Scheduling Architecture for IEEE 802.16

Developed By
Rashid Mahmood
248-FAS/MSCS/F05
Muhammad Imran Tariq
249-FAS/MSCS/F05

Supervised By
Prof. Dr. Malik Sikandar Hayat Khiyal

Department of Computer Science
Faculty of Basic and Applied Science
International Islamic University Islamabad

2008

International Islamic University, Islamabad

Dated:-21-05-2008

Final Approval

It is stated that we have read the thesis titled as **"Scheduling Architecture for IEEE 802.16"** submitted by Rashid Mahmood Reg # 248-FAS/MSCS/F05 and Muhammad Imran Tariq Reg # 249-FAS/MSCS/F05. It is our judgment that this project is of standard to warrant its acceptance by the International Islamic University, Islamabad, for the Degree of MS in Computer Science.

Project Evaluation Committee

External Examiner
Prof. Dr. Ismail Shah
Head, Department of Computer Science
Iqra University, Islamabad.

Internal Examiner
Prof. Dr. Muhammad Sher
Department of Computer Science
Faculty of Basic and Applied Sciences
International Islamic University, Islamabad.

Supervisor
Prof. Dr. Malik Sikandar Hayat Khiyal
Chairperson, Department of Computer Science/Software Engineering
Fatima Jinnah Women University, Rawalpindi.

DECLARATION OF ORIGINALITY

We Rashid Mahmood Reg # 248-FAS/MSCS/F05 and Muhammad Imran Tariq Reg # 249-FAS/MSCS/F05 student of MS (CS) in International Islamic University Islamabad, declare that this submission is our own work and that, to the best of our knowledge and belief, it contains no material previously published or written by another person or our self, nor material which to a substantial extent has been accepted for the award of any other unit, degree or diploma of a university or any other institute of higher learning, except where due acknowledgement is made in the text.

We have made and retained a copy of this original assignment.

SIGNATURE:

Rashid Mahmood ______________

Muhammad Imran Tariq ______________

A dissertation submitted to the

Department of Computer Science

International Islamic University Islamabad
As a partial fulfillment of requirements for the award of
The degree of

MS in Computer Science

Dedication

We dedicated this research project to our beloved Parents, Family members, respected Teachers and sincere Friends

Acknowledgement

At very first, we bestow all praises, acclamation and appreciation to Almighty Allah, the most merciful and compassionate. The most Gracious and Beneficent, Whose bounteous blessings enable us to pursue and perceive higher ideals of life, All praises for His Holy Prophet Muhammad (SAW) Who enabled us to recognize our Lord and creator and brought to us a real source of knowledge from Allah, The Quran, Who is role model for us in every aspect of life. Secondly we must mention that it was mainly due to our family's moral and financial support during our entire academic career that enabled us to complete our work dedicatedly. We would like to thanks to our teacher's and especially consider it a proud privileged to express our gratitude and deep sense of obligation to our reverend supervisor Prof. Dr. Malik Sikandar Hayat Khiyal for his dexterous guidance and kind behavior during the project. We also would like to admit to our truly friends that helped us in every difficulty. We once again would like to admit that we owe all our achievement to our most loving parents who mean most to us, for their prayers are more precious them any treasure on earth.

Project in Brief

Project Title:	**Scheduling Architecture for IEEE 802.16**
Organization:	**International Islamic University, Islamabad (IIUI).**
Undertaken by:	**Rashid Mahmood (248-FAS/MSCS/F05)** **Muhammad Imran Tariq (249-FAS/MSCS/F05)**
Supervised by:	**Professor Dr. Malik Sikandar Hayat Khiyal** **Chairperson, Department of Computer Science and Software Engineering.** **Fatima Jinnah Women University Rawalpindi.**
Starting Date:	**June 2007**
Completion Date:	**March 2008**
Tool Used:	**NS-2 Simulator** **C++** **MS Office**
Operating System:	**Linux Red Hat9** **Windows XP**
System Used:	**Pentium III (1000MHz Genuine Intel)** **RAM 512 MB** **HD 40 GB**

Abstract

Recent development in Broadband Wireless Access (BWA), boosted users to use a multimedia, real–time and high bandwidth intensive applications that lead to a new era of research and development in wireless network. The IEEE 802.16 standard which has come forward as Broadband Wireless Access (BWA) solution is fulfilling all requirements of users. The IEEE 802.16 Broadband Wireless Access (BWA) serve people in those areas (rural area) where it is difficult to deploy wired technologies (such like Fiber Optic cable) and it is also serve in urban area. IEEE 802.16, also known as WiMAX. Even though IEEE 802.16 standard defines Scheduling service flows and Quality of service parameter, scheduling of these flows to maintain QoS and fairness among flows is left open for researchers. We propose scheduling architecture for IEEE 802.16 in both uplink and downlink direction. Our scheduling architecture includes QoS parameters like maximum sustained rate, maximum latency, tolerated jitter, minimum reserved bandwidth, traffic priority, request transmission policy, burst size, SDU size and queue information for various scheduling services flow. We use First in First out (FIFO), Earliest Deadline First (EDF) and Self Clocked Fair Queuing (SCFQ) to schedule different flows to achieve QoS and efficient bandwidth utilization. We also associate weights with scheduling service to achieve fairness, that is calculated by queue information and priority associated with that flows. To evaluate the efficiency, performance and fairness of our architecture, we have carried simulations in both uplink and downlink direction on ns-2. Our simulation results show that there is delay and jitter guarantee to UGS and rtPS scheduling services and bandwidth guarantee to nrtPS and BE under minimum delay. Simulation result also show that it keeps fairness among subscriber station (SS). We also evaluate the role of fragmentation under this architecture.

Table of Contents

Contents Page #

List of Figures

List of Table

Chapter 1

Introduction

1. Introduction

Broadband wireless access (BWA) system is the first step towards, to meet the challenging requirement of future, that challenges lead researcher a new era of research and development. In this chapter, we give a brief introduction about IEEE 802.16 WirelessMAN, problem statements and thesis organization.

1.1 IEEE 802.16 WirelessMAN

Broadband wireless access (BWA) has become the best way to meet escalating business demand for rapid Internet connection, integrated data, voice services, video services and multimedia services. One of the most convincing aspects of BWA technology is that networks can be created in just weeks by deploying a small number of base stations on buildings or poles to create high-capacity wireless access systems. It has following advantages over its wired competitors:

1) Fast deployment and ease to implement, a BWA network can be installed rapidly without extensive underground cable infrastructure as in the case of Cable or DSL networks.
2) High scalability, providers can expand the BWA network as subscribers demand for bandwidth by adding channels, or cells.
3) Lower maintenance and upgrade costs.
4) Higher data rates.
5) Provides easily internet and other multimedia services to remote and rural users.

However, the wide scale adoption of BWA systems will be determined by its ability to overcome cost and performance barriers. If BWA can meet these challenges it could easily be the next revolution in wireless networks systems such as WLANs.

The Institute of Electrical and Electronics Engineers Standards Association (IEEE-SA) take responsibility to make BWA more widely available by developing IEEE Standard 802.16, which specifies the WirelessMAN Air Interface for wireless metropolitan area networks. The standard, which was published on 8 April 2002, was created in a two-year, open-consensus process by hundreds of engineers from the world's leading operators and vendors.

IEEE 802.16 addresses the "first-mile/last-mile" connection in wireless metropolitan area networks. It focuses on the efficient use of bandwidth between 10 and 66 GHz (the 2 to 11 GHz region with PMP and optional Mesh topologies by the end of 2002) and defines a medium access control (MAC) layer that supports multiple physical layer specifications customized for the frequency band of use. The 10 to 66 GHz standard supports continuously varying traffic levels at many licensed frequencies (e.g., 10.5, 25, 26, 31, 38 and 39 GHz) for two-way communications. The draft amendment for the 2 to 11 GHz region will support both unlicensed and licensed bands.

1.2 Problem Statement

At present there is lack of such scheduling architecture which used all mandatory parameters defined by the IEEE 802.16 to schedule packets and to achieve the Quality of Service (QoS) requirements of different application. An efficient scheduling algorithm is a heart component of any communication network which satisfies the QoS requirement. There are following points which describe current problem that need to be addressed.

1) Most authors concentrate Uplink Scheduling at Base Station (BS) and they ignore the importance of Uplink Scheduling at Subscriber Station (SS).
2) No one describe the dynamic calculation of weights in Weighted Fair Queuing (WFQ) algorithm.
3) Performance of fragmentation under uplink and downlink scheduling is needed to be explored.
4) Performance analysis of WFQ using mandatory and optional parameters under dynamic weights.
5) Efficiently bandwidth utilization in uplink and downlink channel.

