Loss	4
Timeout Buffer	4

5.3 Simulation scenarios

The *mobility config* node object defines the mobility model and the movement speed for the multi-interface CR mobile nodes. These mobile nodes are allowed to move only within the WiMax cell and communicate with each other either through their WiMax interface via the BS or through their WLAN (White-Fi/Wi-Fi) interface over one or more hops. The underlying transport layer protocol used is User Datagram Protocol (UDP). All results obtained are based on the average of 10 simulation runs, each of duration 100 seconds.

The investigation on the performance of the multi-interface CR mobile node is carried out by performing simulations under different experimental scenarios, defined by the following three experimental parameters:

(i) Probability of primary channel availability
(ii) Node mobility
(iii)Number of traffic sources

Figure 5.3-1 depicts an example experimental scenario where 20 source-destination pairs are randomly selected to generate the traffic pattern, and each mobile node is randomly placed and assigned with a movement speed and direction. Various simulations are performed with different sets of values for the experimental parameters as shown in Table 5.3-1. The values underlined are the default values for the respective parameters.

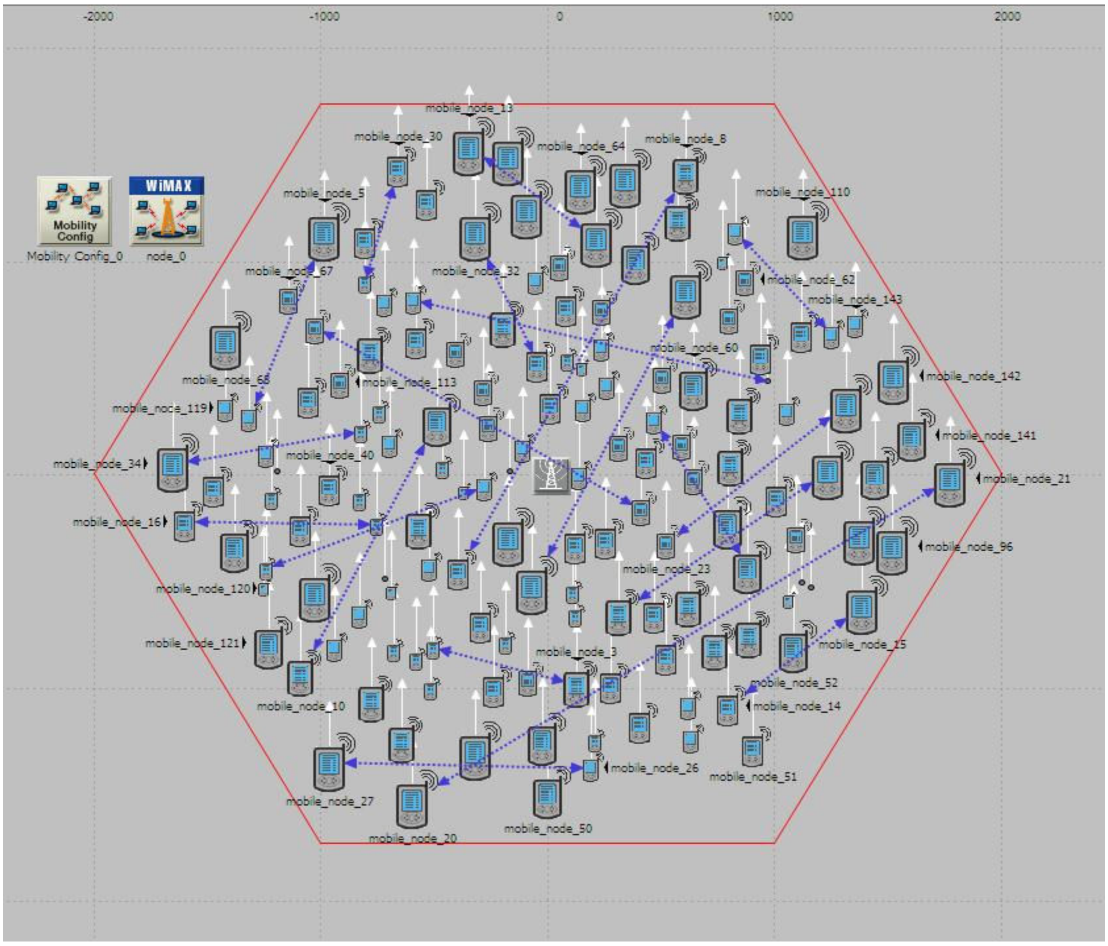


Figure 5.3-1: Experimental scenario.

Table 5.3-1: Experimental parameters and their values.

Parameters	**Values**
Probability of primary channel availability	0.0, 0.3, 0.5, 0.8, 1.0
Node mobility	0.0, 0.5, 1.0, 1.5, 2.0
Number of traffic sources	10, 15, 20, 25, 30

Accordingly, the simulation tasks are classified into three categories:

(i) Varying probability of primary channel availability
(ii) Varying node mobility
(iii) Varying the number of traffic sources

A brief description of the different scenarios for the three above mentioned categories are presented in the following sections.

5.3.1 Varying probability of primary channel availability

In this investigation, the effect of primary channel availability on the performance of the multi-interface CR based MCN is evaluated. Table 5.3-2 show five different probabilities of primary channel availability for five different scenarios while keeping node mobility and number of traffic sources to their default values.

Table 5.3-2: Varying probability of primary channel availability.

Scenario	Primary channel probability	Node mobility (m/s)	Number of traffic sources
1	0.0	1.0	20
2	0.3	1.0	20
3	0.5	1.0	20
4	0.8	1.0	20
5	1.0	1.0	20

5.3.2 Varying node mobility

In this investigation, the effect of node mobility on the performance of the multi-interface CR based MCN is evaluated. Table 5.3-3 show five different node speeds for five different scenarios while keeping the probability of primary channel availability and number of traffic sources to their default values.

Table 5.3-3: Varying node mobility.

Scenario	Primary channel probability	Node mobility (m/s)	Number of traffic sources
1	0.5	0.0	20
2	0.5	0.5	20
3	0.5	1.0	20
4	0.5	1.5	20
5	0.5	2.0	20

5.3.3 Varying traffic sources

In this investigation, the effect of the number of traffic sources on the performance of the multi-interface CR based MCN is evaluated. Table 5.3-4 show five different number of traffic sources for five different scenarios while keeping the probability of primary channel availability and node mobility to their default values.

Table 5.3-4: Varying traffic sources.

Scenario	Primary channel probability	Node mobility (m/s)	Number of traffic sources
1	0.5	1.0	10
2	0.5	1.0	15
3	0.5	1.0	20
4	0.5	1.0	25
5	0.5	1.0	30

5.4 Performance Metrics

A metric is a standard measure for assessing the performance of a network. Three performance metrics used in this project are as follows:

- *Packet delivery ratio (PDR)*: This refers to the ratio of the total number of data packets successfully delivered to the destination node to the total number of data packets sent from the source node.
- *End-to-end delay*: This refers to the average end-to-end packet delay from the time the data packet is generated by the source node to the time it is received by the destination node. This includes the time for route discovery, packet transmission, propagation, and channel scanning due to spectrum sensing operation.
- *Routing load*: This refers to the total number of routing packets generated, including route request packets, route reply packets and route error packets.

