



**DUBLIN CITY UNIVERSITY
SCHOOL OF ELECTRONIC ENGINEERING**

**Development of IEEE 802.16e Functionality
in QualNet**

Stephen Roberts
January 2006

**MASTER OF ENGINEERING
IN
TELECOMMUNICATIONS**

Supervised by Dr. S. Murphy

Acknowledgements

I would like to thank my project supervisor Dr. Sean Murphy for his guidance, enthusiasm and commitment to this project.

Declaration

I hereby declare that, except where otherwise indicated, this document is entirely my own work and has not been submitted in whole or in part to any other university.

Signed:

Date: 10-01-2006

Table Of Contents

| | |
|---|----|
| Chapter 1: Introduction | 1 |
| Chapter 2: Technical Background – The IEEE 802.16 Standards | 4 |
| 2.1 Evolution of the 802.16 standards | 4 |
| 2.2 The IEEE 802.16-2004 standard | 4 |
| 2.2.1 The IEEE 802.16-2004 Physical Layer | 5 |
| 2.2.2 The IEEE 802.16-2004 MAC Layer | 7 |
| 2.2.3 The IEEE 802.16-2004 Frame structure | 10 |
| 2.3 The IEEE 802.16e standard | 13 |
| 2.3.1 Types of Handover | 13 |
| 2.3.1.1 MS Initiated HO | 13 |
| 2.3.1.2 BS Initiated HO | 18 |
| 2.3.1.3 Soft HO | 19 |
| 2.3.2 Sleep Mode | 21 |
| 2.3.3 The IEEE 802.16e Physical Layer | 22 |
| Chapter 3: The QualNet Network Simulator | 23 |
| 3.1 History | 23 |
| 3.2 Operation | 24 |

| | |
|---|----|
| Chapter 4: The Existing QualNet IEEE 802.16 model | 27 |
| 4.1 Structure of Model | 27 |
| 4.2 Operation of protocol | 27 |
| Chapter 5: Required Modifications to implement IEEE 802.16e | 35 |
| 5.1 Design of Model | 35 |
| 5.2 Neighbour Advertisement | 36 |
| 5.3 Negotiating Handover | 38 |
| 5.4 Performing Handover | 39 |
| 5.5 Network Layer Issues | 41 |
| Chapter 6: Testing and Results | 43 |
| 6.1 Interpretations of results | 43 |
| 6.2 Testing procedure and simulation parameters | 43 |
| 6.3 Results | 44 |
| 6.3.1 Traffic Used | 45 |
| 6.3.2 Results of Low Quality Video | 45 |
| 6.3.3 Results of High Quality Video | 48 |
| 6.3.4 Results for VoIP traffic | 50 |
| Chapter 7: Conclusion and further Work | 54 |

List of Figures

| | | |
|------------|----------------------------------|----|
| Figure 2.1 | Architecture of an 802.16 System | 5 |
| Figure 2.2 | Downlink Sub-frame | 11 |
| Figure 2.3 | Uplink Sub-frame | 12 |
| Figure 2.4 | MS initiated HO (in the MS) | 17 |
| Figure 2.5 | MS initiated HO Message exchange | 18 |
| Figure 2.6 | BS initiated HO (in the MS) | 20 |
| Figure 3.1 | QualNet GUI Screenshot | 24 |
| Figure 3.2 | QualNet protocol Stack | 25 |
| Figure 5.1 | Implemented Model | 36 |
| Figure 6.1 | Simulation Set-up | 44 |

Abstract

This assignment looks at the signaling introduced by the IEEE802.16e standard for mobility in Wi-Max systems. The model was developed in the QualNet Network simulator. The system thus developed was tested with an approximation of low quality video, high quality video and voice over IP traffic. It was observed that for low quality video at 384kb/s, packet loss was quite low. For VoIP, packet loss was similarly low but in such a scenario would be more noticeable. For high quality video at 1Mb/s, the data loss was found to be very high, due the higher rate of data generation. The levels of loss here suggest extra measures be taken to prevent this. Routing is clearly an issue and the efficiency with which new routes are found is a concern as is forwarding on packets from an MS' previous serving BS. In the simulations completed, the Handover time was found to range from .319949s to .979931s.

Chapter 1: Introduction

In modern communications, consumers are becoming more and more demanding in terms of the services they require. Multimedia services and high-speed Internet access are now being sought after, in addition to traditional voice services. The provision of these services, by their very nature requires high Bandwidth and a certain guaranteed quality of service. It is natural for such services to be deployed initially over fixed lines.

Technologies such as Asymmetric Digital Subscriber Line, ADSL, address this. The next stage is to look to provide wireless Broadband access to provide the inherent advantages that wireless technologies can offer. More and more, people are choosing wireless devices. Given a wireless situation, it is reasonable to expect mobility.

Before going into detail as to how the IEEE 802.16 Wi-Max systems operate, it is important to differentiate between the IEEE 802.11 Wi-Fi and Wi-Max technologies as there is often confusion between the two. At this moment in time, the dominant wireless technology in Local Area Networks, LAN's, is Wi-Fi. These IEEE 802.11 systems provide large data rates in LAN's over the air. Wi-Max systems, on the other hand, allow for Broadband Wireless Access, BWA, over a much larger range and their capacity is much greater. Studies of 802.16a systems have shown peak data rates of up to 18.36Mb/s are possible with more robust transmission profiles achieving rates of 6Mb/s [1]. The typical range of such systems is 5km for non-Line of Sight operation and 15km for Line of Sight operation, even though up to 50km is achievable in ideal conditions at lower data rates [2].

A common misconception of Wi-Max is that it is simply long range Wi-Fi. Whilst this is one major difference, it is not the sole one. Wi-Max systems are more complex than their Wi-Fi counterparts and thus a Wi-Max Base Station will cost more than its equivalent Wi-Fi node. This is a trade-off for the fact that Wi-Max systems can provide greater range and greater capacity. In addition, Wi-Max has more provision for Quality of Service. In 802.11 based networks, contention algorithms are more distributed, whereas

802.16 systems are more Base Station controlled which allows easier system administration, thus facilitating easier scaling of the networks.

When looking at possible applications of such systems, it soon becomes apparent that Wi-Max and Wi-Fi systems can compliment each other. The three main deployments of these BWA networks appear to be backhaul, last-mile and large-area coverage. Whilst Wi-Max systems appear to have the advantage in large-area coverage, as in Metropolitan Area Networks (MAN's), it seems reasonable to assume that Wi-Max systems may be used as a backhaul network to serve a series of Wi-Fi LAN's. Another application of Wi-Max may be in the backhaul networks of 3G networks. The reason for the interest in Wi-Max as backhaul systems is due to their large capacities, but also an economic one. Wi-Max solutions will be cheaper and more flexible than their wired counterparts. Studies comparing IEEE 802.16e systems to 3G networks have shown the downlink throughput to be largely similar. On top of this, it appears 802.16e systems are 40%-50% more spectrally efficient [3]. However, it is unlikely that Wi-Max systems will challenge 3G networks due to the large amount of investment mobile operators already have in this technology.

At this moment in time, it is still early in the development of Wi-Max technologies. The IEEE 802.16 standard was only initially published in 2001 with several required revisions to follow in recent times. Due to the higher cost of Wi-Max systems, it is not anticipated to challenge the dominance of Wi-Fi in the serving of small LAN's.

The IEEE 802.16e standard addresses the requirement of mobility in Wi-Max systems. The standard aims to support mobility up to vehicular speeds. IEEE 802.16e builds on previous Wi-Max standards, thus allowing the upgrade from Fixed Broadband Wireless Access, FBWA, systems to mobile service provision.

As mentioned above, these systems are at a very early stage in their development and as such, the potential performance is not fully known. This project aims to look at the issues

raised in trying to introduce mobility into an existing model of the IEEE 802.16a standard.

The IEEE standards, which will be discussed in further detail in the next chapter, specify a MAC layer and physical layer. This project will focus on developing a model of the MAC layer functionality. By introducing this into the existing model, the results should indicate the impact the additions shall have on different kinds of traffic.

The IEEE 802.16e standard has been ratified as of December 2005. However, it is important to note that during the implementation of this project, the standard was in the drafting stage and so draft 9 was used.

The structure of this report is as follows. In the next chapter, the first thing to be looked at is the evolution of the 802.16 standards. In addition to this, the operation of the IEEE 802.16-2004 standard and the amendments made by the IEEE 802.16e standard are explained. The next chapter focuses on explaining the existing model of 802.16a, and after this, modifications made to the existing model to implement 802.16e are examined. The penultimate chapter outlines the testing and results and the final chapter is the conclusion.

Chapter 2: Technical Background - The IEEE 802.16 Standards

2.1 Evolution of the 802.16 standards

In October 2001, the IEEE 802.16 standard was published. This defines the air interface and Medium Access Control (MAC) protocols for supplying Broadband Wireless Access in a metropolitan area network. This initial standard specifies the physical layer for frequencies 10 to 66 GHz. The main problem with this is that at such short wavelengths it is required to have a line of sight between the Base station and Subscriber stations. This introduces difficult deployment issues. This issue was addressed in the IEEE 802.16a standard, which was published in January 2003. This introduced provision for the physical layer to operate in the 2 to 11 GHz band. This, together with OFDM technology and sub-channelization allows for Non-Line Of Sight, NLOS, operation [4].

Following this, the 802.16-2004 standard was published, which was basically an amalgamation of 802.16 and 802.16a. The 802.16e standard builds on the previous ones and introduces support for mobile subscriber stations traveling at vehicular speeds. As such, it specifies necessary amendments in the MAC and the PHY layer to allow for this. This project is concerned with the amendments to the MAC layer in order to provide handover.

2.2 The IEEE 802.16-2004 Standard

This architecture of an 802.16 system can be seen on the following page. The standard specifies two modes of operation, point to multipoint and mesh mode. In point to multipoint, PMP, operation, traffic only occurs between the Base Station, BS, and the SS. Mesh mode allows subscriber stations, SS, to communicate directly with each other and

thus traffic can be routed through the subscriber stations. At this moment in time, there is more interest in PMP operation as it is anticipated that in a commercial environment, clients would be reluctant to allow others use their own resources, such as power. However, in the context of a backbone network, where the network provider is in full control of all nodes as far out as the base-station, mesh-mode may be of interest as the distribution of all resources would be at the discretion of the network provider.