1.3 Contribution of this Dissertation

The main contributions of our thesis are,

1) We developed a fair and efficient Scheduling Architecture for IEEE 802.16 in order to providing QoS guarantees for various applications.
2) This architecture characterize and classifies the traffic according to QoS Parameters such like minimum reserved bandwidth, tolerated delay, minimum jitter, maximum sustained rate, traffic priorities, traffic policies and etc.
3) Performance and analysis result are shown by simulation using Network Simulator 2 (NS-2), which have given significant improvement as compare to other scheduling algorithms.
4) WFQ with dynamic weights efficiently support in large scale network.

1.4 Dissertation Organization

Our thesis is organized as follows. In Chapter 2, we will describe the IEEE 802.16 architecture in detail and specially concentrate on Medium Access control layer, Scheduling Services and QoS Parameters. In Chapter 3, we will present the previous work in this area. In Chapter 4, we will give detailed description of component of our scheduling architecture. In Chapter 5, we will present description of ns simulator and our implementation details. In chapter 6, we will evaluate of our architecture on NS-2 and measure the performance of our architecture under performance metrics. Finally in Chapter 7, we will present our conclusions and future work.

Chapter 2

IEEE 802.16 Wireless Interoperability Microwave Access

2. IEEE 802.16 Wireless Interoperability Microwave Access

IEEE 802.16 standard defines air interface for fixed point to multipoint BWA that are competent of providing various services. The standard standardized only Physical layer and Medium Access Control layer. This chapter describes brief overview of Architecture, Medium Access Control layer, Physical Layer, and Scheduling Algorithms.

2.1 Architecture

The architecture comprises two components, a Base Station (BS) and a number of Subscriber Stations (SS). A BS is connected to public network and can handle multiple sectors simultaneously. The SS include buildings like small office, residential and small-medium enterprise as shown in the fig 2.1. Both BS and SS are fixed and whereas users inside a building may be fixed or mobile. A users inside a building may be interconnected any LAN technologies. There is two direction communication, first one is the Downlink (DL) transmission from the BS to the SSs, and is conducted in Point-to-Multipoint access method, whereas the second one is the Uplink (UL) direction. The UL channel is common to all nodes and is slotted via TDD method on a demand basis for multimedia data.

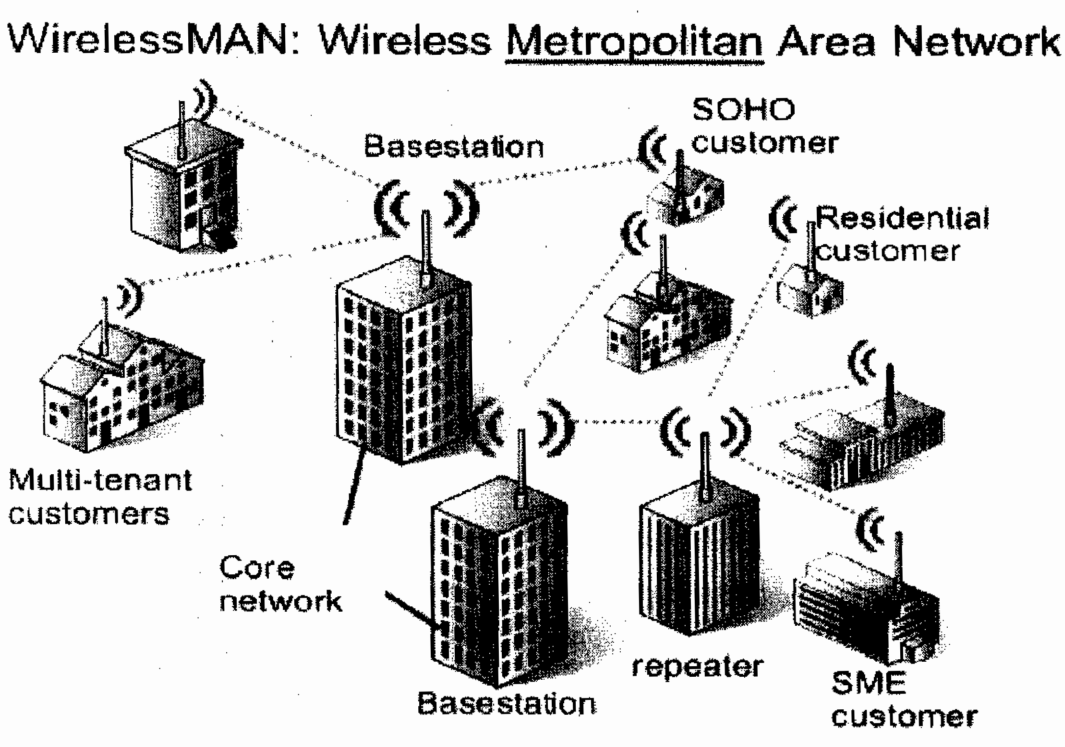

Fig 2.1: Wireless Metropolitan Area Network (Nokia network)

The 802.16 protocol stack is illustrated in fig.2.2.

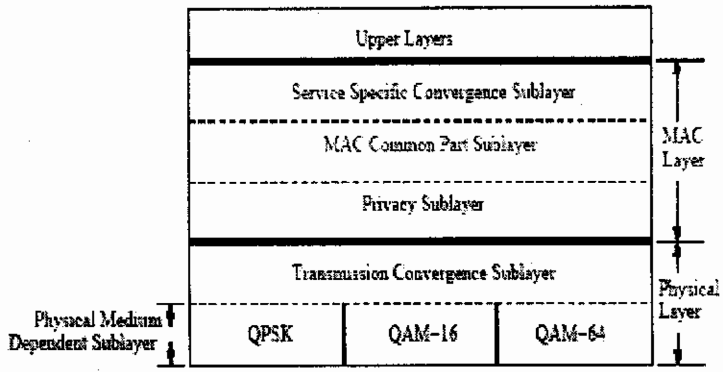

Fig 2.2: 802.16 protocol stack (IEEE standard 802.16**[16]**)

2.2 Physical Layer Details

IEEE 802.16 standard specifies one physical layer specification which operates in 10-66 GHz frequency bands. Waves in this spectrum are short in length, due to which line of sight propagation necessary for that reason millimeter waves in this frequency range travel in straight line as a result BS can have multiple antennas each pointing at a different sector. Due to sharp decline in signal strength of millimeter waves with distance from the BS, signal to noise ratio also drops very fast. For this reason 802.16 uses three different modulation scheme with Forward Error Correction (FEC) to make the channel better than it really as. The 802.16 PHY supports burst profiling in which transmission parameters, including the modulation and coding schemes may be adjustly to individually to each SS on a frame by frame basis. It supports channels as wide as 28 MHz with data rate up to 134 Mbps.

Each frame is divided into two logical channels, downlink channel and uplink channel. Uplink subframe corresponding to Uplink channel and downlink subframe corresponding to downlink channel.

The downlink channel is broadcast channel. It used by BS for transmitting downlink data and control information to various SSs. It maps the downlink traffic onto time slots and transmits a TDM signal, with individual SS allocated time slot serially. The uplink channel is time shared among all SSs. The BS is responsible for granting bandwidth to individual SSs in the uplink direction through Demand Assigned Multiple Access TDMA (DAMA-TDMA). Bs first allocates bandwidth to each SS to enable them to send bandwidth request for uplink data transmission. BS assigns variable number of physical slots to each SS for uplink data transmission according to their bandwidth demand. This information sends through UL-MAP message.

The IEEE 802.16 supports both Time divisions Duplexing (TDD) as shown in fig 2.3 and Frequency Division Duplexing for allocating bandwidth for uplink and downlink channel.