5.5 Results and Analysis

The results of the simulations and the analysis of these results are presented in this section. All line graphs are plotted with 95% confidence interval for each data point. Simulations are performed for different probabilities of primary channel availability, different node speeds and different number of traffic sources. Therefore, the results for these simulations are also classified into three categories:

(i) Varying probability of primary channel availability
(ii) Varying node mobility
(iii) Varying Traffic sources

5.5.1 Varying probability of primary channel availability

This section presents the results obtained under different probabilities of primary channel availability while keeping the node speed and number of traffic sources to their default values. The non-zero probability values indicate that the multi-interface CR mobile node can operate in White-Fi (IEEE 802.11af) mode when using the WLAN interface whereas a zero probability value indicates that the node can only operate in Wi-Fi (IEEE 802.11a) mode when using the WLAN interface.

5.5.1.1 PDR versus primary channel probabilities

Figure 5.5-1 shows the PDR as the primary channel probability increases from 0 to 1. It is observed that without using any TV channels, i.e. zero primary channel probability and thus using only Wi-Fi channels for data transmission, PDR is at its lowest (< 70%). This is mainly due to channel congestion. On the other hand, in addition to providing additional capacity, the better propagation characteristics at TV frequencies enable a higher reception SNR, resulting in better PDR when primary channels are used, i.e. when primary channel probability > 0.

The PDR increases from minimum to 83% but gradually decreases to about 78% as the primary channel availability increases. The gradual decrease in PDR with increasing primary channel probability from 0.3 is unexpected and may be attributed to the increased scanning time due to a higher number of primary channels to be scanned with higher primary channel probabilities. This scanning delay in turn causes the MAC layer of the multi-interface CR node to delay the transmission of acknowledgements for data packets received, resulting in subsequent discarding of unacknowledged data packets after a maximum number of retransmissions by the transmitting node. Table 5.5-1 shows the observed number of missed acknowledgment (ACK) packets by transmitting nodes for different primary channel probabilities. It shows the number of missed ACK packets increases with primary channel probability, which consequently results in lower PDR (though still higher than when using Wi-Fi channels only).

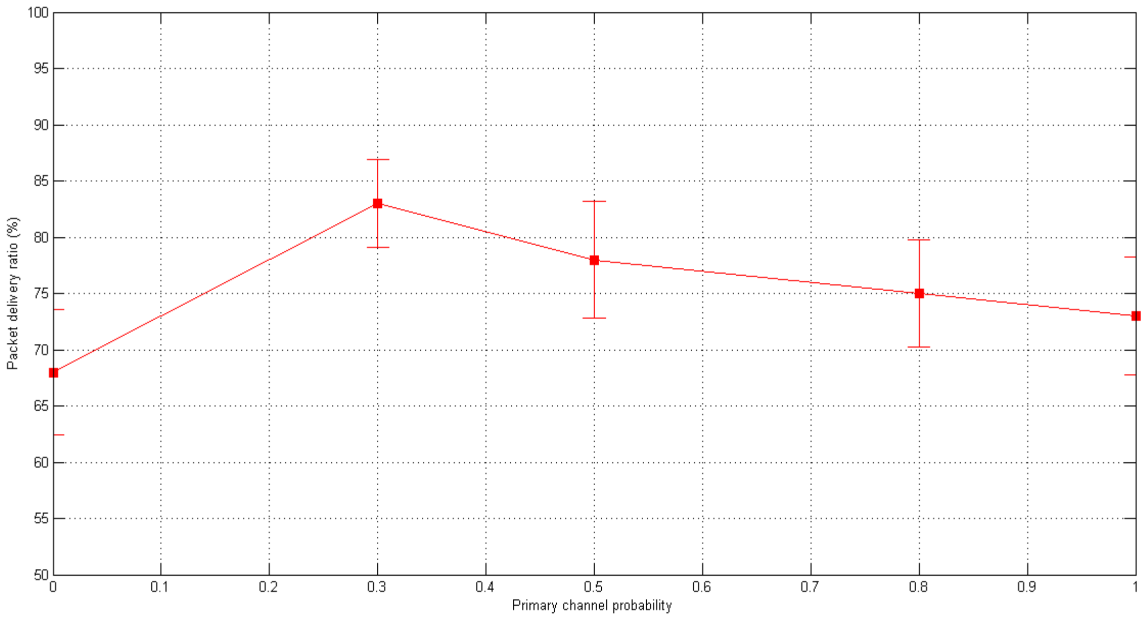


Figure 5.5-1: PDR versus primary channel probabilities.

Table 5.5-1: Missed data acknowledgments for different primary channel probabilities.

Primary channel probability	0	0.3	0.5	0.8	1.0
Number of missed data ACK packets	2	33	50	53	55

5.5.1.2 End-to-end delay versus primary channel probabilities

Figure 5.5-2 shows the average end-to-end packet delay as the primary channel probability increases from 0 to 1. Generally, the end-to-end delay increases as the probability of primary channel availability increases. This behaviour can be attributed to the increase in number of primary channels to be scanned as mentioned when the primary channel probability increases. This results in a longer time required by the nodes to complete channel scanning, which in turn increases the overall end-to-end delay of a data packet. Other possible causes of increasing end-to-end delay such as more frequent route discoveries (and their corresponding latency), congestion delays, and the number of hops traversed by the data packets to their destinations, have been considered but eliminated, as no evidence was found to suggest that the primary channel probability has an impact on these parameters.

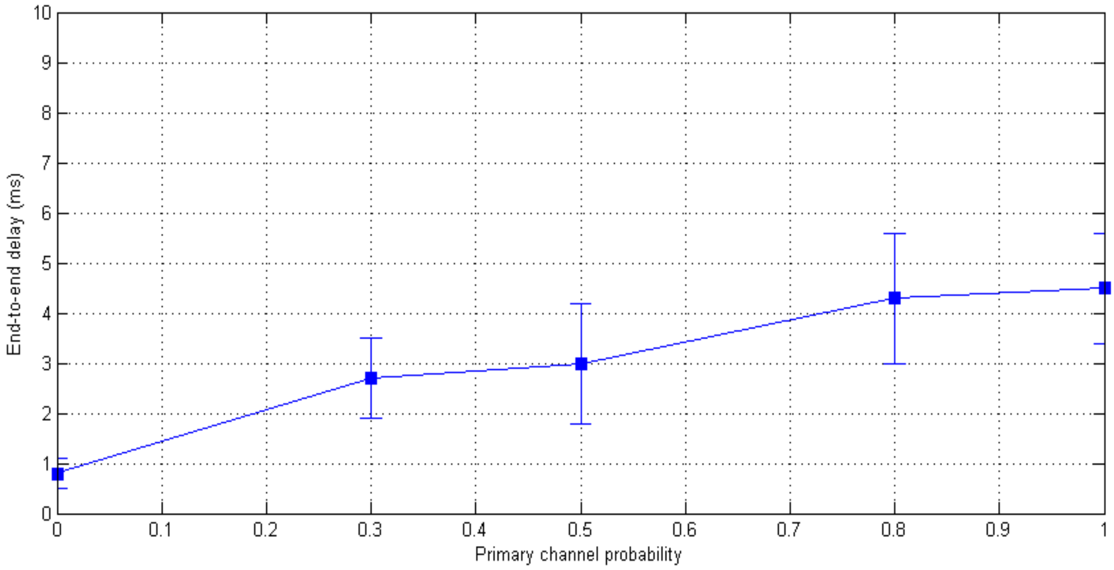


Figure 5.5-2: Delay versus primary channel probabilities.