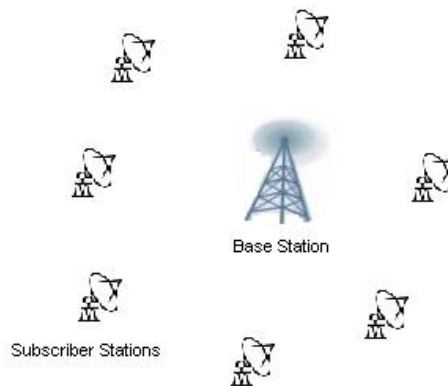


Figure 2.1 Architecture of an 802.16 System

2.2.1 IEEE 802.16-2004 Physical Layer

Whilst this project is concerned with the MAC layer, it is necessary to have some knowledge of the operation of the physical layer, as the two are not entirely independent of each other.

Two modes of duplexing are supported, namely Frequency Division Duplexing, FDD, and Time Division Duplexing, TDD. In FDD operation, the uplink and downlink channels are transmitted at different frequencies. Whilst this set up provides a full duplex scenario, it is important to still maintain support for those SS which only support half

duplex operation, and so for those subscriber stations, the BS must ensure it does not schedule such SS to transmit and receive at the same time.

For TDD operation, the downlink and uplink channel operate at the same frequency, but are temporally separated. The end of the downlink sub-frame and the start of the uplink sub-frame is dynamic and can change from frame to frame. TDD is more demanding on the antenna, as it has to switch from transmit mode to receive mode once every frame, which can have a length of .5ms up to 20ms, depending on the PHY being used.

The physical layer specifications can be divided into two groups: that for operation in the 10-66 GHz range, and that for operation in the 2-11 GHz range.

Operation in 10-66 GHz range

In the IEEE 802.16-2004 standard, the PHY specification for this range is referred to as the WirelessMAN-SC, the SC standing for single carrier modulation. This was chosen as at such high frequencies, it was assumed that Line of Sight was necessary. The BS transmits a Time Division Multiplexing, TDM, signal and provides access by means of Time Division Multiple Access, TDMA. In TDMA, symbols are transmitted in non-overlapping time slots and each client gets access to a given time slot.

Operation in 2-11 GHz range

Given that in a metropolitan area, NLOS operation was deemed important, the IEEE 802.16a standard introduced operation in the 2-11 GHz range. There are three schemes specified in the IEEE 802.16-2004 standard: Single carrier, Orthogonal Frequency Division Multiplexing, OFDM, and Orthogonal Frequency Division Multiple Access, OFDMA.

OFDM takes a block of serial symbols and transmits them in parallel using carriers, which are mathematically orthogonal. Say we have N symbols to be transmitted serially,

with period T_s . If we take those N symbols and transmit them on N parallel orthogonal carriers, each with period T , we in effect reduce the symbol rate on each sub-carrier, as we have $T = NT_s$. This reduction in symbol rate means that the effect of delay spread is reduced, and so there is no need for equalization at the receiver [5]. This is quite useful in an environment where we will get multipath propagation, as we do with NLOS operation [6]. OFDMA works by assigning a certain subset of the sub-carriers to each client to provide multiple-access. It is anticipated there will be future interest in OFDMA but currently, OFDM is receiving most attention.

Adaptive Modulation

The physical layer also provides adaptive modulation. This means that transmission parameters (modulation scheme, channel coding and FEC settings) can be modified on a frame by frame basis for each SS. Therefore, an SS who is far away from the BS and may see poor channel conditions will use more robust transmission parameters than a SS who is closer to the BS. The parameters a SS uses are known as its burst profile. This burst profile will have been negotiated between the BS and the SS. The modulation schemes provided are 64-QAM, 16-QAM and QPSK, in order of decreasing efficiency and increasing robustness. For example, the use of 64-QAM will require a better signal to noise ratio than 16-QAM, but will provide more bits per symbol.

2.2.2 IEEE 802.16-2004 MAC Layer

The MAC layer consists of three sublayers, the Service Specific Convergence Sublayer, the Common Part Sublayer, and the Security Sublayer.

Service Specific Convergence Sublayer

The IEEE 802.16 standards are very much connection oriented. Each service is assigned a connection with a varying level of quality of service, QoS. There are three management connections, coupled with transport connections for transmission of data. The purpose of

the Service Specific Sublayer, is to accept higher layer PDUs and classify them. Based on this classification, they are assigned to a given connection, with a given QoS. The Service Specific Sublayer supports ATM services as well as packet based IP services.

Common Part Sublayer

The purpose of the Common Part Sublayer is to facilitate the sharing of the transmission medium. The system works on a PMP basis, whereby every station hears the same downlink transmission (or at least the part of it they are able to, given they may not be capable of hearing certain burst profiles). The BS grants access to the uplink. This access is very much driven by the class of service and the QoS provisions such a service requires.

MAC Connections

All service flows are assigned to a connection. A connection is a unidirectional flow of data, with an assigned QoS. There are three connections defined for management purposes: These are basic, primary, and secondary connections. The purpose of the basic connection is for short, more urgent MAC messages, which are not very delay tolerant. It is also used for Radio Link Control, RLC, messages. The purpose of these RLC messages is to control power and ranging, in addition to changing burst profiles [7]. The primary connection is responsible for longer management messages, and its principal use is in the security sublayer. The function of the secondary connection is to allow a particular protocol being run at a higher layer, such as the routing protocol, to exchange their management messages. In addition to these there is the transport connection whose purpose is for the transfer of data and the Broadcast connection for the transmission of broadcast MAC messages such as the Neighbour Advertisement message and the MAC messages that deal with channel access. Finally, there is an initial ranging connection, used by SS when initially registering to a BS.

Quality of Service

It is important to note that the IEEE 802.16 standards provide a lot of support for implementation of QoS; indeed this is one of the main selling points of Wi-Max technology. However, this is outside the scope of this project, and so will not be addressed.

MAC Management Messages

In total, there are 49 MAC management messages. These deal with issues such as channel access, channel condition description, BS registration and security among others. They will use either the Broadcast, Basic or Primary connections. The most important ones as regards this project are illustrated below. The frame control section messages will be discussed in the section on frame structure.

RNG-REQ, Ranging Request (Transmitted on Basic or Initial Ranging connection)

This message is transmitted periodically to determine network delay and to request a power or burst profile change.

RNG-RSP, Ranging Response (Transmitted on Basic or Initial Ranging connection)

This message contains the response to the RNG-REQ message, containing such parameters that are relevant to the request. If the RNG-REQ was sent on the initial ranging connection, this message then also contains the connection identifier for the basic and primary connection assigned to this SS.

REG-REQ, Registration Request (Transmitted on Primary connection)

Sent by a SS in order to register with the BS and to negotiate the SS's capabilities, such as its IP version, ARQ parameters and convergence layer capabilities.

REG-RSP, Registration Response (Transmitted on Primary connection)

Sent by BS in response to a REG-REQ. It contains the connection identifier for the SS's secondary connection, in addition to relevant parameters for the SS's capabilities.

DSA-REQ, Dynamic Service Addition Request (Transmitted on Primary connection)

Sent by SS to create a new service flow.

DSA-REQ, Dynamic Service Addition Response (Transmitted on Primary connection)

Sent by BS in response to a DSA-REQ. It contains the connection identifier of the transport connection assigned to this service flow, if this has been approved.

2.2.3 IEEE 802.16-2004 Frame structure

In the simulation model implemented here, TDD is used, and so the frame structure shall be discussed in this context. Each frame is a fixed length, and is divided into first the Downlink Sub-frame and then the Uplink Sub-frame. The division between these two sub-frames is dynamic.

Downlink Sub-frame

In a given frame, the downlink sub-frame occurs first and its make up can be seen in the figure below.

| | | | | | | |
|----------|-----------------------|-------------------------------|-------------------------------|------|------|-------------------------------|
| Preamble | Frame Control Section | Data For SS x Burst Profile x | Data For SS x Burst Profile x | | | Data For SS x Burst Profile x |
|----------|-----------------------|-------------------------------|-------------------------------|------|------|-------------------------------|

Figure 2.2 Downlink Sub-frame

As can be seen, first there is a preamble. The purpose of this is to enable the SS to synchronise to the frame, as it may have lost synchronization at the end of the previous frame. Following this preamble, there is a frame control section. The following messages may be present here, DL-MAP, UL-MAP, DCD and UCD. The DCD and UCD get transmitted at a given interval, whereas the DL-MAP and UL-MAP must be present at the start of every frame.

Downlink-MAP, DL-MAP:

The purpose of the downlink map is to define the access to the downlink information. It tells each SS at what time changes in modulation and coding will occur. The subscriber station listens to all the data it is able to until it reaches data for itself.

Uplink-MAP, UL-MAP:

The purpose of the UL-MAP is to define access to the uplink channel. It specifies a burst profile and a time for each SS to transmit its data.

Downlink Channel Descriptor, DCD:

The purpose of the DCD is to specify the characteristics of a given downlink physical channel. The BS transmits it at a given interval, which can have a maximum value of 10 seconds.

Uplink Channel Descriptor, UCD:

The purpose of this is similar to the DCD, in that it specifies the characteristics of the uplink channel and is transmitted at a given interval, which cannot exceed 10 seconds. This message also contains required information for initial ranging contention.

Following this frame control section, the BS transmits its TDM data. This data is transmitted in order of decreasing robustness, in terms of the burst profile. The reason for this is so that a SS does not lose synchronisation. For example, if we transmitted the least robust profile first, the SS waiting for a more robust burst profile would lose synchronisation, and so have to wait until the start of the next frame to resynchronise, thus missing the data that was addressed it.

Uplink Sub-frame structure

The structure of the uplink sub-frame is more straightforward and can be seen below.

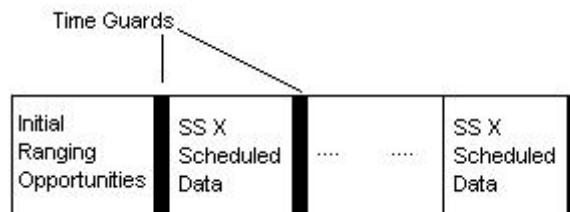


Figure 2.3 Uplink Sub-Frame

Firstly we have initial ranging opportunities. This is to allow any SS, which have not already registered to send a ranging request message. The purpose of this message for a SS, which is not registered, is to discover network delay and to negotiate a burst profile or power change. Because this is the first communication this SS is having with the BS, there are no pre defined uplink opportunities. All SS's, without such uplink opportunities,

transmit their range request in this initial ranging opportunity period. Contention is dealt with using the Truncated Binary Exponential Backoff algorithm, the parameters of which are transmitted in the UCD.

Following this initial ranging opportunity, the SS's transmit their data via TDMA, in the time slot provided to them by the UL-MAP.

It is also worth noting that in addition to the aspects discussed above, there also exists a security sublayer, to facilitate authentication and encryption. This means that each SS cannot listen to traffic that is not addressed to it.