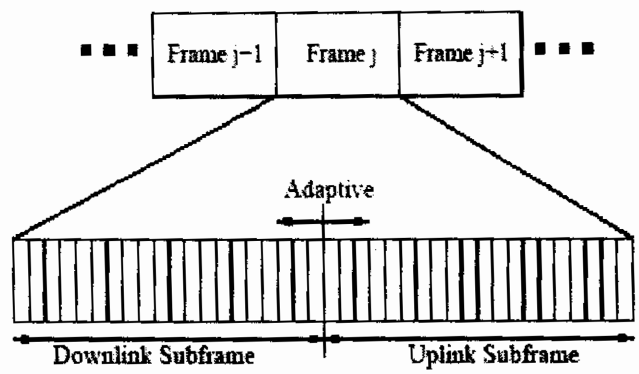

Fig 2.3: The 802.16 TDD Frame Structure (IEEE standard 802.16**[16]**)

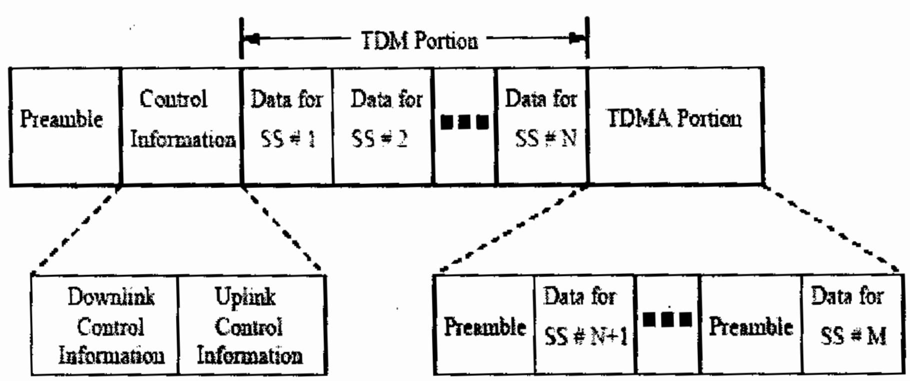

Fig 2.4: The Downlink Subframe Structure (IEEE standard 802.16**[16]**)

A downlink subframe as shown in fig 2.4, starts with a preamble used by the physical layer for synchronization and followed by Control Information which contains Downlink control information (DL-MAP) for the current downlink frame and Uplink control information (ULMAP) for uplink channel specified in the future. DL-MAP message specifies frame duration, frame number, downlink channel ID and time when physical layer transitions occur within the downlink subframe. UL-MAP message specifies uplink channel ID, the start time of uplink subframe relative to the start of the frame and bandwidth grants to specific SSs. Uplink bandwidth is allocated to various SSs in terms of minislots and allocation of minislots stated in UL-MAP. The control section is followed by a TDM portion which carries data, organized into bursts with different burst profiles. Data is transmitted to each SS using a negotiated bursts profile in the order of decreasing robustness to allow SSs to receive their data before being presented with a burst profile that could cause them to lose synchronization with the downlink. Each SS receives and decodes the downlink control information and looks for MAC header indication data for that SS in the remainder of the downlink subframe. TDMA segment contains an extra preamble at the start of each new burst profile that allows them to regain synchronization. A TDD downlink subframe is same as FDD downlink subframe without a TDMA segment.

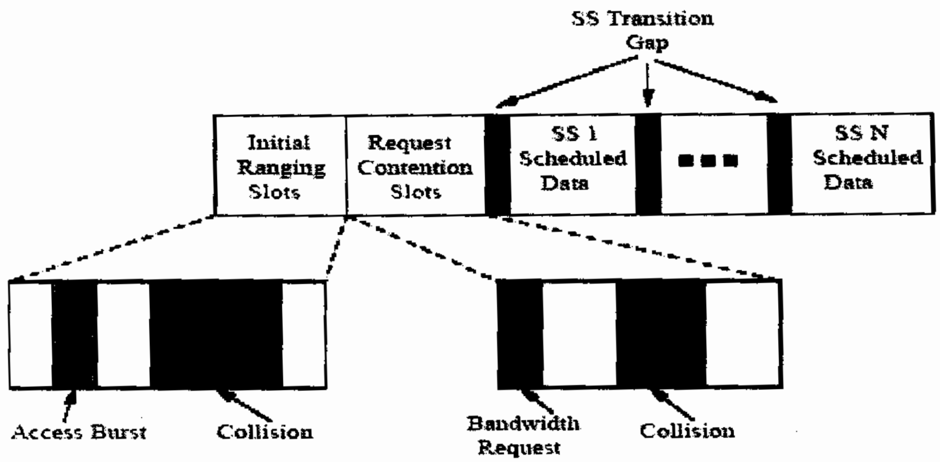

Fig 2.5: The Uplink subframe structure (IEEE standard 802.16**[16]**)

There are three types of slots in uplink subframe as shown in fig 2.5, which are contention slots reserved for initial ranging, unicast slots reserved for requesting bandwidth and unicast slots specifically to individual SSs for transmitting uplink data. They occur in any order and any quantity limited by the number of time slots allocated for uplink transmission by the BS. The SSs transmit in their specified allocation using the burst profile given in UL-MAP entry. SS Transition Gap operates the transmission of the various SSs during uplink subframe, followed by preamble allowing the BS to synchronize to the new SS.

Figure 2.6 describes an example of the OFDMA frame structure for the TDD mode. Each frame is divided into DL and UL subframes by transmit/receive transition gaps (TTGs) and receive/transmit transition gaps (RTGs). All DL subframe has a preamble in the first OFDMA symbol and then there is frame control header (FCH) in the next symbol. The FCH specifies the subchannel groups used for the burst profile, the segment, and the length of the mobile application part (MAP) message, which straight follows the FCH. The UL-MAP message is carried by the first burst allocated in the DL-MAP. Each UL subframe may have one or more ranging slots, which are used for the network entry procedure. UL subframes may also have fast feedback channels for fast channel quality indicator (CQI) reports or other fast and quick operational requests or responses.

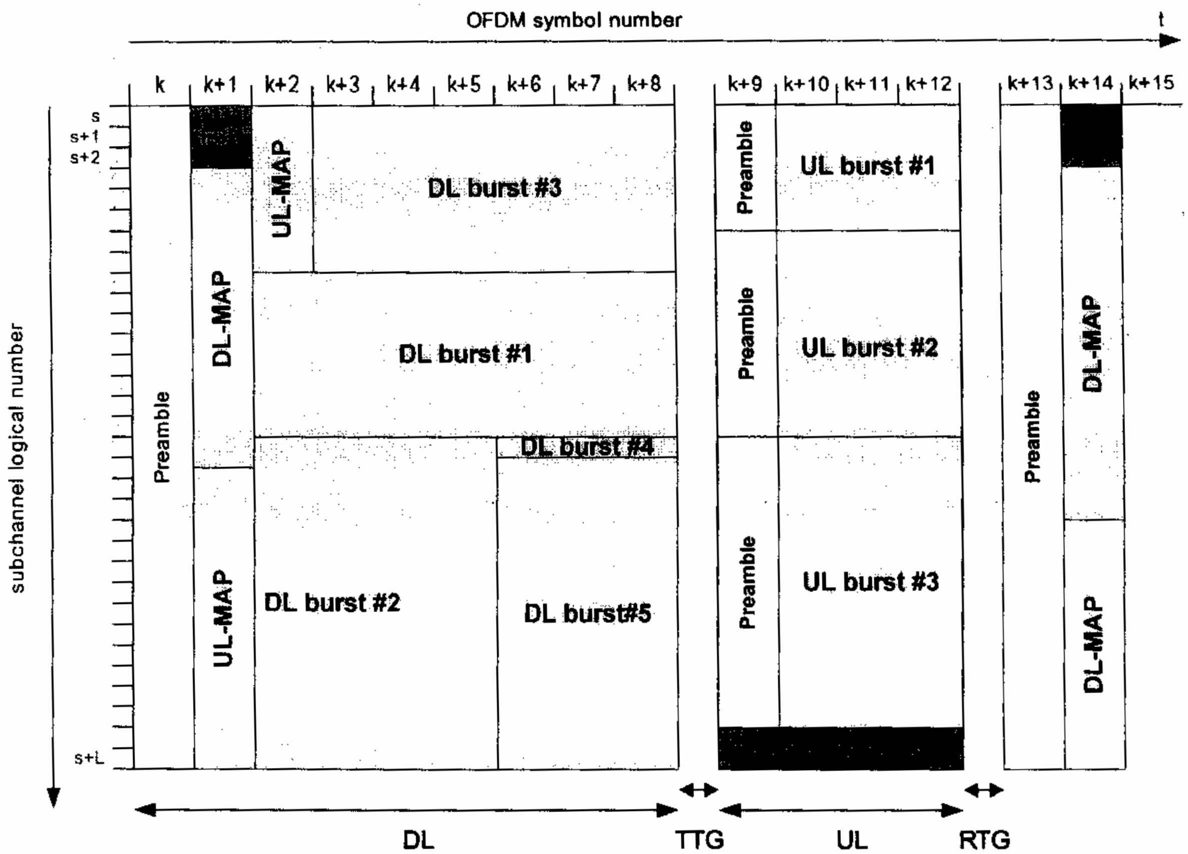

Fig 2.6: OFDMA Frame Structure (IEEE standard 802.16[16])

2.3 MAC Layer Details

The MAC layer of IEEE 802.16 provides a medium-independent interface to the PHY layer and is designed to support the wireless PHY layer by focusing on efficient radio resource management. The MAC layer supports both PMP and mesh network modes. The IEEE 802.16 MAC layer is divided into three parts:

2.3.1 Privacy Sublayer (lower)

It deals with privacy and security. This sublayer provides authentication for network access and connection establishment to avoid theft of service. It also provides encryption, decryption and key exchange for data privacy.