5.5.1.3 Routing load versus primary channel probabilities

Figure 5.5-3 shows the total routing load along with the distribution of different routing packets which constitute the total load under different primary channel probability. It is observed that the routing loads are generally comparable, which agrees with our earlier hypothesis that the primary channel availability has no significant impact on the routing packets generated. This is expected as routing load of multi-hop data transmissions is primarily a function of node density and mobility, which are unaffected by primary channel availability.

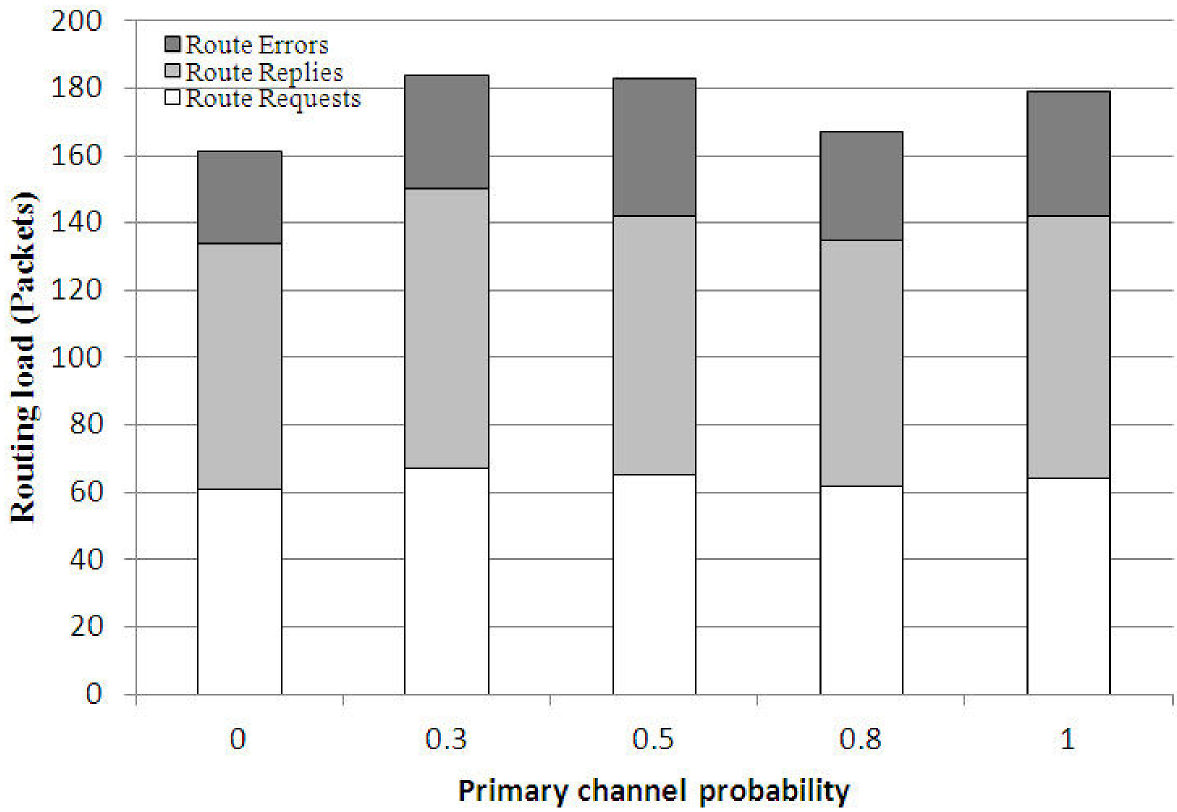


Figure 5.5-3: Routing load versus primary channel probabilities.

5.5.2 Varying node mobility

This section presents the results obtained under different levels of node mobility while keeping the probability of primary channel availability and number of traffic sources to their default values.

5.5.2.1 PDR versus node mobility

Figure 5.5-4 shows the PDR as the node speed increases from 0.0 to 2.0 m/s. Interestingly, it is observed that generally the PDR increases as the node speed increases. This behaviour is due to the difference in usage of WiMax and WLAN interfaces of the multi-interface CR node for packet transmission. It is found that as the node speed increases, the usage of WiMax interface also increases with respect to the usage of WLAN interface (with exception at 2 m/s). This is because when the node speed increases, the routing protocol finds it difficult to find a multi-hop path using WLAN interface and switches to WiMax interface for data transmission via the base station. Hence, less number of data packets is lost due to node mobility while taking a multi-hop path to their destinations. This allows the PDR to increase as node speed increases. However, at 2 m/s the PDR decreases slightly which may be attributed to additional packet lost due to more fragile connectivity between the mobile nodes and the WiMax BS. Table 5.5-2 shows the percentage of data packets transmitted through WiMax and WLAN (White-Fi or Wi-Fi) interfaces for different node mobility.

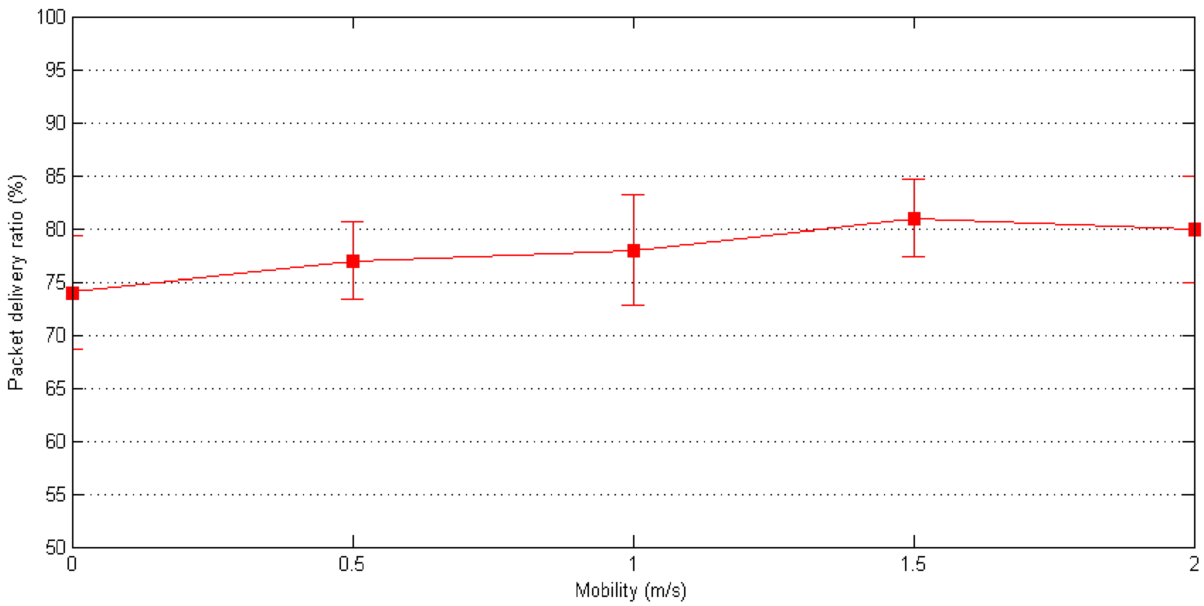


Figure 5.5-4: PDR versus node mobility.