2.3 The IEEE 802.16e Standard

The section of the report will focus on amendments made to allow for mobility. The standard defines Mobile Stations, MS's. It also defines HandOver, HO, to be the process in which, "an MS migrates from the air interface provided by one BS to the air interface provided by another BS" [8]. The best way to look at the amendments to the MAC layer, is to follow through the HO process for various scenarios.

2.3.1 Types of Handover

In an attempt to cover all eventualities, provision has been made for several different types of HO.

2.3.1.1 MS Initiated HO

When an MS is losing signal quality with its serving BS it may wish to move to a new BS. This is inherently linked with mobility and may be used when an MS is moving out of the coverage of its serving BS. Another scenario is that the MS may wish to move if it

can get a higher QoS at another BS. In such situations, the MS will initiate HO. As this will be the most common type of HO, it is the one implemented in simulation. It is important to note the methodology of deciding whether or not to perform HO is not included in the standard. The standard provides a means to gain information about neighbouring stations in order to make this decision.

Neighbour Advertisement

BS's should be capable of transmitting a Mobile Neighbour Advertisement, MOB_NBR-ADV message at a given interval. The purpose of this message is to inform all listening stations of the characteristics of any neighbouring base stations, which they may or may not wish to move to. It contains information such as the frequency of this BS, its BS identifier, the types of services it supports and its available radio resources (available channels etc.). This message is transmitted on the broadcast connection.

An important parameter of this message is the frequency with which it is sent. The standard specifies a maximum of 1s. This is of relevance because if we have a mobile station moving at large speed, it may be moving through the range of coverage of each BS quicker than it receives the MOB_NBR-ADV message. In this case, it would be unable to perform HO. This parameter is thus one limit to the speed with which an MS can move through the network. The network administrator should decide on a value which is regular enough to facilitate all users, and yet not too frequent to disrupt efficiency.

Neighbour Scanning

To facilitate a HO decision, the MS may perform scanning on possible target BS's. To do so, it first sends its serving BS a MOB_SCN-REQ message. This message tells the BS that it wishes to scan its neighbours. It specifies a length of time in frames for this interval and the type of association it will scan its neighbours with. There are three levels of association:

- *Without Coordination*: Target BS has no knowledge of approaching MS
- *With Coordination*: Serving BS coordinates association with requested target BS's.
- *Network Assisted Association Reporting*: Similar to previous method, with the exception that the RNG-RSP message is sent over the network backbone to the serving BS. The serving BS then puts all received RNG-RSP messages (from all BS that were scanned) and transmits them to the MS in the form of an MOB_ASC_REPORT message.

The BS will then respond with a MOB_SCN-RSP message, which indicates the length of the approved scan and the association type for the scanning. The MS is now free to scan its neighbours. It can synchronise with a given BS's downlink transmissions and estimate the quality of the physical channel.

Negotiating Handover with Serving BS

The next stage in the HO process is to send a MOB_MSHO-REQ message to the serving BS. This message is transmitted on the basic connection and contains a list of stations the MS recommends as targets. This list is generated by the results of scanning. On receipt of this message, the serving BS sends a HO-pre-notification message to all BS's on this list. It should then receive a HO-pre-notification-response from these stations. From these responses, the serving BS generates a list of target stations, which it sends to the MS in the MOB_BSHO-RSP message, also transmitted on the basic connection. Upon receipt of the MOB_BSHO-RSP message, the MS may perform HO. The serving BS is informed via a MOB_HO-IND message, transmitted on the basic connection. The MS may also cancel the HO in this message. If it is continuing with HO, it tells the BS that it is leaving that BS and it tells it the parameters of the target BS.

The MS is now ready to register with the target BS and does this in the exact same way as outlined in the previous standards. The two diagrams on the following page illustrate the process discussed above.

It is also possible for the target BS, on receipt of a HO-pre-notification message, to include in its UL-MAP a fast ranging Information Element, IE, which provides the MS undergoing HO, a non-contention based initial ranging opportunity. This is to decrease the overall speed of the HO process.

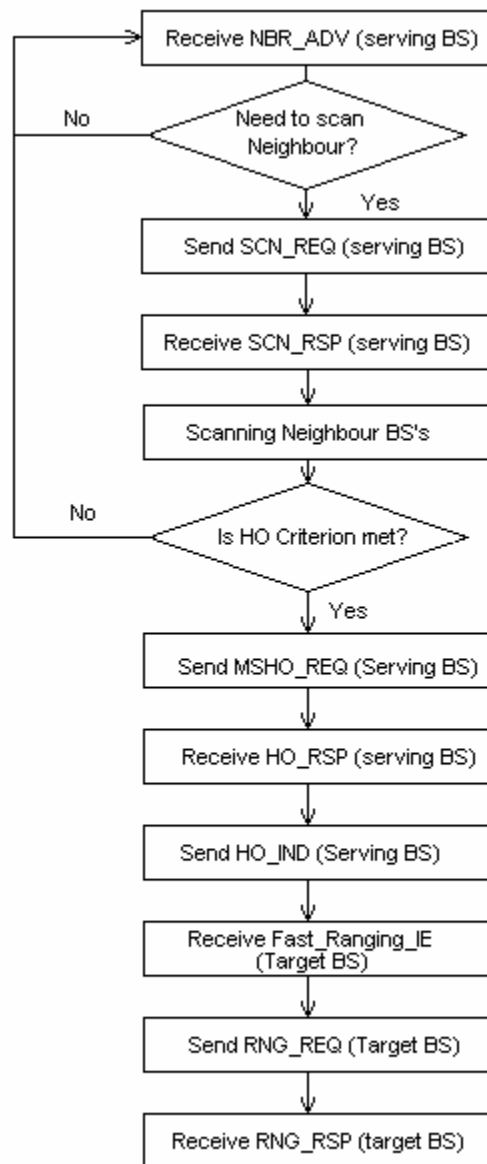


Figure 2.4 MS initiated HO (in the MS)

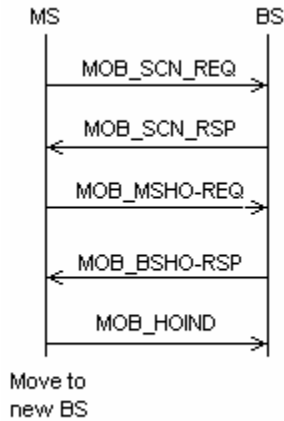


Figure 2.5 MS initiated HO Message exchange

2.3.1.2 BS Initiated HO

The next case to look at is when the BS decides that the MS should move to another BS. This may be required if the serving BS decides it can no longer provide its client with the required QoS. It may also be used if the BS decides that the MS is moving out of range. These are similar examples as in the case where the MS initiates HO. However, it may be useful to have the BS make these decisions so as to centralise the HO procedure. This is because mobile users may have limited power resources and so any computation, which can be performed by the BS, is welcome.

Assuming that the MS has received the MOB_NBR-ADV message as described in the previous section, the first step here is for the BS to send the MS a MOB_BSHO-REQ message, to tell the MS it needs to perform HO and so to start scanning its neighbours. This message contains the BS's list of recommended target stations together with such relevant information as service level predictions and channel details. This message is transmitted on the basic connection.

On receipt of the MOB_BSHO-REQ message, the MS may scan the BS's detailed in this message. It then sends a MOB_BSHO-RSP to the serving BS, with its list of

recommended stations. From here, the HO process is the same as for the MS initiated case. The process is illustrated in the flow chart on the next page.

2.3.1.3 Soft HO

The last type of HO to be looked at is soft HO, SHO. The idea behind this is that the MS can receive the same MAC/PHY protocol data units from one or more BS's. This is achieved by using diversity combining at the antenna. This basically involves exploiting the different spatial, pattern and polarization diversity viewed by multiple antennas [9]. The major advantage of this is that whilst performing HO, an MS can continue to receive data. This is quite important in the context of real time traffic such as Voice over IP, VoIP. However, the trade-off is that as multiple antennas are required, the systems are more costly and more complex.

For a given MS supporting SHO, each BS keeps a list of BS's involved in SHO with the MS. This list is known as the active set. Within this active set, there is an anchor BS. The MS may monitor the anchor stations downlink control information alone, or that of all stations in the active set. If the MS is only monitoring the anchor BS, the anchor station should then contain burst allocation information for the other stations in the active set in its DL-MAP and UL-MAP.

Otherwise, the MS can monitor all stations in the active set. All BS's involved in SHO with the MS shall use the same connection identifiers and must have a synchronized frame structure.

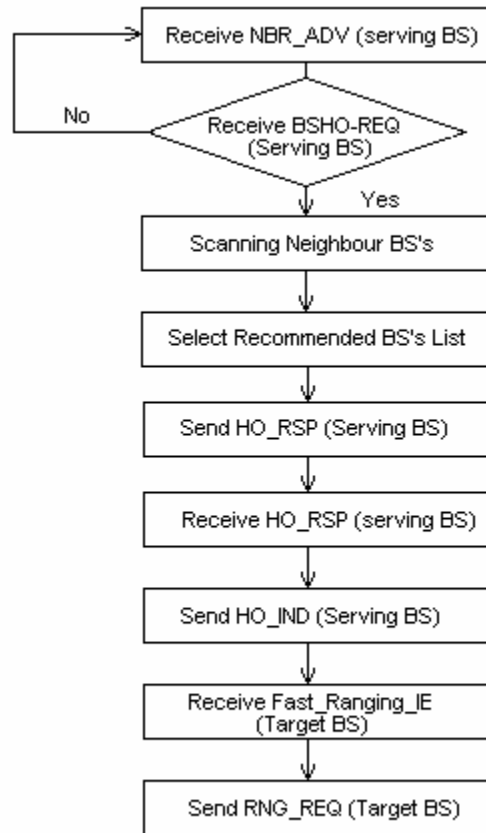


Figure 2.6 BS initiated HO (in the MS)

2.3.2 Sleep Mode

One final amendment worth noting is that of sleep mode. When mobility is introduced to the equation, clearly power becomes a concern. If a user is mobile, they may not be able to recharge their unit for quite some time, and so it is important to look at ways in which power can be conserved, much more so than in a fixed setting.