2.3.2 MAC Common Part Sublayer (middle)

The core MAC layer is Common Part Sublayer (CPS). The MAC CPS is designed to support PMP and mesh network architecture. It is basically designed to make efficient use of spectrum. It can support hundred of users per channel and provides high bandwidth to the users. It accommodates heavy, low, bursty and continuous traffic. The common part sublayer of the transport mechanism, which is the kernel bearing all the

MAC characteristics. It is responsible for fragmentation and segmentation of each MAC SDU into MAC protocol data units (PDU), system access, bandwidth allocation, connection maintenance, QOS control and scheduling transmission, etc.

2.3.3 Convergence Sublayer (upper).

It provides interface to the network layer above the MAC layer. Its function to map transport layer specific traffic to 802.16 MAC which is flexible enough to carry any type of traffic. The 802.16 specifies CSs that is ATM CS and Packet CS. The primary task of service-specific convergence sublayer is to classify external service data units (SDU) and associate each of them with proper MAC service flow identifier and connection identifier.

The IEEE 802.16 MAC is connection oriented. Upon entering the network, each SS creates one or more connections over which their data packets are transmitted to and from the BS. Each packet has to be associated with a connection at MAC level. This provides a way for bandwidth request, association of QoS and other traffic parameters and data transfer related actions. Each connection has a unique 16-bit connection identifier (CID) in downlink as well as in uplink direction.

The MAC PDU is data unit used to transfer data between MAC layers of BS and SS. The standard defined two types of MAC header first Generic MAC (GM) header and second is Bandwidth Request (BR) header. The generic header is used to transfer data or MAC messages while BR header is used to send bandwidth request packets to BS. SSs send their bandwidth request either in bandwidth contention period or in allotted unicast uplink slots or piggybacked with data packets. The standard defines binary truncated exponential backoff algorithm for collision resolution in contention period. Collision happens only at BS.

The standard defines a number of MAC management messages, which has to be transferred between SS and BS before actual data transfer. Any upcoming SS first synchronize itself with downlink and uplink channel to get Downlink Map (DL-MAP) and Uplink Map (UL-MAP) from BS. DL-MAP contains the information regarding downlink sub-frame while UL-MAP contains the information regarding uplink sub-frame. To setup a connection, each SS has to perform ranging, capability negotiation, authentication, registration process in-sequence. Ranging process starts by sending Ranging Request (RANG-REQ) packets to BS in ranging contention slots. SSs send RANG-REQ in each frame till it gets Ranging Response (RANG-RSP). SSs do capability negotiation, authentication in-sequence after successful RANG-RSP. Registration is also done in request-response manner by sending Registration Request (REG-REQ) packet to BS and then BS send Registration Response (REG-RSP) packet to SS. Now any SS is ready to set up a connection with BS. Connection formation is also done in request-response manner.

2.4 Bandwidth Request and Grant Mechanism

SSs use bandwidth request mechanism to specify uplink bandwidth requirement to the BS. BS polls SS by allocating bandwidth to them for the purpose of making bandwidth requests. Bandwidth always requested on per connection basis. Bandwidth can be requested by sending a bandwidth request packet or piggybacking it with a data. Request can be aggregate or incremental. When the BS receives an incremental bandwidth request, it adds the quantity of bandwidth requested to its current perception of the bandwidth needs of the connection. When the BS receives an aggregate bandwidth request, it replaces its perception of the bandwidth needs of the connection with the quantity of bandwidth requested. The IEEE 802.16 defines the following two ways for allocation of bandwidth grants:

2.4.1 Grant per Connection (GPC)

Bandwidth is allocated to a connection and SS uses this grant only for this connection. Every connection cannot use the bandwidth of the other on the same subscriber station.

2.4.2 Grant per Subscriber Station (GPSS)

SS granted bandwidth aggregated into a single grant. This SS needs more intelligent to distribute this grant into various flows, running at this SS.

2.5 QOS of IEEE 802.16

The IEEE 802.16 supports many traffic types (video, voice, and data) with different QoS requirements. The IEEE 802.16 supports constant bit rate and variable bit rate traffic. In this context, the MAC layer defines QoS signaling mechanisms and functions for data control transmissions between the BS and the SSs. In addition, the standard defines four types of data flows, each one with distinct QoS requirements.

2.5.1 Scheduling Service flows

Scheduling services represent the data handling mechanism supported by the MAC scheduler for data transport on a connection. Each connection is linked with a single data service. Each data service is linked with a set of QoS parameters that quantify aspects of its behavior. Following four scheduling services are supported by IEEE 802.16.

2.5.1.1 Unsolicited Grant Service Flows (UGS):

This service flow is designed to support Real time data streams, where fixed data packets are generated on periodic basis, such as voice over IP without silence suppression and T1/E1. The service offers fixed-size grants on a real-time periodic basis, which eliminate the overhead and latency of SS request and assure that grants are available to meet the flow's real time needs. The BS shall provide Data Grant Burst IEs to the SS at periodic intervals based upon the Maximum Sustained Traffic Rate of the service flow. The size of these grants shall be sufficient to hold the fixed-length data associated with the service flow (with associated generic MAC header and Grant management subheader) but may be large at the discretion of the BS scheduler. In order for this service to work correctly, the Request/Transmission Policy setting shall be such that the SS is prohibited from using any contention request opportunities for this connection. The key service IEs are the Maximum Sustained Traffic, Maximum Latency, the Tolerated Jitter, and the Request/Transmission Policy

The Grant Management subheader is used to pass status information from the SS to the BS regarding the state the UGS service flow. The most significant bit of the Grant Management field is the Slip Indicator (SI) bit. The SS shall set this flag once it detects that this service flow has exceed its transmission queue depth. Once the SS detects that the service flow's transmit queue is back within limits it shall clear the SI flag. The flag allows the BS to provide for long term compensation for condition, such as lost maps or clock rate mismatches, by issuing additional grants. The poll-me (PM) bit may be used to request to be polled for a different, non-UGS connection.

The BS shall not allocate more bandwidth than the Maximum Sustained Traffic Rate parameter of the Active QOS Parameter Set, excluding the case when the SI bit of Grant Management filed is set. In this case, the BS may grant up to 1% additional bandwidth for clock rate mismatch compensation.

2.5.1.2 Real-Time Polling Service Flows (rtPS):

This service flow is designed to support real-time service flow that generate variable size packet on periodic basis such as moving picture experts group (MPEG) video. The service offers real-time, periodic, unicast request opportunities, which meet the flow's real time needs and allow the SS to specify the size of the desired grant. The service requires more request overhead than UGS, but support variable grant sizes for optimum data transport efficiency.

The Bs shall provide periodic unicast request opportunities. In order for this service to work correctly the Request/Transmission Policy setting shall be such that SS is prohibited from using any contention request opportunities for that connection. The BS may issue request opportunities as prescribed by this service even if prior requests are currently unfulfilled. This result in the SS using only unicast request opportunities in order to obtain uplink transmission opportunities (the SS could still use unsolicited Data Grant Burst Types for uplink transmission as well). All other bits of the Request/Transmission Policy are irrelevant to the fundamental operation of this scheduling service and should be set according to network policy. The key service IEs are the Maximum Sustained Traffic, Minimum Reserved Traffic Rate, Maximum Latency and the Request/Transmission Policy

2.5.1.3 Non Real-Time Polling Service Flows (nrtPS):

nrtPS is designed to support delay-tolerant data streams consisting variable-sized data packets for which a minimum data rate is required, such as FTP. The nrtPS offers unicast polls on a regular basis, which assure that the service flow receives request opportunities even during network congestion. The BS typically polls nrtPS CIDs on an interval on the order of one second or less. The BS shall provide timely request unicast request opportunities. In order for this service work correctly, the Request/Transmission Policy setting shall be such that the SS is allowed to use contention request opportunities. This is result in the SS using contention request opportunities as well as unicast request opportunities and unsolicited Data Grant Burst Types. All other bits of the Request/Transmission Policy are irrelevant to the fundamental operation of this scheduling service and should be set according to network policy. The key service IEs are the Maximum Sustained Traffic, Maximum Latency, the Tolerated Jitter, and the Request/Transmission Policy.