Table 5.5-2: Percentage of data packets transmitted through WiMax and WLAN (White-Fi or Wi-Fi) interfaces for different node mobility.

Node Speeds (m/s)	WiMax interface (%)	WLAN interface (%)
0.0	33.5	66.5
0.5	37.3	62.7
1.0	38.7	61.3
1.5	44.2	55.8
2.0	39.1	60.9

5.5.2.2 End-to-end delay versus node mobility

Figure 5.5-5 shows the end-to-end packet delay as the node speed increases from 0.0 to 2.0 m/s. From the result, no notable trend is observed to suggest that the different node speeds has an impact on the data packet's end-to-end delay.

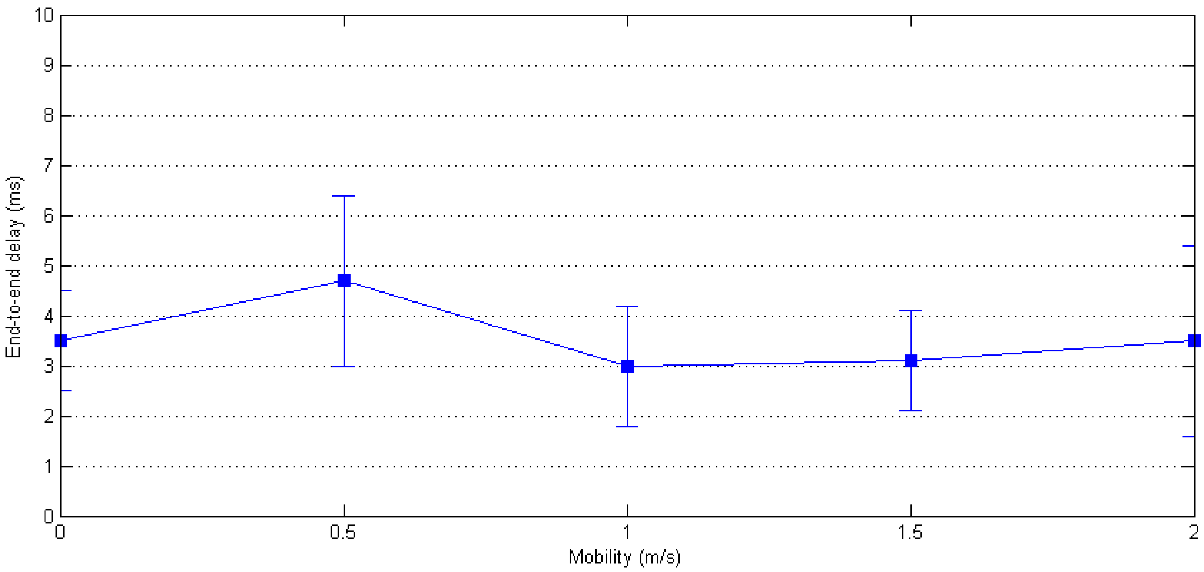


Figure 5.5-5: Delay versus node mobility.

5.5.2.3 Routing load versus node mobility

Figure 5.5-6 shows the total routing load along with the distribution of different routing packets which constitute the total load under different node speeds. It is noted that the routing loads and distributions are generally comparable, which indicate that the node speeds has no significant impact on the amount of routing packets generated. This may again be attributed to the increased usage of the WiMax interface at higher node speeds, which avoids frequent route discoveries by the routing protocol due to increased route failures as mobility increases.

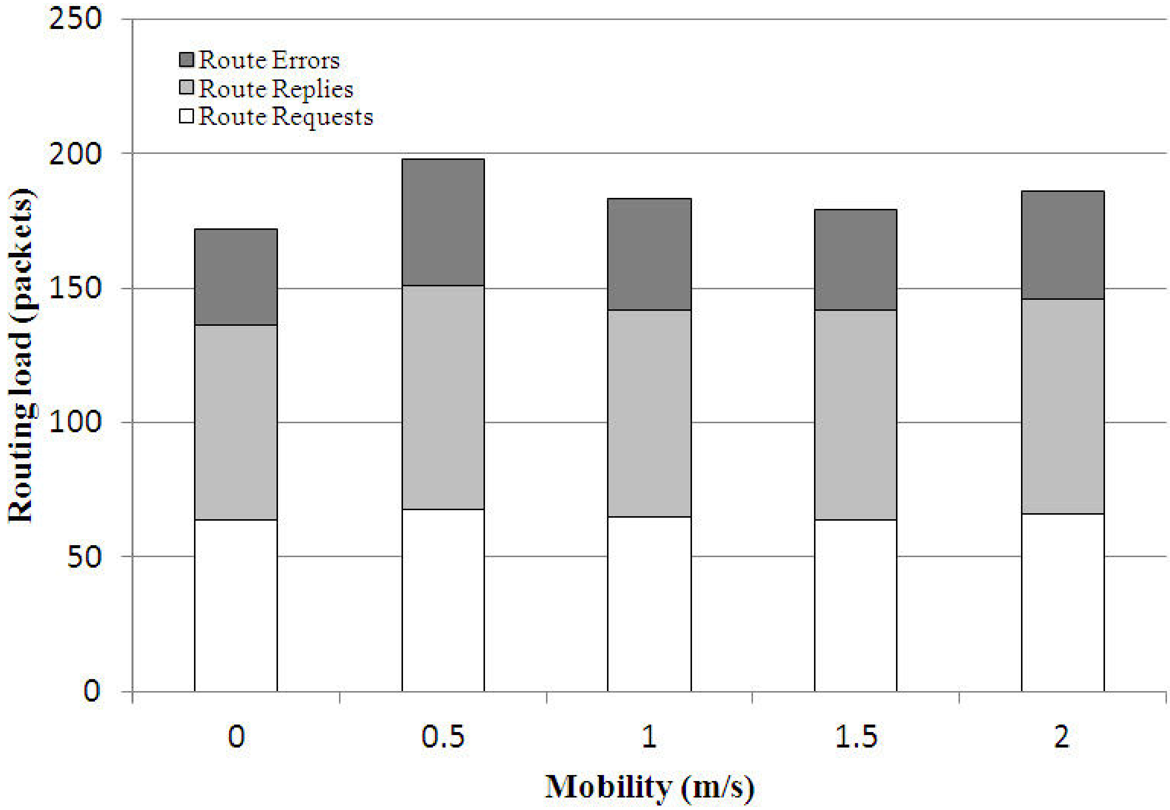


Figure 5.5-6: Routing load versus node mobility.

5.5.3 Varying traffic sources

This section presents the results obtained under different number of traffic sources while keeping the probability of primary channel availability and node speed to their default values.

5.5.3.1 PDR versus traffic sources

Figure 5.5-7 shows the PDR as the number of traffic sources increases from 10 to 30. It can be seen that the PDR decreases gradually as the number of traffic sources increases from 10 to 20, and more substantially when the number of traffic sources increases from 20 to 30.

This is mainly due to network congestion when more data packets are transmitted within the network as the number of traffic sources increases.