The IEEE 802.16e defines a new mode the MS can occupy, sleep mode. In sleep mode, the MS is no longer receiving any traffic from the BS and so it can set its power usage to the minimum. To enter sleep mode, the MS sends a MOB_SLP-REQ to the BS, requesting to enter sleep mode for a given duration. The BS may or may not approve, and does so in the form of a MOB_SLP-RSP message. Initially, the MS may go into sleep mode for a requested period of time, T . When this has expired, the MS leaves sleep mode and has a short listening interval to see if there is any traffic from the BS for it. If not, it resumes sleep mode, this time for an interval twice as long. This continues until the MS reaches a sleep interval a T_{\max} . Therefore, the j^{th} sleep interval will be $2^j T$, or T_{\max} , whichever is smaller. T_{\max} is quite an important value. It has been shown that the larger this is, the more energy will be saved. However the trade off of this is that frame response time (the length of time it takes for an MS to respond to a frame destined for it) increases [10].

2.3.3 IEEE 802.16e Physical Layer

Whilst the majority of amendments have been made to the MAC layer, there have also been changes made to the physical layer. Though this project is concerned with the MAC layer, it is worth noting that the changes made to the physical layer will have applications in both fixed and mobile applications.

The main changes have been made to the OFDMA PHY. It is now being referred to as Scalable OFDMA, SOFDMA, and the channels used will now be scalable. This improves the multi-access capabilities of the previous OFDMA PHY. A major part of this SOFDMA is that the use of larger Discrete Fourier Transform sizes makes it more resistant to delay spread and so more resistant to multipath fading, which is of concern in NLOS operation.

Chapter 3: The QualNet Network Simulator

3.1 History

Qualnet is a commercial spin-off from the GloMoSim simulator, which was developed at the University of California, Los Angeles, UCLA, and is distributed by Scalable Network Technologies.

The simulator itself is C++ based. All protocols are implemented in a series of C++ files, and called by the simulation kernel. One of the major selling points of QualNet is that it is very scalable. No efficiency is lost in running large simulations of many thousands of nodes with heavy traffic.

Whilst it is possible to run simulations completely from the command line, QualNet comes with a java based graphical user interface. A screenshot of this can be seen on the next page. This allows for easy, more intuitive setup of simulations as well as allowing easy observation of results. During simulation runtime, it allows the user to observe the signals being transmitted and received at each node, which aids in the understanding of what is physically happening.

The three main programs used in QualNet are the simulator, the analyser and the packet tracer. The simulator runs the given simulation, the analyser displays the results and the packet tracer allows us to follow the path of a packet through the network. There is also a protocol developer in the GUI, however, it is much more powerful to develop protocols from C++ coding.

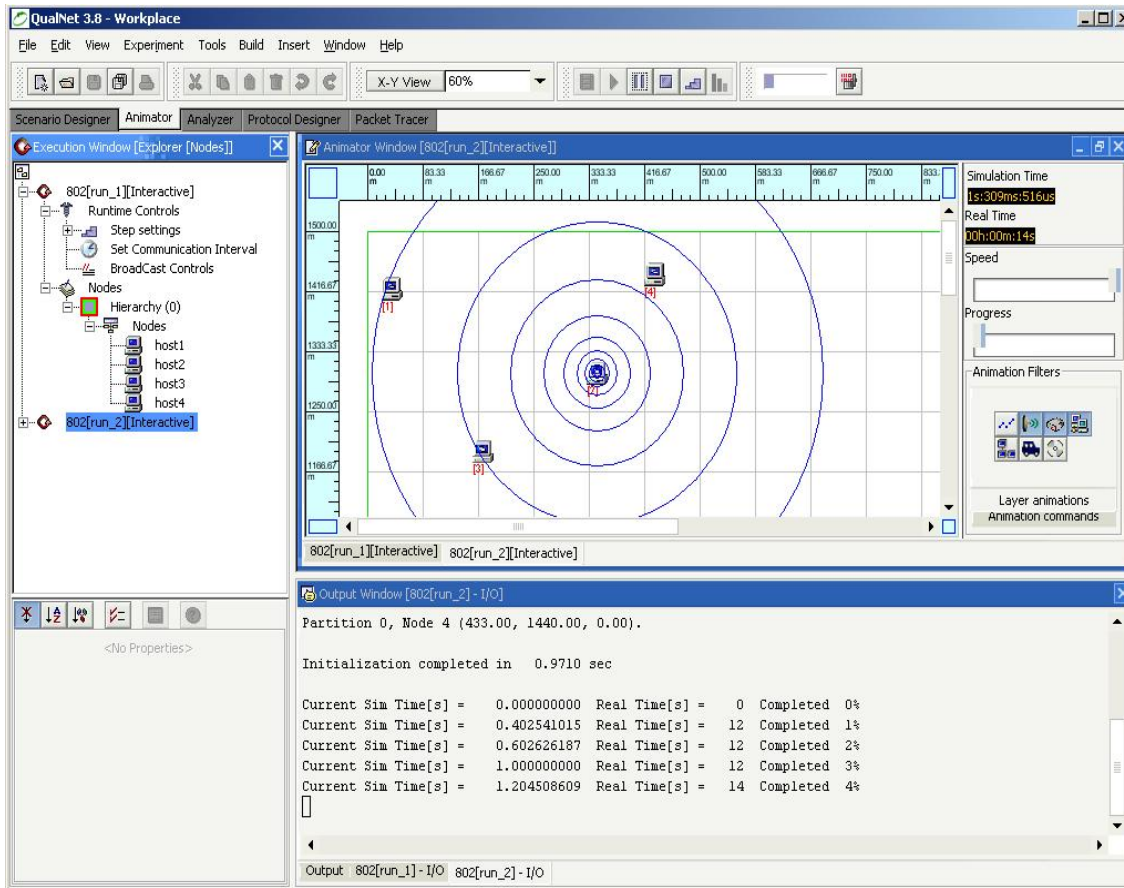


Figure 3.1 QualNet GUI Screenshot

3.2 Operation

Qualnet bases its operation around the OSI protocol stack. At each layer in the stack, a different protocol can be implemented for a given simulation. The protocol stack used can be seen on the following page. Because of this modular approach, it makes it very easy to integrate existing protocols with ones we develop ourselves.

For any given simulation, we have a configuration file. This file specifies what protocol is to run at each layer and any parameters such protocols may require. QualNet provides a whole range of existing protocols. For example, at the application layer, models exist for Constant Bit Rate, CBR, File Transfer Protocol, FTP, HTTP and TELNET traffic. It also contains models for TCP and IP at the transport and network layers respectively.

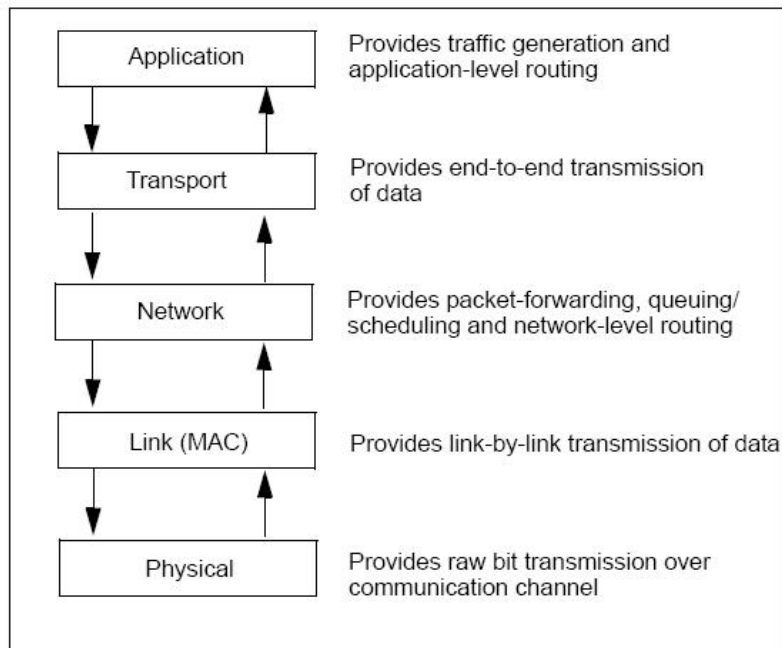


Figure 3.2 QualNet protocol stack

A number of routing protocols are also available such as AODV, RIP, OSPF and Bellman-Ford to name but a few. At the MAC layer, there are implementations of the IEEE 802.3 protocol and the IEEE 802.11 (Wi-Fi) protocol.

A major strength of QualNet is the detail with which it models the physical channel. It is because of this that QualNet performs well in wireless scenarios. Detailed models exist for fading, path loss and shadowing. In addition to this, the antennas are very well modeled, providing omnidirectional, steered beam and switched beam antennae. Physical layer protocols exist for IEEE 802.11, IEEE 802.3 and an abstract physical layer. The modular design of QualNet allows combinations of all these protocols to produce a very powerful network simulator.

Whilst each protocol obviously performs different functions, there is a common ground to their operation. Firstly, the protocol undergoes initialisation at each node it is to be run at. Depending on the node, different states may be occupied at initialisation. For example,

a node running IEEE 802.16 as its MAC layer will have different states depending on whether it is a Base Station or a Subscriber Station. After this step, the two other functions of importance are the event dispatcher and the event handler. The event dispatcher is responsible for starting any events such as sending packets or starting timers etc. The event handler is responsible for reacting to certain events. An event may constitute receipt of a packet, or a change of state. Whilst it is difficult to appreciate how a protocol may work at such a high level, it is important to think of any protocol being run as a state machine, which then implements an event dispatcher and an event handler. It will become clearer later during discussion of the IEEE 802.16 models. At the end of the simulation, a finalisation function is run, which allows us to print out statistics.

It is useful to discuss the structure of a QualNet configuration file, which is the input to the program. Firstly, the network structure is specified. This involves telling the program the number of subnets, their addresses and the number of nodes in each one. Following this, the channel parameters should be specified. This involves the channel frequencies, propagation limits and propagation models. The next step is to declare the physical layer model. Here, the transmit power, receiver sensitivity and all antenna parameters must be specified. Next the MAC layer should be specified, followed by the network layer and its routing protocol. The parameters here are specific to the protocol used. The application layer is one of the last things to be specified, and is contained in a separate *.app file. In this file, the type of traffic being sent between two nodes is provided. Giving this .config file as a parameter to the QualNet program allows us to run the simulation. The output is the statistics file, which gives us details of what has happened at each layer.

This section gives us a high level idea of how QualNet operates, and a feeling of how powerful it is. The next step is to discuss the IEEE 802.16 existing model in the context of its QualNet implementation.

Chapter 4: The existing QualNet IEEE 802.16 model

4.1 Structure of model

The existing model in QualNet developed in a previous M. Eng project [11] is based on the IEEE 802.16a standard. In terms of file structure, there exists a `mac_802_16a.cpp` and `mac_802_16a.h` file. There is a certain amount of coding to integrate this into QualNet, such as adding these files to the makefile and adding it to the list of MAC protocols among others. However, this coding will not be discussed here and can be found in [12]. Within the `mac_802_16a.cpp`, the protocol is implemented. This file must cover two scenarios. If the given node at which it is running is a Subscriber Station the MAC layer performs different functions than if that node was a Base Station.