2.5.1.4 Best Effort Service Flows (BE):

The BE service is designed to support data streams for which no minimum service level is required and therefore may be handled on a space-available basis. The mandatory QoS service flow parameters for this scheduling service are Maximum Sustained Traffic Rate, Traffic Priority and Request/Transmission Policy (a range 0-7).

In order for this service to work correctly, the Request/Transmission Policy setting shall be set such that the SS is allowed to use contention request opportunities. This is result in the SS using contention request opportunities as well as unicast request opportunities and unsolicited Data Grant Burst Types. All other bits of the Request/Transmission Policy are irrelevant to the fundamental operation of this scheduling service and should be set according to network policy. Table 2.1 describes the scheduling services usage rules according to scheduling type.

Table 2.1: Scheduling Services and Usage Rules

Scheduling Type	Piggyback Request	Bandwidth Stealing	Polling
UGS	Not Allowed	Not Allowed	PM bit is used to request a unicast poll for bandwidth needs of non-UGS connection
rtPS	Allowed	Allowed	Scheduling only allows unicast polling
nrtPS	Allowed	Allowed	Scheduling may restrict a service flow to unicast polling via the Request/Transmission policy, otherwise all forms of polling are allowed.
BE	Allowed	Allowed	All forms of polling allowed

2.5.2 QoS Parameters

Each scheduling service flow has its distinct QoS parameters that defined in standard. These are the following parameters:

2.5.2.1 Maximum Sustained Traffic Rate:

It defines the peak information rate of the service. The rate expressed in bits per second and pertains to the SDUs at the input of the system. At the SS in the uplink direction, the service shall be policed to conform to this parameter, on the average, over time. At the BS in the downlink direction, it may assume that the service was already policed at the ingress to the network and the BS is not required to do additional policing. This field specifies only a bound, not a guarantee that the rate is available.

2.5.2.2 Minimum Reserved Traffic Rate:

It defines the minimum rate reserved for this service flow. The rate is expressed in bits per second and specifies the minimum amount of data to be transported on behalf of the service flow when averaged over time. The specific rate shall only be honored when sufficient data is available for scheduling. When insufficient data exists, the requirement imposed by this parameter shall be satisfied by assuring that the available data is transmitted as soon as possible. The BS shall be able to satisfy bandwidth requests for a service flow up to its Minimum Reserved Traffic Rate. If this parameter is omitted, then its default value of 0 bits per second.

2.5.2.3 Tolerated Jitter:

It defines the maximum delay variation (jitter) for a connection with respect to time. It measures the quality of a connection. This parameter is most important for delay tolerant traffic.

2.5.2.4 Maximum Latency:

The value of this parameter specifies the maximum latency between the reception of a packet by the BS or SS on its network interface and the forwarding of the packet to its RF interface.

2.5.2.5 Traffic Priority:

It defines the priority assigned to a service flow. If there are two service flows identical in all QoS parameters besides priority, the higher priority service flow should be given lower delay and higher buffering preference. For uplink service flows, the BS shall use this parameter when determining precedence in request service and grant generation, and the SS shall preferentially select contention Request opportunities for priority Request CIDs based on this priority and its Request/Transmission policy.

2.5.2.6 Request/Transmission Policy:

It provides the capability to specify some attributes for the associated service flow. These attributes include options for PDU formation for uplink service flows, restrictions on the types of bandwidth request options that may be used. An attribute is enabled by setting the corresponding bit position to 1. For attributes affecting uplink bandwidth request types, a value of zero indicates the default action described in table 1 and value of one indicate the action associated with the attribute bit overrides the default action.

2.5.2.7 Maximum Traffic Burst:

It defines the maximum burst size that shall be accommodated for the service. Since the physical speed of ingress/egress ports, the air interface, and the backhaul will, in general, be greater than the maximum sustained traffic rate parameter for a service, this parameter describes the maximum continuous burst the system should accommodate for the service, assuming the service is not currently using any of its available resources.

2.5.2.8 Minimum Tolerable Traffic Rate:

Minimum Tolerable Traffic Rate=R (bits/s) with time base T (sec) means the following. Let S denote additional demand accumulated at the MAC SAP of the transmitter during an arbitrary time interval of the length T. Then the amount of data forwarded at the receiver to CS (in bits) during this interval should not be less than minimum {S, R*T}. In the case of downlink connection, Minimum Tolerable Traffic Rate may be monitored by the BS to make decision on rate change or deletion of the connection in the case of high SDU loss rate.

2.5.2.9 Fixed versus variable length SDU Indicator:

This parameter defines whether the SDUs on the service flow are fixed-length or variable-length. The parameter is used only if packing is on for the service flow. The default value is 0 that is variable-length SDUs.

2.5.2.10 SDU Size:

The value of this parameter specifies the length of the SDU for a fixed length SDU service flow. This parameter is used only if packing is on and the service flow is indicated as carrying fixed-length SDUs. The default value is 49 bytes.

2.5.2.11 Service Flow Scheduling Type:

It describes the scheduling service that shall be enabled for the associated service flow. If the parameter is omitted, BE service is assumed.

2.5.2.12 Vendor Specific Information:

This parameter allows vendors to encode vendor-specific QoS Parameters. This parameter helps the vendors and service provider to execute their own QoS for service flows.

2.5.3 QoS Features

The scheduler is in charge of controlling the common uplink bandwidth as well as distributing resources to flows for maintain quality. The QoS features provided by the scheduler are expected to be the only amendments to the protocol allowed, and therefore the most possible to be custom-tailored by the client Telecom according to each needs.

2.5.3.1 Fragmentation

It is the process by which a MAC SDUs divided into one or more MAC PDUs. This process is undertaken to allow efficient use of available bandwidth relative to the QoS requirements of a connection service flows. Fragmentation may be initiated by a BS for Downlink transmission and by a SS for Uplink transmission.

2.5.3.2 Piggybacking

It is used as a request for additional bandwidth sent together with a data transmission. The key advantage of this approach is that piggybacking obviates contention.

2.5.3.3 Concatenation

It is used in the MAC protocol to send more than a frame during a transmission opportunity so as to reduce packet overhead. Multiple MAC PDUs may be concatenated into a single transmission in either the Uplink or Downlink direction.

2.5.3.4 Contention

It is used in the MAC protocol to send request for more bandwidth to guarantee the QoS requirement. The size of the contention period affects the throughput of the system.

2.6 Scheduling Algorithms

There are number of scheduling algorithms developed by researcher for an efficient and fair scheduling of resources. Some of these algorithms are following:

2.6.1 Weighted Fair Queuing (WFQ)

(WFQ) is a data packet scheduling technique allowing different scheduling priorities to statistically multiplexed data flows. WFQ is a generalization of Fair Queuing (FQ). Both in WFQ and FQ, each data flow has a separate FIFO queue. In FQ, with a link data rate of R, at any given time the N active data flows (the ones with non-empty queues) are serviced simultaneously, each at an average data rate of R / N. Since each data flow has its own queue, an ill-behaved flow (who has sent larger packets or more packets per second than the others since it became active) will only punish itself and not other sessions. By regulating the WFQ weights dynamically, WFQ can be utilized for controlling the Quality of Service, for example to achieve guaranteed data rate. Proportional fairness can be achieved by setting the weights to $w_i = 1 / c_i$, where c_i is the cost per data bit of data flow i. WFQ provides bit wise bit fairness. It also provides flow isolation. It guarantees both differentiated fairness and delay among its queues. There are number of variants of WFQ such as self clocked fair queuing (SCFQ), start time fair queuing (STFQ) and worst case fair queuing (WF^2Q). SCFQ is used to handle the costly computation of round number.