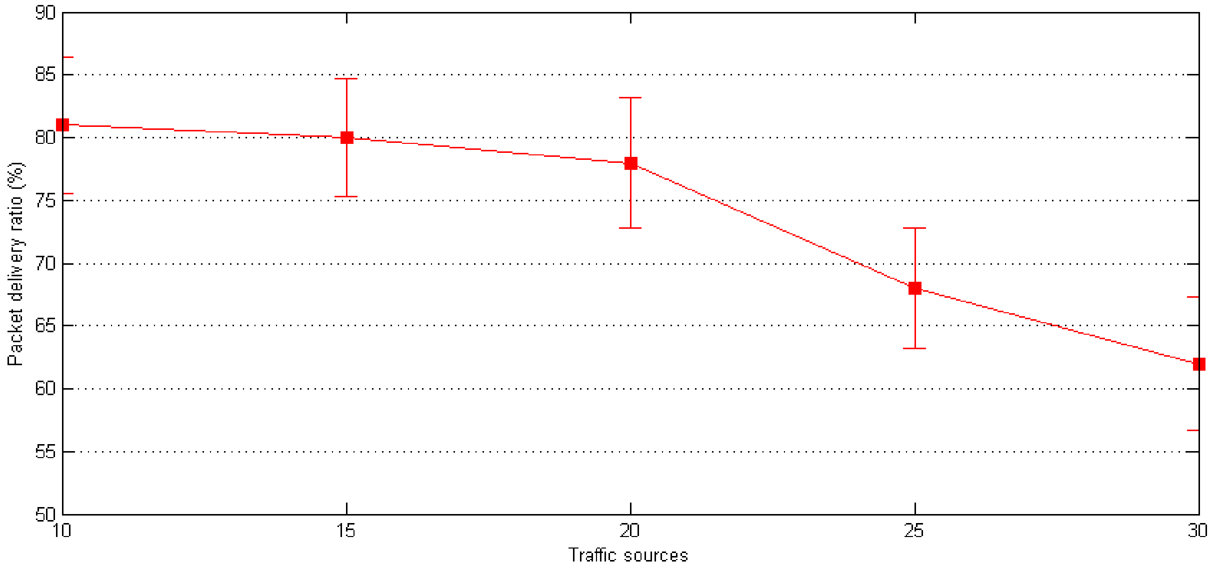


Figure 5.5-7: PDR versus traffic sources.

5.5.3.2 End-to-end delay versus traffic sources

Figure 5.5-8 shows the end-to-end packet delay as the number of traffic sources increases from 10 to 30. It is observed that up to 20 traffic sources, the levels of the end-to-end delays are comparable, but beyond 20 traffic sources, the end-to-end delay increases significantly. This is in part due to increasing network congestion as mentioned earlier and in part due to a higher percentage of data packets transmitted through the WLAN (White-Fi/Wi-Fi) interface as the number of traffic sources increases. The latter is because of a longer time it takes for nodes to receive their RREP through the WiMax interface than the WLAN interface during route discovery due to congestion in WiMax BS as a result of being the single intermediary between all source and destination nodes. In turn, this subjects more data packets to scanning delay when the packets are transmitted through the WLAN interface in White-Fi mode, hence increases their end-to-end delay. Table 5.5-3 shows the percentage of data packets transmitted through WiMax and WLAN (White-Fi or Wi-Fi) interfaces for different traffic sources.

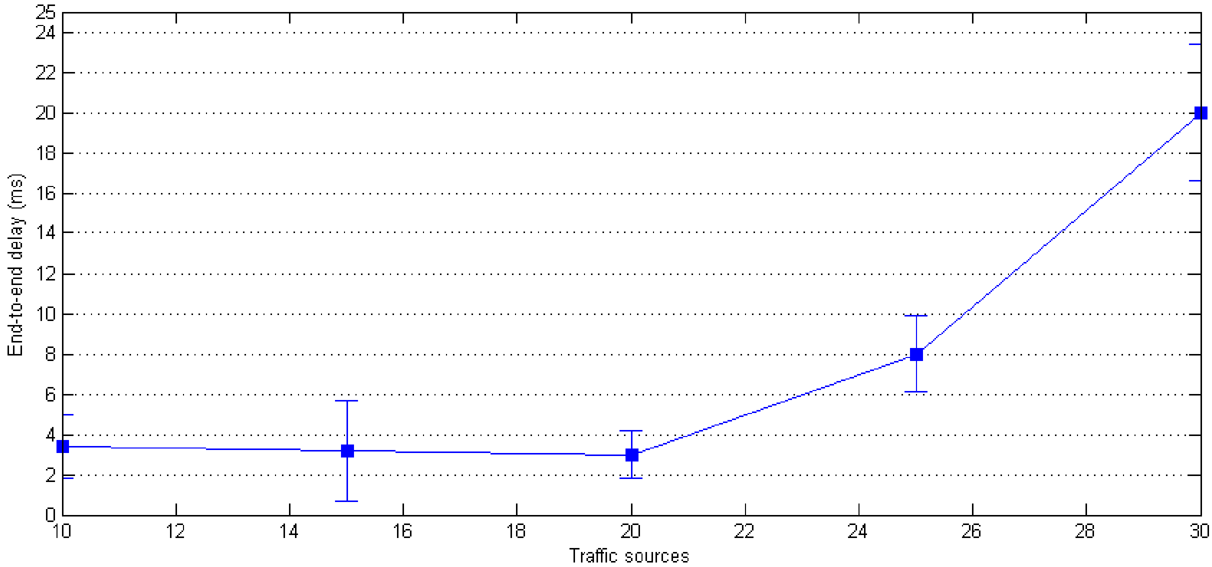


Figure 5.5-8: Delay versus traffic sources.

Table 5.5-3: Percentage of data packets transmitted through WiMax and WLAN (White-Fi or Wi-Fi) interfaces for different traffic sources.

Traffic sources	WiMax interface (%)	WLAN interface (%)
10	32.9	67.1
15	35.5	64.5
20	38.7	61.3
25	24.7	75.3
30	15.5	84.5

5.5.3.3 Routing load versus traffic sources

Figure 5.5-9 shows the total routing load along with the distribution of different routing packets which constitute the total load under different number of traffic sources. The total routing load expectedly increases with increasing number of traffic sources. It is also observed that for 10-20 traffic sources, the routing packets due to route discovery (RREQ and RREP) constitute the largest proportion of the routing load. However, beyond 20 traffic sources, the Route Error (RERR) packets dominate the routing load. This is because as more route discovery packets are transmitted by the traffic sources, the control channel becomes congested, which increases the loss of periodic Hello packets from neighbouring nodes due to packet collisions. This loss of Hello packets is falsely detected as link breakage and causes RERR packets to be generated.