In the previous section, the structure of a protocol was discussed. Firstly there is the initialisation function, then the event dispatcher and event handler and finally the finalisation function. In `mac_802_16a.cpp` there are two event handlers and two event dispatchers, one for the SS and one for the BS.

4.2 Operation of protocol

The diagram on the following page illustrates the most important states the node can take on.

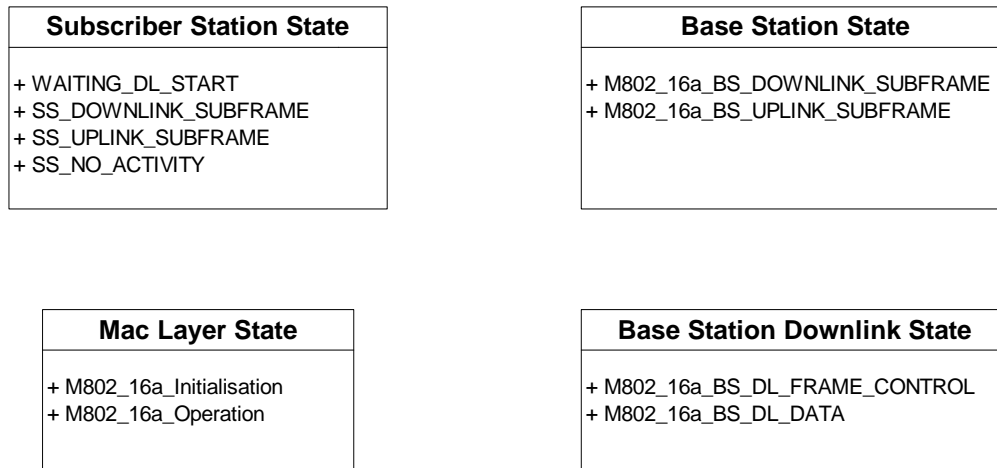


Figure 3.1 Possible states

The initialization function does the following:

Base Station:

- Initialises current DCD and UCD
- Initialises current UL-MAP and DL-MAP. UL-MAP is null and DL-MAP only contains initial ranging opportunities.
- Sets up Broadcast and Initial Ranging connections.
- MAC Layer State = M802_16a_Operation
- BS State = M802_16a_BS_DOWNLINK_SUBFRAME
- Sends Internal Message to start Downlink Subframe

Subscriber Station:

- MAC Layer State = M802_16a_Initialisation
- Mainly used for memory allocation and setting parameters to null.

The function which models the MAC layer operation is called `M802_16aLayer`. This is effectively the event dispatcher and controls the frame sequencing. In the header file there is a definition made between Frame events and Timer Events. A Frame Event may

be the start of the Uplink sub-frame whilst a Timer event may be the expiration of a timer which tells us to send another DCD, for example. It checks if the station is a BS or SS, then it checks if the event in question is a Frame Event or Timer Event. Depending on the result it calls one of the following:

- `Mac802_16aBSFrameEventHandler`
- `Mac802_16aSSFrameEventHandler`
- `Mac802_16aBSTimerEventHandler`
- `Mac802_16aSSTimerEventHandler`

Mac802_16aBSFrameEventHandler

Firstly, we will look at `Mac802_16aBSFrameEventHandler`. This first checks the BS State to see if we are in the DL Subframe or the UL Subframe. It then checks the event type and acts accordingly:

BS State = DL_Subframe

Event Type:

- `BS_START_DOWNLINK_SUBFRAME`
Sends message to start UL Subframe after required time. Sets Base Station DL state to be frame control. Sends message of event type `FRAME_EVENT_END_TRANSMISSION` with 0 delay.
- `FRAME_EVENT_END_TRANSMISSION`
If `BSDLState` is frame control, we send DCD, UCD, UL-MAP and DL-MAP on Broadcast connection. Having done this, reset state to `BS_DL_DATA` and send `FRAME_EVENT_END_TRANSMISSION` with 0 delay. If `BSDLState` is `BS_DL_DATA`, it loops through all existing connections and attempts to serve the packets buffered in them.
- `BS_END_DOWNLINK_SUBFRAME`
Sends Message `START_UL_SUBFRAME` and changes `BSSState` to be `UL_Subframe`.

BS State = UL_Subframe

Event Type:

- **START_UL_SUBFRAME**
Sends a message to end the UL Subframe after UL Subframe length. Also calls the `Mac802_16aBSUplinkOpportunityTimingHandler` function. This function sends a message of event type `BS_START_UPLINK_OPPORTUNITY` for each uplink opportunity in the UL-MAP, to start after the required delay.
- **BS_START_UPLINK_OPPORTUNITY**
Received for each UL opportunity. Sets the authorised Connection Identifier, CID. As long as the authorised CID is set, only packets from this connection will be accepted.
- **BS_END_UPLINK_SUBFRAME**
Changes `BSState` to `M802_16a_BS_DOWNLINK_SUBFRAME` and sends message to start DL Subframe immediately.

Mac802_16aSSFrameEventHandler

This function operates in a similar way to the one above, performing a case on each possible event type in the received event.

Event Type:

- **SS_START_DOWNLINK_SUBFRAME**
Sends messages of type `SS_START_DOWNLINK_SUBFRAME`, `SS_START_UPLINK_SUBFRAME` and `SS_END_UPLINK_SUBFRAME` after the appropriate delay. Sets `SSState` = `M802_16a_SS_DOWNLINK_SUBFRAME`.
- **SS_START_UPLINK_SUBFRAME**
This event type changes the `SSState` to `M802_16a_SS_UPLINK_SUBFRAME` and calls the `Mac802_16aSSUplinkOpportunityTimingHandler` function. This causes a message to be sent of event type

START_INITIAL_RANGING_OPPORTUNITY (for when the SS is just initialized, and needs to send a RNG_REQ),
 START_BASIC_CONNECTION_OPPORTUNITY,
 START_PRIMARY_CONNECTION_OPPORTUNITY,
 START_SECONDARY_CONNECTION_OPPORTUNITY or
 START_TRANSPORT_CONNECTION_OPPORTUNITY depending on the UL-MAP.

- For each of the Connection opportunities sent by the `Mac802_16aSSUplinkOpportunityTimingHandler` function, we serve any packet that is in that connection.
- `FRAME_EVENT_SS_END_UPLINK_SUBFRAME`
 This sets the `SSState` to no activity. From here we are waiting for the DL Subframe to start.

The timer event handlers are responsible for responding to expired timers and are used to send the DCD and UCD at specified intervals.

The packet handlers are the next to look at. These respond to any packets received that are relevant to the MAC layer. Therefore, all code is in response to the MAC management messages. They are also responsible for interfacing with the network layer (on receipt of a data packet) and the physical layer (when a data packet is to be sent). Structures are defined for each of the relevant MAC management messages:

| | | | | |
|-----|--------|---------|---------|---------|
| UCD | DL-MAP | RNG-REQ | REG-REQ | DSA-REQ |
| DCD | UL-MAP | RNG-RSP | REG-RSP | DSA-RSP |

The two functions to look at are `Mac802_16aSSHandlePacket` and `Mac802_16aBSHandlePacket`.

Mac802_16aSSHandlePacket

The first thing this function must do, is if the SS has just come out of initialization, the MAC layer is in the `M802_16a_INITIALISATION` state, and so the delay between the BS and SS must be calculated. It does this by using the packet creation time.

Following this, the first thing to be done is check if the packet has come from the Broadcast connection. If so, we check if it is a DCD, UCD, UL-MAP or DL-MAP. If it is one of these, and if it needs to be updated, we copy the data into the SS structure. The next thing to check is the Initial Ranging CID. The only thing that we could get here is a `RNG_RSP`. If received, this will contain the basic and primary connection details. These are noted and a `REG_REQ` is sent.

The primary connection is checked next. We must check for the following messages:

- `REG_RSP`: Contains details of the secondary connection.
- `DSA_REQ`: This will contain details of our new transport connection. We respond with a `DSA_RSP`
- `DSA_RSP`: Sets the given transport connection to the CID sent in the `DSA_RSP` and sets this connection state to `OPERATION`. Then we send a `DSA_ACK` to acknowledge activation of this connection.

Finally, we check the transport connections. Any packets received here get their MAC headers removed and then get sent to the Network Layer.

Mac802_16aBSHandlePacket

The first two things that this function checks are that the `BSState` is `M802_16a_BS_UPLINK_SUBFRAME` and that the CID over which the packet was received is the authorised CID. Having done this, we proceed in a similar fashion as before.

Firstly, checking the Initial Ranging CID, if we receive a RNG_REQ, we generate a Basic and Primary connection (uplink and downlink) and send this to the SS in the form of a RNG_RSP. We also add this SS to the list of registered stations, and modify the UL-MAP to provide opportunities for these new connections.

Next we check the primary connection for the following messages:

- REG_REQ: Creates secondary connection (uplink and downlink), modifies UL-MAP and sends details back to SS in REG_RSP.
- DSA_REQ: Creates transport connection (uplink and downlink), modifies UL-MAP and sends details back to SS in DSA_RSP.
- DSA_RSP: Activates relevant transport connection, and sends DSA_ACK back to SS.
- DSA_ACK: Activates relevant transport connection.

Finally, we check the transport connection and send any packets from here up to the Network layer after removing their MAC header.

Mac802_16aNetworkLayer- HasPacketToSend

The one other function worth mentioning is the `Mac802_16aNetworkLayer-HasPacketToSend`. The Network layer calls this whenever it has a packet to be forwarded and it operates as follows. The packet is de-queued and its next hop address is obtained. If it is a broadcast packet, we buffer the packet onto the Broadcast connection in the case of a BS and onto the secondary connection, in the case of a SS. Otherwise the packet has a specific destination and we check for an existing transport connection. If none is found we create one, via a DSA_REQ. Note that this is done separately for a BS and SS. Finally, the packet is buffered onto this connection.

Summary

In this chapter, the existing model of the MAC layer of IEEE 802.16a is outlined. Clearly, this is only a subset of the functionality outlined in the standard. However, it is a good starting model, which allows us to simulate Wi-Max networks running traffic. There are limitations to this code. For example, the allocation of the Uplink opportunities is very static which leads to a bottleneck for MAC traffic, and no quality of service is implemented. Also, there is a limit to the amount of traffic this model can support.