2.6.2 Weighted Round Robin (WRR)

It a best-effort connection scheduling disciplines. It is the simplest emulation of generalized processor sharing (GPS) discipline. While GPS serves infinitesimal amount of data from each nonempty connection, WRR serves a number of packets for each nonempty connection (number = normalized (weight / mean packet size). To obtain normalized set of weights a mean packet size must be known. Only then WRR correctly emulates GPS. It is best to know this parameter in advance. Compared with Fair queuing (FQ) scheduler that has complexity of $O(\log(n))$ (n is the number of active flows), the complexity of WRR is $O(1)$.

2.6.3 Deficit Round Robin (DRR)

It also *Deficit Weighted Round Robin (DWRR)*, is a modified weighted round robin scheduling discipline. DRR was proposed by M. Shreedhar and G. Varghese in 1995. It can handle packets of variable size without knowing their mean size. A maximum packet size number is subtracted from the packet length, and packets that exceed that number are held back until the next visit of the scheduler.

WRR serves every nonempty queue whereas DRR serves packets at the head of every nonempty queue which deficit counter is greater than the packet's size. If it is lower then deficit counter is increased by some given value called quantum. Deficit counter is decreased by the size of packets being served. Compared with Fair queuing (FQ) scheduler that has complexity of $O(\log(n))$ (n is the number of active flows), the complexity of DRR is $O(1)$.

2.6.4 Earliest Deadline First

Earliest deadline first (EDF) scheduling is a dynamic scheduling principle used in real-time operating systems. It places processes in a priority queue. Whenever a scheduling event occurs (task finishes, new task released, etc.) the queue will be searched for the process closest to its deadline. This process will then be scheduled for execution next.

2.6.5 First In First Out

FIFO is an acronym for First In, First Out. This expression describes the principle of a queue or first-come, first-served (FCFS) behavior: what comes in first is handled first, what comes in next waits until the first is finished, etc. Thus it is analogous to the behavior of persons queuing, where the persons leave the queue in the order they arrive. It is also the other name for the FIFO operating system scheduling algorithm, which gives every process CPU time in the order they come.

2.7 Need and Importance of Scheduling Architecture for IEEE 802.16

IEEE 802.16 has been developed keeping in view the stringent QoS requirement of various applications. However it does not suggest how to efficiently schedule packets from various classes to meet their diverse QoS requirements [16][15][1][2] as shown in fig 2.7. Therefore a fair and efficient Scheduling Architecture for IEEE 802.16 required in order providing QoS guarantees for various applications. We will propose a fair and efficient scheduling architecture for IEEE 802.16 Wireless MAN with a fixed point to multipoint topology that effectively utilizes the QoS parameters that defines by standard.

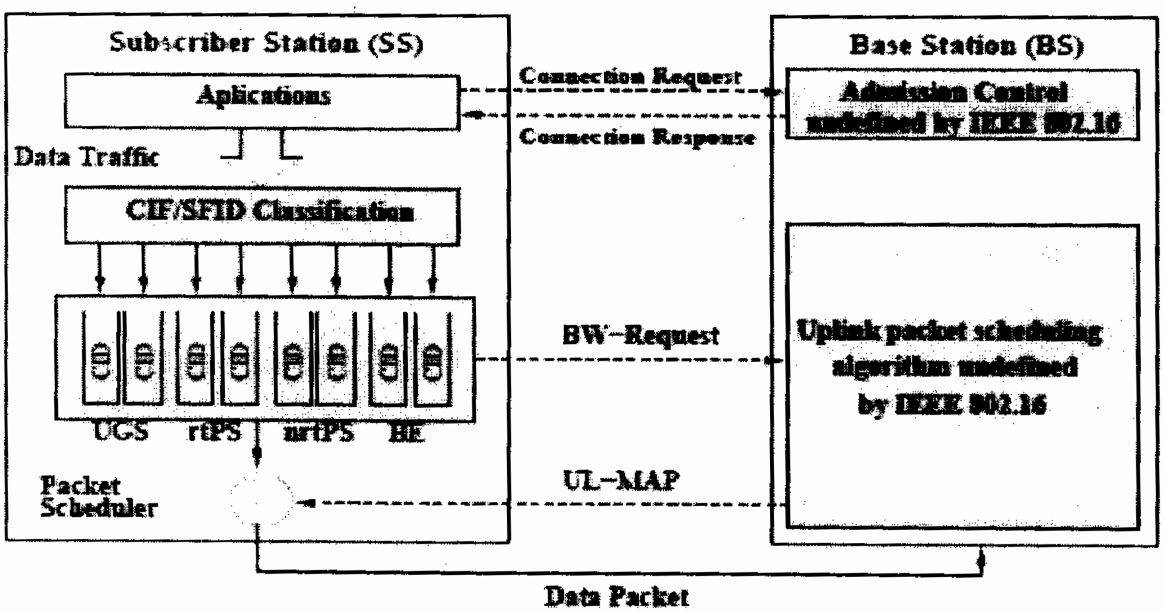

Fig 2.7: QoS Architecture in IEEE 802.16 [6, 11]

Chapter 3

Literature Survey

3. Literature Survey

Since, last few years in the field of wireless communication researcher has shown much interest. There are numbers of good papers available in Broadband Wireless System. But in this chapter, we will discuss relevant paper and how their work is different from our.

3.1 Previous Work

M. Hawa and D. W. Petr in 2002, suggested an uplink scheduling architecture for IEEE 802.16 and DOCSIS (standard for delivering broadband services over Hybrid Fiber Coax) with GPC grant mode [1]. The authors are more focused on DOCSIS rather then IEEE 802.16. The authors define three types of queues. Type 1 queues (FIFO queue) are for UGS flows and unicast request for rtPS and nrtPS flows. Type 2 (FIFO queue) queues are for flows with minimum reserved bandwidth and type 3 queue (priority queue) for flow with no bandwidth reservation. The suggested scheduling algorithm does desired slot allocation of type 1 queues then Prioritized WFQ (PWFQ) scheduling is employed for type 2 and type 3 queues in remaining slots. The authors do not specify the weight assignment for WFQ and also priority assignment for type 3 queues. The authors also provide an algorithm to calculate the number of contention slots and proposed an algorithm for dynamically adjust the no. of contention slots in frame. The authors also provide buffer management of various queues. The authors also provide an algorithm to deal with priorities. The authors believe that this scheduling architecture perform well in hardware rather than software. The authors do not provide any simulation or theoretical results for their support. Moreover the BS downlink scheduling algorithm not mentioned either.

Supriya Maheshwari in 2005 has described his scheduling architecture that based on GPSS grant mode with WFQ for downlink scheduling and min-max fair allocation for uplink scheduling [2]. He also shows his scheduling architecture effectiveness through QualNet simulations. The uplink scheduling is not exactly GPSS mode scheduling (as claimed by author) because in first stage slots are distributed into four uplink flows (on max-min fair basis) then each flow bandwidth is distributed into SSs. Moreover constant weights are used for UGS (weight 4), rtPS (weight 3), nrtPS (weight 2) and BE (weight 1) flows. The choices of these (constant) weights do not have any underlying justification. In SS uplink scheduler of their architecture, SSs send bandwidth request packets (just after finish sending UGS packets) to BS in unicast uplink slots allotted to this SS. The author do not mention anything in the context of to how these bandwidth request packets are different from the packets sent in bandwidth contention period. If these two packets are same, then why should we send them twice? In simulation analysis of this study, the author is more focused on number of SSs rather than the number of flows.

Guosang Chu et. al. in July 2002, suggested Weighted Round Robin (WRR) as uplink scheduling algorithm with GPSS (Grant per Subscriber Station) grant mode [3]. The duration of contention slots and uplink data slots are dynamically distributed according to bandwidth requirements of the SSs. Base station dynamically adjust the ratio of the bandwidth allocated to the contention slots and reservation slots. The authors chose five priority queues with dynamic priority competitive ratio parameter assignment. This ratio assignment has no justification. The authors use FIFO scheduling for lower priority service, WRR scheduling for middle priority service and WFQ scheduling for higher priority service. The authors did not describe what weights to use for WRR scheduling. The authors also describe traffic policing and traffic shaping methods that control by the BS to stop SSs intentionally and unintentionally exceed the traffic parameters negotiated during connection setup. Moreover no simulation or theoretical analysis results are presented in the study.