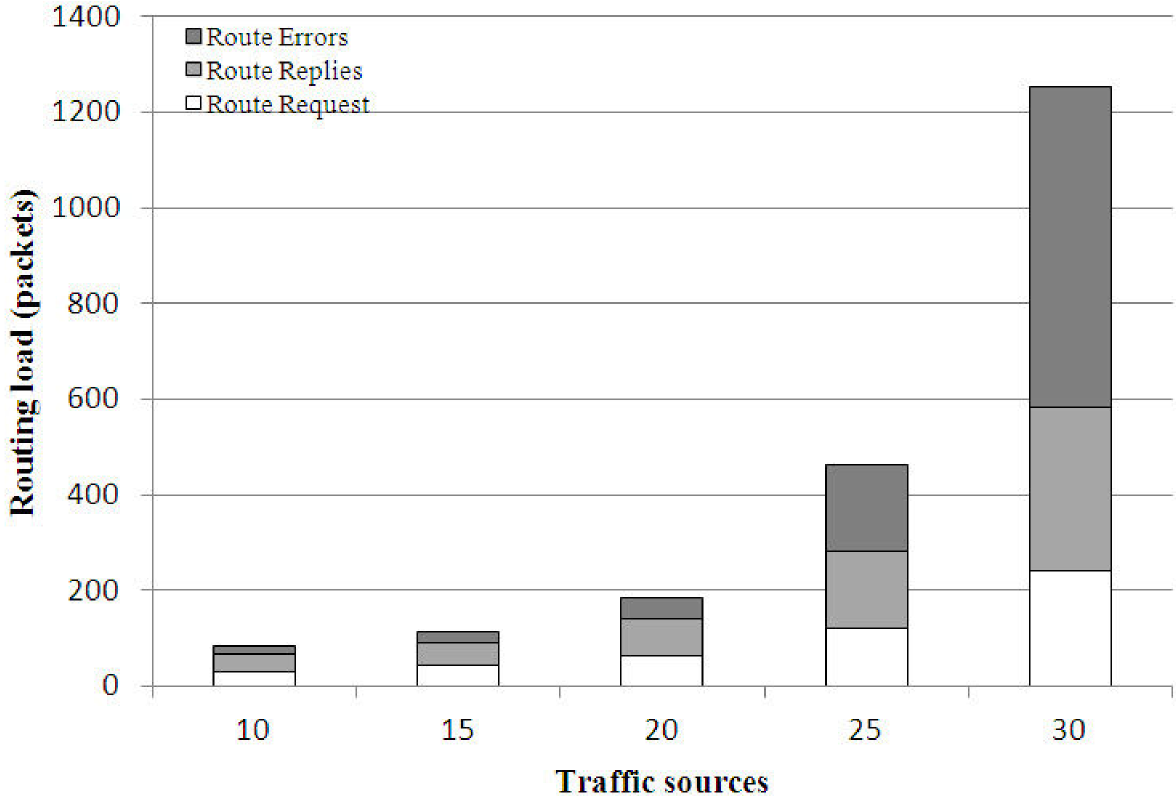


Figure 5.5-9: Routing load versus traffic sources.

5.5.4 Comparison with single interface (WiMax/Wi-Fi/White-Fi) networks

Finally, this section evaluates the performance of three different networks, each consisting of only single interface nodes based on WiMax, Wi-Fi, or White-Fi. The performance of these networks is then compared with that of the multi-interface CR based MCN. The intention is to determine whether the additional complexity introduced by the proposed multi-interface CR node model is justified when set against the performance of simpler networks that use only single-interface nodes. Table 5.5-4 shows the performance of the different networks in terms of the PDR, end-to-end packet delay and routing load. The results are obtained for default values of the experimental parameters as shown in Table 5.3-1.

Table 5.5-4: Comparison with single interface (WiMax/Wi-Fi/White-Fi) networks.

Performance metrics	WiMax network	Wi-Fi network	White-Fi network	Multi–interface CR based MCN
PDR (%)	49	63	67	78
End-to-end delay (ms)	36.56	0.24	1.13	20.87*
Total routing load (packets)	246	224	256	184

* Includes the delay of packets not only from the WLAN (Wi-Fi/White-Fi) interface but also the WiMax interface.

It can be seen that of the single interface networks, White-Fi network provides a better PDR than the other two networks. In terms of end-to-end delay, White-Fi is having higher delay than Wi-Fi due to its requirement for channel scanning before data transmission. The WiMax network appears to be in congestion under the default experimental parameters, resulting in large end-to-end delay and low PDR. This reflects a lower system capacity of the WiMax network as compared to Wi-Fi and White-Fi, which may be attributed to the capacity bottleneck in the WiMax BS where all traffic generated must flow through. On the other hand, through the cognitive use of multiple interfaces and additional spectrum, the multi-interface CR based MCN is found to provide the best performance in terms of PDR and routing load and a moderate delay performance due to the WiMax interface usage for some proportion of the data packets transmitted as shown earlier in Table 5.5-3.

CHAPTER 6 Conclusions and Future Work

6.1 Conclusions

This research develops a multi-interface CR node model in OPNET to enhance the routing performance in MCN. The multiple interfaces used in this node model are the IEEE 802.11a (Wi-Fi), IEEE 802.11af (White-Fi) and IEEE 802.16 (WiMax). The Wi-Fi and White-Fi interfaces are used when a multi-hop path for communication through intermediate nodes is available, whereas the WiMax interface is used for single-hop communication with BS only when there is no multi-hop path available. The White-Fi interface operates in multiple channels and uses the CR feature to perform channel selection depending on the availability of these channels. The selection of the communication interface is performed by the multi-hop routing protocol enabled on this multi-interface CR node model. The multi-hop routing protocol used is a modified version of AODV developed in this research. The route selection of this routing protocol is based on the output interface available (Wi-Fi or White-Fi) when multi-hop communication is possible. Otherwise, the WiMax interface is used. The main investigations in this research are the opportunistic spectrum utilisation and the routing performance of the multi-interface CR node model in MCN.

Simulation methodology is adopted in this thesis. OPNET Modeller simulation tool is used in this research to develop the multi-interface CR node model and to investigate its routing performance in MCN. This node model is developed from the OPNET's WLAN workstation node model and WiMax (IEEE 802.16) SS node model. The MAC layer model of the WLAN interface is modified to incorporate CR spectrum sensing, CR spectrum management, and transceiver synchronisation techniques for multi-channel operations in White-Fi mode. The original AODV routing protocol is also modified in order to be used with the developed multi-interface CR node model. The route selection of this modified protocol is not based on minimum hops as in conventional AODV, but based on the available output interface used for data transmission.

The results of the investigations are presented in Chapter 5. The simulation environment for investigating the performance of the multi-interface CR node model in MCN is a WiMax cellular network. The performance is measured in terms of three different performance metrics: PDR, end-to-end delay, and routing load. The investigations are carried out by varying three experimental parameters: i) probability of primary channel availability; ii) node mobility; and iii) number of traffic sources.

The results for varying the probability of primary channel availability show that PDR can be improved with the use of primary channels, but the extent of improvement has been limited by packet drops at MAC layer due to missed acknowledgements caused by channel scanning. This scanning of primary channels also increases the end-to-end delay of data packets as the probability of primary channel availability increases. The results for varying node mobility interestingly show that better PDR can be achieved at higher node speeds. This is due to an increase in the number of data packets transmitted through the WiMax interface at higher mobility when a multi-hop path for communication cannot be found by the routing protocol. The results for varying the number of traffic sources show that beyond 20 traffic sources, the PDR starts to deteriorate while the end-to-end delay and routing load start to increase due to increasing network congestion. It is also found that the loss of Hello packets due to increased packet collisions can lead to false detection of link breakages, which in turn causes a large number of RERR packets to be generated. Finally, the additional complexity of the proposed multi-interface CR node model is justified by a better overall performance against cases when only single-interface nodes based on WiMax, Wi-Fi, or White-Fi, are used.