Chapter 5: Required Modifications to implement IEEE 802.16e

5.1 Design of model

Given that the standard is so broad, the first step in implementation was to decide on what subset was to be implemented. In a practical situation, it is most likely that the Mobile Station will make the decision whether or not to perform Handover. The station itself could decide if the signal it is receiving is not strong enough or if its received QoS is not adequate. Therefore, this was the Handover scenario that was implemented.

Initially it was chosen to omit the scanning procedure with a view to later implementation if time permitted. Unfortunately, it was not possible to implement this. The reasoning behind omitting this at the start was that the scanning procedure is really more concerned with the MS finding a set of stations, which it may move to. It was felt that looking at the Handover process itself would be more beneficial. Omitting this is obviously not ideal, as the scanning process will clearly introduce extra delay to the system. The flow chart on the following page illustrates the model, which was implemented.

This still leaves us with a decision to make. What is the Handover criterion to be defined as? In this simulation, we are not trying to mimic a real practical situation. Investigation of the effects of Handover does not require a reason for Handover to occur, unlike in a practical situation. Therefore, for the purpose of the simulation, Handover was simply specified to occur on expiration of a timer specified in the QualNet configuration file.

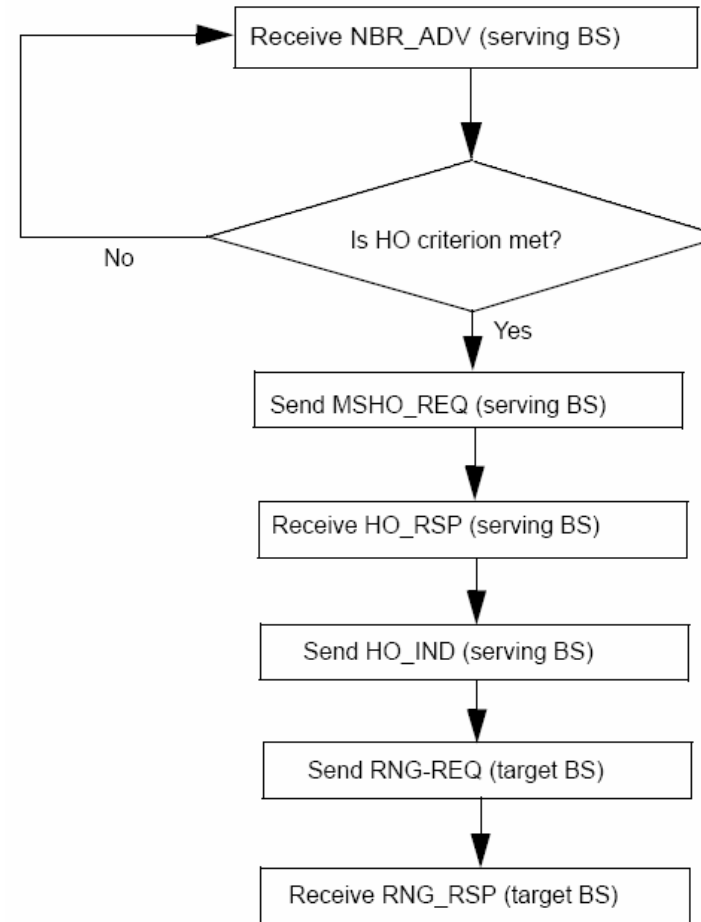


Figure 5.1 Implemented Model

5.2 Neighbour Advertisement

The first step in the process was to implement neighbour advertisement. As outlined in chapter 2, it is necessary for a BS to transmit a MOB_NBR-ADV message at a given interval in order to determine characteristics of surrounding BS's, which may be possible targets.

A BS should be capable of transmitting this message on the broadcast connection at a specific interval. In a practical situation, it is most likely that a BS would be configured with this neighbouring information when it is being set up. There is no specification in the standard for the exchange of information between BS's to allow the formation of a complete MOB_NBR-ADV message. Therefore, all neighbouring information is

specified in the QualNet configuration file. On initialisation, each BS reads in this data to the structure, `BSDescriptor`, for each neighbour and stores it. From this stored data, a structure, `M802_16e_MOB_NBR_ADV`, is formed which consists of a linked list of `BSDescriptors`.

In the header file a timer, `NBR_ADV_INTERVAL`, is specified as being 1 second. Upon initialisation and whenever a `MOB_NBR-ADV` message is sent, a timer is set to expire after this interval. Upon expiration of this timer, a flag is set to send a `MOB_NBR-ADV` message at the next downlink subframe opportunity.

From the point of view of the simulation, the necessary information to perform HO to another BS is the stations node identifier and its channel index. This is the information stored in the structure `BSDescriptor`. However, in the standard, these station descriptors contain much more information. Therefore, when sending a message it is very important to allocate the correct packet size. The standard specifies that a fixed header of 8 bytes be present, together with a variable amount for each descriptor. For each descriptor, there are optional fields. All fields bar the 'scheduling services supported' field were included. This leaves,

$$\text{Message size} = 8 \text{ bytes} + 15 \text{ bytes} * \text{number of descriptors.}$$

The sending of this message is implemented in the function, `Mac802_16eSendNBR_ADV`, which constructs the message, allocates the packet and transmits immediately on the broadcast connection.

5.3 Negotiating Handover

As discussed in an earlier section, Handover in a node is set to occur after the expiration of a timer. Once this timer has expired, the MS transmits a MOB_MSHO-REQ message to its serving BS. Recalling from chapter 2, this message should contain a list of the BS's that the MS has deduced to be appropriate HO targets from scanning. In the absence of the scanning process, we make the assumption that all stations in the MOB_NBR-ADV message are appropriate targets. The MOB_MSHO-REQ message contains a linked list of `BSDescriptor` structures, one for each recommended target station.

Again, a decision had to be made on what size the message should be. This message has a fixed header of 4 bytes which contains such information as message type, number of descriptors present etc. For each descriptor, a size of 8 bytes is used. This omits the 48-bit BSID and instead uses the compressed 8-bit version. In addition to this, there are 3 bytes of TLV (Type, Length, Value) information included which contains authentication information. This leaves,

$$\text{Message size} = 4 \text{ bytes} + 8 \text{ bytes} * \text{number descriptors} + 3 \text{ bytes}$$

Transmission of this message is implemented in the function, `Mac802_16eSendMSHO_REQ`. This function constructs the message, allocates the packet and buffers the message onto the MS's basic connection.

In the BS's packet handling function, provision must be made for the receipt of this MOB_MSHO-REQ message. The BS should take the list of received stations and consult with them over their suitability as targets. The BS should then return whatever subset of the received stations, which are appropriate targets. Again, this has been omitted under the assumption that the MOB_NBR-ADV message contains all suitable targets, and so it is this list that is returned to the MS.

Therefore, on receipt of a MOB_MSHO-REQ message, the BS first scrolls through its list of MS descriptors in order to find the appropriate basic connection on which to return the response. On finding this message, it calls the `Mac802_16eSendMOB_BSHO_RSP` message. This message contains a fixed header of 3 bytes. Each descriptor is then 11 bytes. The reason that this is larger than for the MOB_MSHO-REQ message is that this message contains extra information such as service level predictions etc. This leaves us with:

$$\text{Message size} = 3 \text{ bytes} + 11 \text{ bytes} * \text{number descriptors}$$

The function `Mac802_16eSendMOB_BSHO_RSP` then allocates this packet size and buffers it onto the BS's basic connection that was previously found.

The final step in the negotiation of Handover is the MOB_HO-IND message. On receipt of the MOB_BSHO-RSP message, the MS should reply with this message. Provision for this is made in the `Mac802_16eSSHandlePacket` function. Upon receipt of a MOB_BSHO-RSP message, the MS must generate this MOB_HO-IND message. The standard specifies a fixed size of 9 bytes for this message. It should indicate if Handover is to proceed, and if so, the target station. At this stage, the decision must be made as to which BS to use from the list provided by the MOB_BSHO-RSP message. This simulation simply chooses the first station in the list as its target. This message is constructed and placed on the MS's basic connection.

5.4 Performing Handover

Now that Handover has being negotiated between the BS and MS, the next stage is to implement the transfer from the serving BS to the target. First we will look at the procedure on the serving BS side.

On receipt of the MOB_HO-IND message, the BS must perform three functions. Firstly, the UL-MAP must be modified. All uplink opportunities concerning the MS undergoing Handover must be removed. This is performed in the function `M802_16eRemoveUplinkOpportunities`. Each BS contains a list of the stations, which are registered to it. These descriptors contain references to that station's connections. The function `M802_16eRemoveSSFromBSDescriptor` is responsible for deletion of these references. The final step is to delete the connections themselves, and is completed in the function `M802_16eRemoveSSConnectionsFromBS`.

On the MS side, things are slightly more complicated. Upon sending the MOB_HO-IND message, the MS must set a flag to end its frame sequencing at the next opportunity. In the function, `Mac802_16eSSFrameEventHandler`, frame sequencing is controlled. If this flag is set to true, the next downlink and uplink subframes are not scheduled. When the frame sequencing is finished, the next step is to switch the channel. The channel frequencies are specified in the QualNet configuration file and are associated with a channel index, which from then on is used to reference the channel. The function, `Mac802_16eSwitchChannel`, is responsible for this channel switching. Firstly, we must stop the station listening to the old channel and then start it listening to the new channel. Then we must set the transmission channel to be the new channel. The next step is to reinitialise the station for network re-entry. This is completed in the function `M802_16eSSInitForNetworkEntry`. This involves switching some of the station states and is summarised below:

```
SSState = M802_16e_SS_WAITING_DL_START
SSBasicallyRegistered = FALSE
SSRegistered = FALSE
MacState = M802_16e_INITIALISATION
```

Together with this the connections must all be set to null. Before this is done, any packets in these connections are counted as being dropped and the memory is freed. The function, `M802_16eCheckBuffers`, is responsible for this.

Now that the MS is free of its previous serving BS and listening on the channel of its new target BS, the next step is to begin frame sequencing on the BS. Before this can be completed, the station must first receive the DCD to discover the channel conditions. Once this is received, the DCD is updated. The next downlink subframe is scheduled to start after the time at which the DCD received plus the frame length. In this way, the MS is now synchronised with the BS's frame. This now completes the Handover process in terms of the MAC layer. The station can now register in the exact same way as before.

5.5 Network Layer issues

Despite the Handover process being complete, there is still one more problem. Any routing protocol would have routes already existing through the network. Now that this node has moved, any routes involving that node must be disabled and new routes must be found. In terms of development of the simulator, this proved quite troublesome and it was not possible to generalise this section of the code due to time constraints. It had to be written specifically for the Ad-hoc On-Demand Distance Vector, AODV routing protocol. There are two reasons for this choice of routing protocol. Firstly, it is quite efficient as routes are only established when two nodes wish to communicate. Secondly, because it was designed for Ad-hoc networks, it can quickly establish new routes and disable old ones.