Jianfeng Chen et. al. in 2005, suggested Quality of Services (QoS) enhancement of IEEE 802.16 standard based on cross layer optimizations in PMP mode [4]. These optimizations include traffic classifications and packet mapping strategies of DiffServ services. The authors also design some admission control mechanism at BS. A hierarchical scheduling algorithm is deployed at BS. Six queues are defined according to their direction (uplink and downlink) and service classes (rtPS, nrtPS and BE). For UGS flows bandwidth cut from total available bandwidth before hierarchical scheduling). Deficit Fair Priority Queue (DFPQ) is used as first layer scheduling algorithm. Scheduling algorithms for different flows (except UGS), Early Deadline First (EDF) used rtPS, WFQ for nrtPS and Round Robin (RR) for BE flows. This work is more towards cross layer QoS optimization rather then QoS scheduling algorithm for IEEE 802.16 architecture. The authors present simulation results to show the effectiveness of their cross layer QoS architecture without mentioning their simulation platform.

L. F. M. de Moraes and P. D. Maciel Jr. in 2005, proposed a new MAC protocol for BWA systems with a traffic scheduling mechanism based on message and Subscriber Station's priorities [5]. The authors divide uplink sub-frame into Transmission interval and TDMA interval (or reservation interval) with dynamically changing the length of these two intervals and place the reservation period at the end of the frame. TDMA interval is used by each of the stations to inform the BS about the services for which a bandwidth reservation is being requested, as well as the number of packets to be transmitted for each of those services. In TDMA interval each station gets only one slot to send the above specified information to BS. This information could be different for different stations. The authors do not comment what happens when the information data need more then one slot for transmission. The authors proposed two versions of their analytic model, version 1 is all the message of A class transmitted before any message of class B, independent of station and is more focused on degree of fairness while version 2 all message of a station transmits before another station transmission, independent of it's priority class. The authors do not present any calculation to show how to change the length of transmission interval and TDMA interval. Moreover the calculation of average waiting time is calculated (both in version 1 and version 2) in terms of number of slots but the length of one slot is not mentioned in paper. Moreover no simulation results are presented in the study.

Aura Ganz and Kitti Wongthavarawat in 2003, suggested uplink bandwidth allocation algorithms based on flow type and strict priority from highest to lowest - UGS, rtPS, nrtPS and BE. For UGS flow fixed bandwidth is allocated, for rtPS flows Early Deadline First (EDF) service, for nrtPS WFQ and remaining slots allocated for BE flow [6]. An overall bandwidth allocation module is proposed to stop higher priority flow to use more then their allocated bandwidth. The authors use simulation model developed in C++ to show the effectiveness of their algorithm.

Howon Lee et. al. in 2004, developed an ON-OFF model to model voice traffic [7]. The authors represent the system model as Markov chain. The authors proposed an algorithm to overcome the problem of a waste of uplink resources in UGS case and MAC overhead and access delay in rtPS case. Base Station assigned slots to Subscriber Station's based on voice state transition of Subscriber Station's, which is transferred to Base Station using reserved Grant-Me(GM) bit of generic 802.16 MAC header. Base Station simply allocates full slots when GM bit is ON and exponential decrease slot size when GM bit is OFF. The authors also define the numerical formula to calculate throughput and access delay and find that access delays same as UGS traffic. Analytic and simulation results are presented to show the effectiveness of their algorithm. This study is more focused on VoIP traffic only.

Sung-Min Oh and Jae-Hyun Kim in 2005, suggested a method to calculate optimal contention period according to the number of users [8]. The authors use OPNET simulator for simulation analysis. The conclusion says that the optimal contention period duration should be 2 times the number of users. This result is unclear in the context of how many slots will be used for this optimal contention period. Moreover in system model, the authors assume that each user transmits only one bandwidth request message in each frame. This is not a valid assumption in IEEE 802.16 architecture.

Abhishek Maheshwari in 2006, proposed Weighted Fair Queue (WFQ) based MAC scheduling architecture for IEEE 802.16 WirelessMANs in both uplink and downlink direction [9]. The author scheduling architecture accommodates parameters like traffic priority, minimum reserved bandwidth and queue information for various applications. The author through ns-2 performed extensive simulation in both uplink and downlink direction for different kind of application. The author performed simulation in two mode of operation. In No Bandwidth Contention Period (NBWC) the author completely remove bandwidth contention period and send bandwidth request piggybacked with data packets thus we are removing any possibility of collisions at BS. The author shows that NBWC mode performs better in terms of delay for real time traffic and in terms of through for high data rate traffic in both uplink and downlink direction. In the performance analysis to NBWC mode, we have shown that it is possible to omit the bandwidth contention period from IEEE 802.16 standards. The author used static weights based on average minimum reserved bandwidth. The author did not used traffic priority parameter as he claimed in his abstract. Packet classification is performed on the basis of Flow ID but in real scenario Flow ID is not defined for any packets.

Claudio Cicconetti et. al. in 2006, assess the performance of IEEE 802.16 in two of the most promising scenario that are residential and small and medium-sized enterprise (SME) envisaged by the Wimax forum [10]. The authors concluded that average delay of the Uplink traffic is higher than that of the downlink traffic and they have shown that requesting bandwidth using unicast polls yields better estimation than requesting bandwidth on a contention basis by responding to broadcast polls. In this paper, they writer have not presented any scheduling architecture and they assumes Weighted Round Robin (WRR) as an uplink scheduler and Deficit Round Robin (DRR) as a Downlink Scheduler. They performed simulation on c++.

G.S. Paschos et. al. in 2006, used heuristic approach to propose a Quality of Service (QoS) strategy [11]. They implemented call admission control for high priority traffic so as to overcome the problem of starvation of network resources and also investigated different contention mini slots allocation strategy for low priority traffic. They consider four main features for QoS support which are fragmentation, concatenation, contention and piggybacked and simulate them under five different scenario using OPNET modeler. The authors have not given any scheduling algorithm at all. The authors try to overcome the collision through differentiating the backoff window of the exponential backoff algorithm which is not suitable for higher loads. The authors take very simple scenario to evaluate its results.

Seungwoon Kim and Ikjun Yeom in 2006, proposed a new uplink scheduling scheme for best-effort TCP traffic in IEEE 802.16 networks [12]. The proposed scheme does not need any bandwidth request process for allocation. Instead, it estimates the amount of bandwidth required for a flow based on its current sending rate. Through NS-2 simulation, authors show that the proposed scheme is effective to allocate bandwidth for TCP flows. The authors proposed scheduling scheme only for TCP traffic and only in uplink direction.

Victor Rangell et. al. in 2004, suggested a scheduling algorithm called EBSA that combines Early Deadline First and Prioritization, Round Robin and Weighted Fair Queuing to match VBR and CBR traffic over the IEEE 802.16 air interface [13]. The authors define different queue for UGS, rtPS, nrtPS and ordered them using the Earlier Deadline First and tolerated grant jitter is taken as ordering parameter. For BE, the grants are ordered using FIFO scheme. The purpose of *EBSA* is to provide a higher transmission priority to service flows with minimum tolerated jitter. It provides tight delays guarantees for *UGS* and *rtPS* and minimum bandwidth reservations for *nrtPS* and *BE* flows. Simulation results in OPNET of *EBSA* show that real-time services, such as VoIP, can be supported with very low access delays even on high congestion periods. The authors described only Uplink Scheduling.

Xiaojing Meng in 2007, proposed a scheduling algorithm for (OFDM/TDMA) based WiMAX network to extend proportional fairness scheme to multiple service types with diverse Quality of Service requirements [14]. The objective is to provide differentiated services according to quality of service requirement. The propose algorithm is named as Adaptive Proportional Fairness (APF). The author uses grant per type of service which aims to differentiate delay performance of each queue. For simulation, author use MatLab as a tool. In the simulation the author consider bandwidth utilization, QoS requirements, fairness, implementation complexity and scalability. Author also compares his simulation results with the result of three other scheduling algorithms namely, Round Robin, Proportional Fairness and Integrated Cross Layer Scheduling. Author also defines a priority function calculate the priority of different flows. Author only performs downlink scheduling through APF, while the uplink scheduling is not consider. Moreover, author only consider three others algorithms to construct his own algorithm but he does not mention the reason of selecting only these three scheduling algorithms. The author assumes a general traffic model. The traffic variations can affect the performance of this scheme. Further more the key parameter in the algorithm is Time Window. Accurate time estimation of the time window deserves further research.