6.2 Future work

In this research, the focus has been on developing a multi-interface CR node model which integrates Wi-Fi/White-Fi multi-hop communication with a WiMax based cellular network. A possible direction for future work is to investigate the development of such a model for other emerging mobile communication standards for 4th generation (4G) networks such as Long Term Evolution (LTE) Advanced [31].

From the experimental findings, it has also been found that the channel scanning delay is a limiting factor in the performance of the multi-interface CR node model. This scanning delay has limited the improvement of PDR and increased the end-to-end packet delay. Therefore, another possible direction for future work is to investigate the design of a channel scanning scheme that allows data or control packet transmissions to be interleaved with channel scans, rather than delaying them until the completion of scanning all channels. In this way, less data packets would be dropped due to missed MAC layer acknowledgements, and data packets with delay constraints can be transmitted which improves the end-to-end delay.

References

[1] R. Ananthapadmanabha, B. S. Manoj, and C. S. R. Murthy, "Multi-hop cellular networks: the architecture and routing protocols", *Proceedings of the 12th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications,* San Diego, CA, USA, October 2001.

[2] X. Li, B.-C. Seet, P. H. J. Chong, "Multihop cellular networks: Technology and economics", *Computer Networks*, Vol. 52, No. 9, 2008, pp. 1825-1837.

[3] A. K. Pandey, and H. Fujinoki, "Study of MANET routing protocols by GloMoSim simulator ", *International Journal of Network Management*, Vol. 15, No. 6, 2005, pp. 393-410.

[4] Y.-D. Lin, and Y.-C. Hsu, "Multihop Cellular: A New Architecture for Wireless Communications", in *INFOCOM 2000*. IEEE, 2000.

[5] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, "Next generation/dynamic spectrum access/cognitive radio wireless networks: A survey", *Computer Networks*, Vol. 50, No. 13, 2006, pp. 2127-2159.

[6] L. Khalid, and A. Anpalagan, "Emerging cognitive radio technology: Principles, challenges and opportunities", *Computers Electrical Engineering*, Vol. 36, No. 2, 2010, pp. 358-366.

[7] R. Saeed, "Cognitive Radio and advanced spectrum management", *Proceedings of the Mosharaka International Conference on Communications, Computers and Applications,* Aman, Jordan, August 2008.

[8] J. Mitola III, "Cognitive radio for flexible mobile multimedia communication", *in: Proc. IEEE International Workshop on Mobile Multimedia Communications (MoMuC) 1999*, November 1999, pp. 3–10.

[9] J. Mitola III, Cognitive radio: an integrated agent architecture for software defined radio, Ph.D Thesis, KTH Royal Institute of Technology, 2000.

[10] X. Lin, and Y. Fang, "A survey on the available digital TV bands for cognitive radio operations," *Proceedings of the Fourth International Conference on Communications and Networking in China,* Xi'an, China, August 2009.

[11] P. Bahl, R. Chandra, T. Moscibroda, R. Murty, and M. Welsh, "White space networking with Wi-Fi like connectivity", *SIGCOMM Computer Communication Review*, Vol. 39, No. 4, 2009, pp. 27-38.

[12] IEEE P802.11af, "Wireless LAN in the TV white space," http://www.ieee802.org /11/Reports/tgaf update.htm

[13] W. Guo, and X. Huang, "On throughput enhancement of multi-hop wireless networks using interference alignment", *Proceedings of the 20th Annual Wireless and Optical Communications Conference,* Newark, NJ, USA, April 2011.

[14] X. Hong, K. Xu, and M. Gerla, "Scalable routing protocols for mobile ad hoc networks," *IEEE Network Magazine,* Vol. 16, No.4, 2002, pp. 11-21.

[15] B. R. Arun Kumar, L. C. Reddy, and P. S. Hiremath, "Performance Comparison of Wireless Mobile Ad-Hoc Network Routing Protocols", *International Journal of Computer Science and Network Security*, Vol. 8, No. 6, 2008, pp. 337-343.

[16] M. Timmers, S. Pollin, A. Dejonghe, L. Van der Perre, and F. Catthoor, "A Distributed Multichannel MAC Protocol for Multihop Cognitive Radio Networks", *IEEE Transactions on Vehicular Technology,* Vol. 59, No.1, 2010, pp. 446-459.

[17] Y. R. Kondareddy, and P. Agrawal, "Synchronized MAC Protocol for Multi-Hop Cognitive Radio Networks", *Proceedings of the IEEE International Conference on Communications,* Beijing, China, May 2008.

[18] N. Choi, M. Patel, and S. Venkatesan, "A Full Duplex Multi-channel MAC Protocol for Multi-hop Cognitive Radio Networks", *Proceedings of the 1st International Conference on Cognitive Radio Oriented Wireless Networks and Communications,* Mykonos Island, Greece, June 2006.

[19] H. Ma, L. Zheng, X. Ma, and Y. Luo, "Spectrum Aware Routing for Multi-Hop Cognitive Radio Networks with a Single Transceiver," *Proceedings of the 3rd International Conference on Cognitive Radio Oriented Wireless Networks and Communications,* Singapore, May 2008.

[20] S. M. Kamruzzaman, E. Kim and D. G. Jeong, "Spectrum and Energy Aware Routing Protocol for Cognitive Radio Ad Hoc Networks," *Proceedings of the IEEE International Conference on Communications,* Kyoto, Japan, June 2011.

[21] A. Chehata, W. Ajib, and H. Elbiaze, "An on-demand routing protocol for multi-hop multi-radio multi-channel cognitive radio networks", *Proceedings of the IFIP Wireless Days Conference,* Niagara Falls, ON, Canada, October 2011.

[22] B. Coll-Perales, and J. Gozalvez, "Energy efficient routing protocols for multi-hop cellular networks", *Proceedings of the IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications,* Tokyo, Japan, September 2009.

[23] G. Kannan, Y. Huang, S. N. Bhatti, S. N. Merchant, and U. B. Desai, "A Cross Layer Routing Protocol for Multihop Cellular Networks", *Wireless Personal Communications,* Vol. 51, No. 3, 2009, pp. 427-447.

[24] Z. Li, and H. Shen, "An efficient routing protocol to increase the communication reliability of multi-hop cellular networks", *Proceedings of the IEEE Military Communications Conference,* San Diego, CA, USA, November 2008.

[25] H. Zhuang, Z. Luo, J. Zhang, and H. Yanikomeroglu, "Hierarchical and Adaptive Spectrum Sensing in Cognitive Radio Based Multi-Hop Cellular Networks", *Proceedings of the IEEE 72nd Vehicular Technology Conference,* Ottawa, Canada, September 2010.

[26] Y.-J. Choi, and K. G. Shin, "Opportunistic Access of TV Spectrum Using Cognitive-Radio-Enabled Cellular Networks", *IEEE Transactions on Vehicular Technology,* Vol. 60, No. 8, 2011, pp. 3853-3864.

[27] H. Venkataraman, V. Tomar, D. Gandhi, B. Godara, and H. Haas, "Multihop Multi-band Cognitive Architecture for Next Generation Cellular Networks", *Architecture*, 2010, pp. 1-11.