The AODV protocol works as follows. A node keeps track of its neighbours by means of a HELLO message. These are nodes with which the node can directly communicate. When the node wishes to establish a route to a node, which is not a neighbour, it must generate a route request, RREQ, message and broadcast it to its neighbours. If the neighbours cannot find a route, they too generate RREQ messages and broadcast them. This process continues until a route is found and this is sent back in the form of a route

reply, RREP. If a route becomes inactive, the node will generate a route error message, RERR and broadcast it. This will cause all nodes using this route to disable it, allowing for a new route to be searched for.

In the simulation, there are two points at which a route must be disabled. If a BS is expected to forward a packet onto a node, which is not registered to it, it must disable that route. Also, when a node undergoes Handover, all its routes are rendered invalid and this must be accounted for. This is completed in the function

`Mac802_16eNetworkLayerHasPacketToSend`. When a BS tries to send a packet to a node, which is not registered, we must send a message to the network layer to disable that route. This uses functions written in the `aodv.cpp` file to disable a route. This in turn forces the protocol to update the routing tables. When an MS performs HO, it must reinitialise its routes, again sending a message to the network layer to do so. In the file, `aodv.cpp` the AODV event handler must be altered to add in these two new events.

The final step was to print out any statistics that had been gathered through the simulation.

Chapter 6: Testing and results

6.1 Interpretation of Results

Having previously outlined the model used to implement IEEE 802.16e, the next stage is to use it to produce results. An important point to note is that in the absence of an accurate physical layer, the results thus achieved must be treated with caution. The results will not be indicative of a practical situation. Therefore to get any real meaning out of them it is important to interpret them correctly. The purpose of this project was to examine the effects of the mobility signaling introduced by IEEE 802.16e. Therefore, if we consider results where HO has not occurred, and compare them to the same scenario, involving HO, we can achieve useful results. Comparing these scenarios on the same footing can give us a good idea of the effects of HO, even if the actual numbers are not accurate.

6.2 Testing Procedure and Simulation parameters

The simulator was tested with three types of traffic:

- Low quality video at 384 kb/s
- High quality video at 1Mb/s
- Voice over IP traffic

For each traffic type, three scenarios are examined. Firstly, we look at the situation with no HO to give us a benchmark. Next, we look at the situation where the client undergoes HO at a specified time. Finally, we look at the situation where the server undergoes HO.

The following parameters were specified in the configuration file:

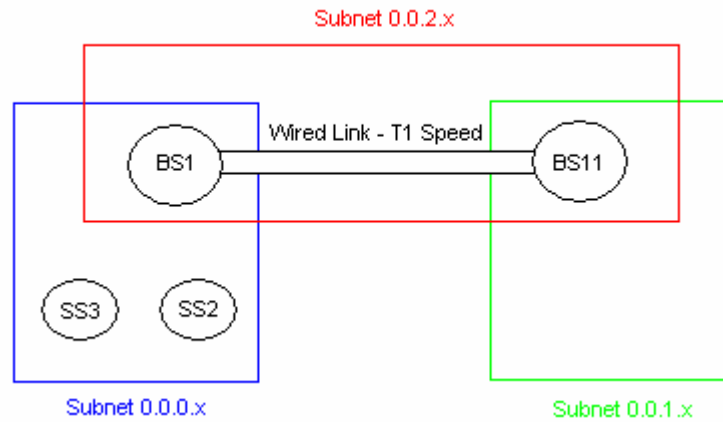


Figure 6.1 Simulation set-up

The routing protocol used is AODV, as explained in the previous chapter. An important point to note is that the AODV protocol contains a buffer. Therefore, if this buffer was large enough, no packets would be dropped, and so this would skew our results. For each simulation, an optimum value of this buffer size is set. The reason it cannot be zero is as follows. When a stations network layer initially wishes to send a packet it must wait for the required transport connections to be set up. During this time packets generated are buffered. The optimum value for this experiment is defined to be the smallest AODV buffer size that leads to zero packet loss in the absence of HO. Therefore, when HO is introduced, we can assume all packet loss is due to this alone. We should also take into account the packets in the AODV buffer, which would otherwise have been dropped.

6.3 Results

6.3.1 Traffic used

Low quality video

Modeling low quality video at 384 kb/s, traffic of constant bit rate, CBR, was used. A packet size of 1460 bytes with an inter-departure time of 30.416ms will provide this.

High quality video

Modeling high quality video at 1Mb/s, CBR again was used. Again packets of 1460 bytes were used, this time with an inter-departure time of 11.68ms.

Voice over IP

For VoIP, CBR was used. Here, packets of 200 bytes were with an inter-departure time of 20ms.

6.3.2 Results of Low Quality video

No Handover:

Here, node 2 is the server and node 3 is the client. Optimum AODV buffer size = 6 packets.

| | Node 2 (Server) | | Node 3 (Client) |
|--------------------------|------------------------|--------------------------|------------------------|
| First Packet t/x | 5s | First packet r/x | 5.2202s |
| Last packet t/x | 213.3192s | Last packet r/x | 213.34s |
| Total Bytes t/x | 10.001MB | Total Bytes r/x | 10.001MB |
| Total Packets t/x | 6850 | Total Packets r/x | 6850 |
| | | Throughput | 384.432kb/s |

Client undergoing Handover:

After 100s, node 3 migrates to BS 11.

| | Node 2 (Server) | | Node 3 (Client) |
|--------------------------|-----------------|--------------------------|-----------------|
| First Packet t/x | 5s | First packet r/x | 5.2202s |
| Last packet t/x | 213.3192s | Last packet r/x | 213.3399 |
| Total Bytes t/x | 10.001MB | Total Bytes r/x | 9.957MB |
| Total Packets t/x | 6850 | Total Packets r/x | 6820 |
| | | Throughput | 382.748kb/s |

Time to complete Handover = .979931s

Time during which MS was not registered = .94013s

Packet loss = 30

1 packet was dropped at BS on discovery of no route to MS. Route error messages were then forwarded on to all nodes, at which point node 2 dropped the following 29 packets until a route was discovered.

Server undergoing Handover

After 100s node 3 undergoes Handover to BS 11.

| | Node 3 (Server) | | Node 2 (Client) |
|--------------------------|-----------------|--------------------------|-----------------|
| First Packet t/x | 5s | First packet r/x | 5.2202s |
| Last packet t/x | 213.19 | Last packet r/x | 213.3399 |
| Total Bytes t/x | 10.001MB | Total Bytes r/x | 9.96012MB |
| Total Packets t/x | 6850 | Total Packets r/x | 6822 |
| | | Throughput | 382.86 kb/s |

Time to complete Handover = .759836s

Time during which MS was not registered = .720133s

All 28 packets were dropped due to node 3 not being registered. No connections or routes existed and so the network layer dropped all 28 packets.

Comment

With lower quality video, we are dealing with a lower data rate. Hence, during Handover, less traffic gets generated and so less data is lost. In the first scenario, where there is no HO, we observe a throughput of 384.432kb/s.

When the client undergoes HO, we see that 30 packets are lost in the time it takes to regenerate a route to the node. Hence the throughput drops correspondingly.

When the server undergoes HO, we see that the packet loss is 28. The difference here can be put down to the shorter HO time. The low data rate ensures that all traffic generated gets served nearly instantaneously. No traffic gets buffered and so no data is lost due to this. Because of this, packet loss is approximately the same for the client and server undergoing HO.

6.3.3 Results of High Quality Video

No Handover:

Here, node 2 is the server and node 3 is the client. Optimum AODV buffer = 14.

| | Node 2 (Server) | | Node 3 (Client) |
|--------------------------|------------------------|--------------------------|------------------------|
| First Packet t/x | 5s | First packet r/x | 5.2202s |
| Last packet t/x | 84.99632s | Last packet r/x | 142.20008s |
| Total Bytes t/x | 10.001MB | Total Bytes r/x | 10.001MB |
| Total Packets t/x | 6850 | Total Packets r/x | 6850 |
| | | Throughput | 584.085kb/s |

Client undergoing Handover:

After 50s node 3 undergoes HO to BS 11.

| | Node 2 (Server) | | Node 3 (Client) |
|--------------------------|------------------------|--------------------------|------------------------|
| First Packet t/x | 5s | First packet r/x | 5.2202s |
| Last packet t/x | 84.99632s | Last packet r/x | 141.48s |
| Total Bytes t/x | 10.001MB | Total Bytes r/x | 9.904MB |
| Total Packets t/x | 6850 | Total Packets r/x | 6784 |
| | | Throughput | 581.584kb/s |

Time to complete Handover = 0.319949

Time during which MS was not registered = 0.280133s

66 packets were dropped. The BS, node 1, due to no route existing to node 3, dropped 29 packets. On propagation of the route error, node 2 then dropped the remaining 37 packets.

Server undergoing Handover:

Node 3 undergoes HO after 50s

| | Node 3 (Server) | | Node 2 (Client) |
|--------------------------|-----------------|--------------------------|-----------------|
| First Packet t/x | 5s | First packet r/x | 5.22.02s |
| Last packet t/x | 84.99632s | Last packet r/x | 109.42s |
| Total Bytes t/x | 10.001MB | Total Bytes r/x | 7.5482 |
| Total Packets t/x | 6850 | Total Packets r/x | 5170 |
| | | Throughput | 579.517kb/s |

Time to complete Handover = 0.779886s

Time during which MS was not registered = 0.740133s

1680 packets were dropped in total. 1624 of these were being buffered in node 3's transport connection, awaiting service. They were dropped when this connection was deleted. The remaining 56 were dropped in node 3's network layer when it had no routes during HO.

Comment

In this situation, the traffic rate is considerably higher. The system, can serve the data somewhat faster, but not instantaneously. In the case with no HO, we can see that the throughput increases to 584.085kb/s. This is still substantially less than 1Mb/s. The reason for this is a bottleneck in the simulation of the MAC layer. The model being used is not perfect. The division between the Downlink and the Uplink sub-frame is not dynamic and neither is the manner in which opportunities are assigned, leading to less than optimal results. There is a larger packet delay, as can be seen in the difference between the times when the last packet was transmitted and when the last packet was

received. When the client undergoes HO, we see packet loss is now 66. This is larger than the low quality video case. The reason for this is the higher data rate; more data gets generated in the time during where a new route is being searched for and so more data is lost. It is also interesting to note that the HO time is .319949s. Of all simulations run, this was the lowest HO time observed.