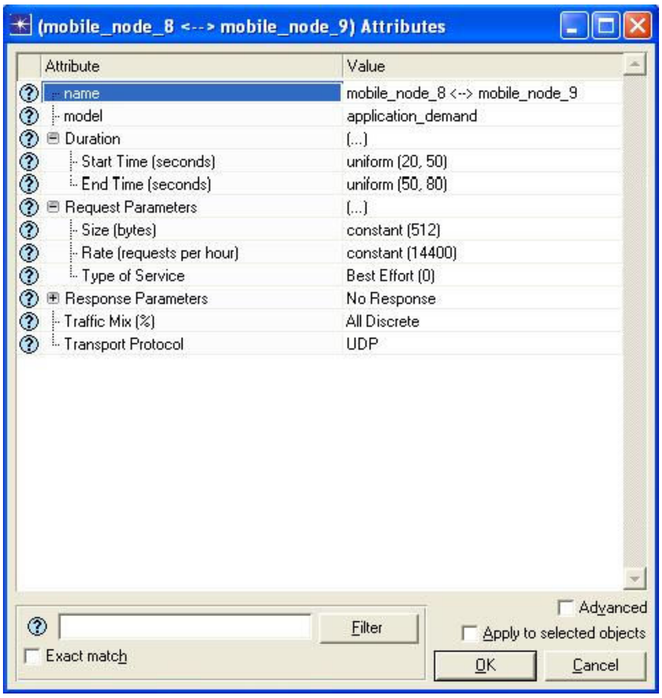
[28] Q. Xin, X. Wang, J. Cao, W. Feng, "Joint Admission Control, Channel Assignment and QoS Routing for Coverage Optimization in Multi-Hop Cognitive Radio Cellular Networks", *Proceedings of the 8th IEEE International Conference on Mobile Adhoc and Sensor Systems,* Valencia, Spain, October 2011.

[29] *OPNET Modeller, Product Documentation Release 16.0*

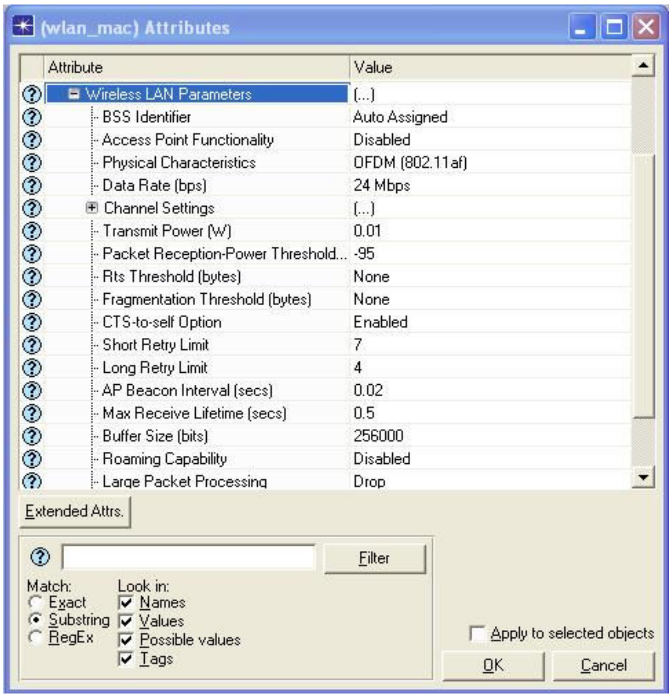
[30] Discussion Paper, *Spectrum Allocation Options*, Radio Spectrum Management Group, September 1998.

[31] M. Barker, "From LTE-Advanced to the Future", *IEEE Communications Magazine*, Special Issue on "LTE-Advanced and 4G Wireless Communications", Vol. 50, No. 2, 2012, pp. 116-120.

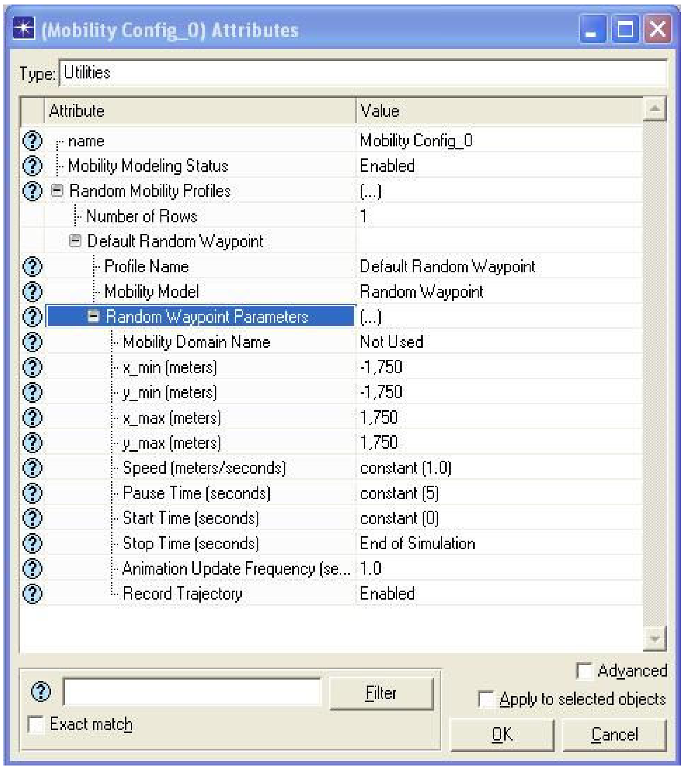
Appendix



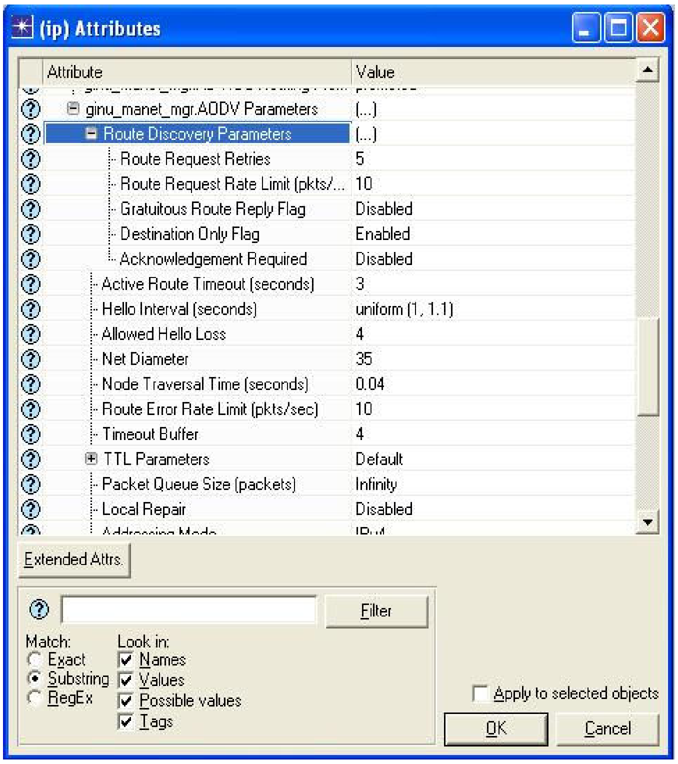
Appendix A: Application demand configuration



Appendix B: Wireless LAN configuration



Appendix C: Mobility configuration



Appendix D: AODV routing protocol configuration