When the server undergoes HO, the packet loss is much larger, being 1680. This is because as traffic is being generated, it is being buffered on the transport connections. The high data rate results in a backlog of data on these connections, a backlog that was not seen with lower data rates. When the station performs HO, these transport connections get deleted, and so the packets are lost. Clearly, a packet loss of this size is unacceptable. One way around this would be to postpone deletion of these connections, until a new active transport connection is found. Then this data could be transferred over to the new connection, however, there is no provision for anything like this in the standard.

Because of the higher packet loss, the throughput drop is greater also than that seen in the case of lower quality video.

6.3.4 Results for VoIP traffic

When talking about VoIP, it obviously does not make much sense to talk about a server and client, as traffic will be flowing in both directions. However, if we think of the server as being the traffic generator, it allows us to meaningfully interpret the results.

No Handover:

Here, node 2 is the server and node 3 is the client. Optimum AODV buffer = 9.

| | Node 2 (Server) | | Node 3 (Client) |
|--------------------------|-----------------|--------------------------|-----------------|
| First Packet t/x | 5s | First packet r/x | 5.2202s |
| Last packet t/x | 99.98s | Last packet r/x | 100.2s |
| Total Bytes t/x | 1.52MB | Total Bytes r/x | 1.52MB |
| Total Packets t/x | 4750 | Total Packets r/x | 4750 |
| | | Throughput | 128.027kb/s |

Client undergoing Handover:

Node 3 undergoes HO after 50s.

| | Node 2 (Server) | | Node 3 (Client) |
|--------------------------|-----------------|--------------------------|-----------------|
| First Packet t/x | 5s | First packet r/x | 5.2202s |
| Last packet t/x | 99.98s | Last packet r/x | 100.18s |
| Total Bytes t/x | 1.52MB | Total Bytes r/x | 1.50MB |
| Total Packets t/x | 4750 | Total Packets r/x | 4694 |
| | | Throughput | 126.544kb/s |

Time to complete Handover = 0.799886s

Time during which MS was not registered = 0.760133s

Total packet loss is 56. 11 are dropped in BS 1 when no route to 3 exists. The remaining 45 are dropped in node 2 when it gets notification of no route to 3.

Server undergoing Handover

Node 3 undergoes HO after 50s.

| | Node 3 (Server) | | Node 2 (Client) |
|--------------------------|-----------------|--------------------------|-----------------|
| First Packet t/x | 5s | First packet r/x | 5.2202s |
| Last packet t/x | 99.98s | Last packet r/x | 100.18s |
| Total Bytes t/x | 1.52MB | Total Bytes r/x | 1.50MB |
| Total Packets t/x | 4750 | Total Packets r/x | 4707 |
| | | Throughput | 126.894kb/s |

Time to complete Handover = 0.777465s

Time during which MS was not registered = 0.744233

43 packets were lost here. 34 were lost in node 3's network layer when no routes existed. The remaining 9 were buffered in node 3's transport connection and so dropped on deletion of these connections.

Comment

It can be seen in the case of no HO, that the data gets delivered with low delay. This is because of the lower data rates than seen previously and is clearly necessary for voice traffic. When the client undergoes HO, packet loss is 56. This is equivalent to $56 \times 20\text{ms} = 1.12$ seconds of call outage. Given the smaller packet size than previously, the actual data loss is not huge, but an outage of 1.12 seconds is noticeable, but still workable. When the server undergoes HO, packet loss is lower, being 43 (equivalent to .86 seconds call outage). The lower HO time may explain some of this. In addition, a new route may have been generated quicker.

Impact of simulator quality on results

Due to differences between the physical layer used in the simulation and the actual 802.16e physical layer, on top of some of the limitations of the existing 802.16a model, these results must be treated with caution. So what differences would such simulator improvements make to our results? Using the 802.16e physical layer would have introduced adaptive modulation into the system. This means that different stations would be using different physical layer parameters depending on channel conditions, proximity to BS etc. Therefore we would have some stations operating at lower bit rates than others and this would probably lead to a decrease in throughput.

A more accurate model of the MAC layer would have removed the bottleneck discussed earlier. If there are only two stations using the system, they should be assigned half of the uplink opportunities each (in the absence of any QoS). This is not the case in this simulator, but allowing this would lead to an increase in overall throughput.

Time taken to complete Handover

The time taken to complete handover is defined as the time from when the MS decides to undergo HO to when it successfully registers with its new target BS. An alternate definition could have been the time from when the HO request is sent to the time of successful registration with new BS. The difference here is that after deciding to transfer, the MS will have to wait until its next uplink opportunity before sending this request. Therefore, this definition is longer, but it was felt it to be more realistic.

It can be seen in the results that there is some variation in the HO time. This is due to the fact that once the MS leaves its current BS, it must wait for anything up to 1 second for another DCD.

Chapter 7: Conclusion and Further Work

The results outlined in the previous chapter illustrate the data loss that the HO procedure introduces. For each traffic type, different packet losses are observed. Fundamentally, this comes down to the difference in data rates in the traffic types.

For low quality video, we can see that the data loss is not too significant in the overall context of the data transfer.

In the case of voice traffic, the data loss is still quite low, but will be more noticeable to the user. This would not be particularly significant if HO was only to occur once or twice during a conversation. However, if we have a node moving at vehicular speeds (as the standard was designed for) it may move through several air interfaces, and HO may occur several times. This may cause annoyance to the user and is difficult to overcome. In the case of data transfer, it would be possible to develop algorithms to ensure that data loss is reduced and hence only delay is introduced. However, with VoIP, any delay introduced is very undesirable and can cause a conversation to become unworkable.

The last kind of traffic looked at was high quality video, which effectively means a higher data rate. Because of this higher rate, the same Handover time leads to higher data loss. In chapter 6, it was explained that in the case of the server undergoing Handover, there needs to be provisions made to transfer data from the old connections to the new ones. There is no provision for this in the standard but it would be quite straightforward and would considerably improve the performance of this scenario.

Because of the limitations of the simulator, the results produced are not necessarily the most interesting part of this project. In chapter 2, the evolution of the Wi-Max standards was looked at. Initially the standard required a Line Of Sight between stations in the 802.16 standard. Following this, non-Line of Sight operation was catered for in the 802.16a standard. This included significant changes to the physical layer. These standards

were amalgamated for the publication of 802.16-2004, together with some slight improvements to the physical layer. Whilst the physical layer had been the subject of revision for the initial amendments, the MAC layer was looked at in 802.16e. Significant changes had to be introduced to allow for mobility. These amendments were looked at in detail, which allowed for the software implementation.

Understanding the signaling involved allows the appropriate management messages to be selected for implementation. The model defined for implementation involved acquiring a list of possible target BS's from the current serving BS. From this list a target destination is selected and the appropriate handover signaling is performed. This Handover process had to be integrated into the existing architecture of the 802.16a simulator.

Further Work

The next step in the development of the 802.16e simulator would be to introduce the signaling required to scan neighbouring BS's. Introducing node mobility would provide us with physical layer parameters (signal strength etc) allowing us to program some kind of logic into target BS selection. This would provide a better overall view of the Handover procedure.

It was mentioned in chapter 6 that in the case of high quality video or indeed any traffic at a high data rate, when the server undergoes Handover data loss is quite high. It would be interesting to implement a buffering scheme to prevent such large data loss. It doesn't make sense for the BS to delete packets when it knows where they should be going.

In terms of the overall simulator, I believe the static manner in which the frame is divided and in which uplink opportunities are assigned is a serious obstacle to getting meaningful results and this would be well worth fixing.

The two reasons for packets being dropped are for no connections existing and for no route existing. Therefore the routing protocol will have obvious implications on the

result. Being independent of the standard, different routing algorithms may be appropriate to different scenarios. It was felt AODV was the best way of finding a new route as quickly as possible though it would be interesting to run similar simulations with various routing protocols.

In chapter 2, a method of Handover called Soft Handover, SHO, was discussed. This means that the MS can receive MAC PDU's from more than one station. This would mean that there would be no data loss and so would be appropriate for the case of VoIP. However, systems which support SHO are considerably more complex and correspondingly, more expensive. Whilst it would be quite complex to introduce this to the simulator, the results would be quite interesting.

The results here have focused very much on the data loss introduced. However, in the standard itself, there is a large emphasis placed on power saving. It would be interesting to look at the standard from this viewpoint by implementing a sleep mode in the simulator.

One final area worth looking at might be the impact certain parameters have on simulation results. Once a station decides to undergo Handover and leaves the air interface of its original Base Station, the time taken to synchronise to the new Base Station depends on how long the station must wait for a DCD message. The DCD for these simulations was set at 1s, and again, it would be interesting to see if changing this parameter would have significant difference on the results.

References

- [1] A. Ghosh, D. R. Wolter, J.G. Andrews and R. Chen, "Broadband Wireless Access with WiMax/802.16: Current Performance Benchmarks and Future Potential", *IEEE Communications Magazine*, pp. 129-136, February 2005.
- [2] "WiMAX, making ubiquitous high-speed data services a reality", Alcatel White Paper, June 2004.
- [3] F. Wang et al, "IEEE 802.16e System Performance: Analysis and Simulations", presented at PIMRC Conference, 2005.
- [4] "WiMax's technology for LOS and NLOS environments", WiMax Forum White Paper, August 2004.
- [5] J. D. Gibson, "The Mobile Communications Handbook", Boca Raton, New York, 1996.
- [6] G. L. Stuber et al, "Broadband MIMO-OFDM Wireless Communications", in "Proceedings of the IEEE, vol. 92, No. 2", pp. 271-294, February 2004.
- [7] C. Eklund, R. B. Marks, K. L. Stanwood and S. Wang, "IEEE Standard 802.16: A technical Overview of the WirelessMAN™ Air Interface for Broadband Wireless Access", *IEEE Communications Magazine*, pp. 98-107, June 2002.
- [8] IEEE 802.16e/D9, "IEEE Standard for Local and Metropolitan Area Networks - Part 16: Air Interface for Fixed Broadband Wireless Access Systems. Part 16: Amendment for Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands", June 2005.
- [9] M. Alouini, "Further Results on Low-Complexity Diversity Combining Schemes", International Symposium on Control, Communications and Signal Processing 2004.
- [10] Y. Xiao, "Energy Saving Mechanism in the IEEE 802.16e Wireless MAN", "IEEE Communications Letters, vol. 9, No. 7", pp. 595-598.
- [11] B. Louazel, "Implementation of IEEE 802.16a in GloMoSim/QualNet", M. Eng Thesis, DCU, August 2004.
- [12] "QualNet Programmers Guide", Scalable Network Technologies.