Claudio Cicconetti et. al. in 2007, verified via simulation the effectiveness of rtPS, nrtPS, and BE (but UGS) in managing traffic generated by data and multimedia sources [15]. The author's concluded that there is a trade-off between average delay and throughput with respect to frame duration. Specifically, the longer the frame durations, the higher the average delays (the lower the throughput). The author's also concluded that transmission of physical preambles increases with the number of SSs which decreases throughput. Finally, the author have shown that SSs are able to request uplink bandwidth to the BS efficiently using piggybacked bandwidth requests, unless the system is lightly loaded. The author concluded that nrtPS scheduling service does not improve the performance of uplink connections with respect to the BE scheduling service in terms of throughput and average delay. The author also describes traffic with QoS requirements, the performance of uplink connections, in terms of delay, is highly dependent on the delay introduced by the bandwidth request mechanism. Specifically, having shorter frame duration entails lower delays, even though it increases the MAC overhead, thus reducing the throughput. Moreover, SSs might effectively exploit piggybacking and bandwidth stealing to improve the delay performance. Finally, author have shown that rtPS outperforms nrtPS in terms of delay, at least under the considered scenarios.

Jenhui Chen et. al. in 2006, designed WiMAX module for ns–2 [17]. In this paper, authors present their detailed design and implementation of the WiMAX module based on the IEEE 802.16 standard with the point-to-multipoint (PMP) mode for the ns-2. The implemented module comprises fundamental functions of the service-specific convergence sublayer (CS), the MAC common part sublayer (CPS), and the PHY layer. A simple call admission control (CAC) mechanism and the scheduler are also included in this module. This preliminary WiMAX module can benefit academic researchers and industrial developers for early verification of designing the WiMAX system.

Vandana Singh et. al. in 2006, develop new scheduling algorithms for the IEEE 802.16d OFDMA/TDD based broadband wireless access system, in which radio resources of both time and frequency slots are dynamically shared by all users [18]. The authors provide a fair and efficient allocation to all the users to satisfy their quality of service. The authors also explain the overall setup for allocation in upstream and downstream by the BS to different SSs and then by an SS to its different users belonging to the four service classes so that the QoS of the users can be satisfied in a fair and efficient way. No simulation results are presented in the study.

Maria Adamou et. al. in 2006, assess the performance of four different algorithm and evaluate them using simulation [21]. In this paper authors show that no online algorithm can guarantee a bounded performance ratio with respect to the optimal algorithm. The authors then compare four different online algorithms and evaluate them using simulations. The first two are EDF (Earliest Deadline First) and GDF (Greatest Degradation First) that consider only one aspect of our scheduling goal respectively. EDF is naturally suited for maximizing throughput while GDF seeks to minimize the maximum degradation. The next two are algorithms, called EOG (EDF or GDF) and LFF (Lagging Flows First) that consider the two aspects of our scheduling goal. EOG simply combines EDF and GDF, whereas LFF tries to favor lagging flows in a non-trivial manner. The author's simulation results show that LFF is almost as good as EDF in maximizing the throughput and also is better than GDF in minimizing the maximum degradation.

3.2. Summary of Literature Survey

IEEE 802.16 does not suggest how to efficiently schedule packets from various classes to meet their diverse QoS requirements. This area is left open for the vendors. It is necessary to provide QoS guaranteed with different characteristics for BWA networks. Therefore, an effective scheduling is critical for IEEE 802.16. There are number of author's proposed scheduling architecture for IEEE 802.16 and most of them concentrate on Uplink Scheduling and some of them also propose downlink scheduling but all these scheduling lack of some parameters that defined by IEEE 802.16 to achieve stringent quality of service. All authors used static weights for WFQ, No one describe the dynamic calculation of weights. No authors done comparative study on different scheduling algorithm that suitable for IEEE 802.16. No one analysis the performance of QoS features fragmentation, concatenation, piggyback and contention under uplink and downlink scheduling. Performance of Packing under uplink and downlink scheduling is still uncovered. Following tables 3.1 and 3.2 describe the summary of previous work.

Table 3.1: Comparison of Previous Work According to QoS Parameters

Study	**Quality of Service Parameters**					
Paper Name	Max. Sustained Traffic Rate	Min. Reserved Traffic Rate	Tolerated Jitter	Max. Latency	Traffic Priority	Request Transmissi on Policy	Other Optional Parameters
Hawa [1]		Yes	Yes				
Supriya [2]		Yes			Yes		
Chu [3]						Yes	
Chen [4]	Yes	Yes		Yes			
Ganz [6]	Yes			Yes			Maximum Burst Size
Abhishek [9]		Yes					
Cicconetti [10]		Yes					
Rangell [13]			Yes		Yes	Yes	

Table 3.2 categorizes previous work on basis of uplink and downlink scheduling, fragmentation, SS's scheduler, Queuing disciplines and simulation.

Table 3.2: Comparison of Previous work on basis of Uplink and Downlink Scheduling

Study	Analysis & classification	Subscriber Station Upstream Scheduler	Base Station Upstream Scheduler	Base Station Downstream Scheduler	Queuing disciplines	Fragme-ntation	Simulation
Paper Name							
Hawa [1]			Weighted Fair Queuing (WFQ)		FIFO with semi-preemptive Priority, FIFO with Priority enhanced WFQ, Priority Queue		
Supriya [2]	Yes	Weighted Fair Queuing (WFQ)	Max-min Fair Allocation	Weighted Fair Queuing (WFQ)	FIFO		Yes
Chu [3]		Multiclass Priority Fair Queuing (MPFQ)	Weighted Round Robin (WRR)		Wireless Fair Queuing (WFQ), Weighted Round Robin (WRR), First in First out (FIFO)		
Chen [4]	Yes		Deficit Fair Priority Queue (DFPQ)		Earliest Deadline First (EDF), Weighted Fair Queuing (WFQ), Round Robin (RR)		
Ganz [6]			Strict Priority Service		Earliest Deadline First (EDF), Weighted Fair Queuing (WFQ)		Yes
Abhishek [9]	Yes		Weighted Fair Queuing (WFQ)	Weighted Fair Queuing (WFQ)			Yes
Cicconetti [10]		Deficit Round Robin (DRR)	Weighted Round Robin (WRR)	Deficit Round Robin (DRR)			Yes
Rangell [13]	Yes		Weighted Fair Queuing (WFQ)		Earlier Deadline First (EDF), First in First out (FIFO), Round Robin (RR)	Yes	Yes

3.3. Problem Statement and Objectives

At present there is lack of such scheduling architecture which used all mandatory parameters defined by the IEEE 802.16 to schedule packets and to achieve the QoS requirements of different application. Most authors concentrate Uplink Scheduling at BS and they ignore the importance of Uplink Scheduling at SS. No one describe the dynamic calculation of weights in WFQ algorithm. Performance of fragmentation under uplink and downlink scheduling is needed to be explored. Our main objectives are following:

1) An efficient and fair scheduling architecture that support four scheduling services that are UGS, rtPS, nrtPS and BE.
2) A scheduling Architecture fully incorporates QoS parameter.
3) A scheduling architecture provides QoS guarantees and fairness among different flows.
4) Uncover the Fragmentation role under this architecture.

4. Proposed Scheduling Architecture for IEEE 802.16

Our proposed scheduling architecture incorporates all mandatory scheduling parameters of all scheduling service as defined in standard. Our proposed scheduling architecture is only for IEEE 802.16d version and it supports only point to multipoint topology. This chapter describes design goal and proposed scheduling architecture.

4.1 Design Goals

We have designed scheduling architecture with the following stated goals:

1) To provide QoS guarantees to different flows.
2) To use all QoS parameters efficiently to achieve Quality of service for scheduling service flows.
3) To provide delay bound scheduling for UGS and rtPS traffic and also provide jitter bound scheduling for these real-time traffic.
4) To provide minimum traffic rate for nrtPS traffic.
5) To provide bandwidth guarantees and less response time for BE traffic.
6) To maintain bandwidth fairness among subscriber station (SS).

4.2 Proposed Architecture Details

We proposed a scheduling architecture as shown in fig 4.1 for the IEEE 802.16 Mac protocol that incorporate the QoS. Our proposed QoS scheduling architecture mainly include BS Uplink Bandwidth Management Module, BS Downlink Bandwidth Management Module, SS Uplink Scheduler, BS Downlink Scheduler, Packet Ordering Module and Fragmentation Module. We designed such a scheduling architecture that meet the QoS of each flow and still achieving the high system bandwidth utilization. IEEE 802.16 Mac Layer provides two approaches for bandwidth allocation 1) GPC (Grant per connection) 2) GPSS (Grant per subscriber station). We used second approach, so there is need a Subscriber Station scheduler that allocate bandwidth different connection to meet the QoS.

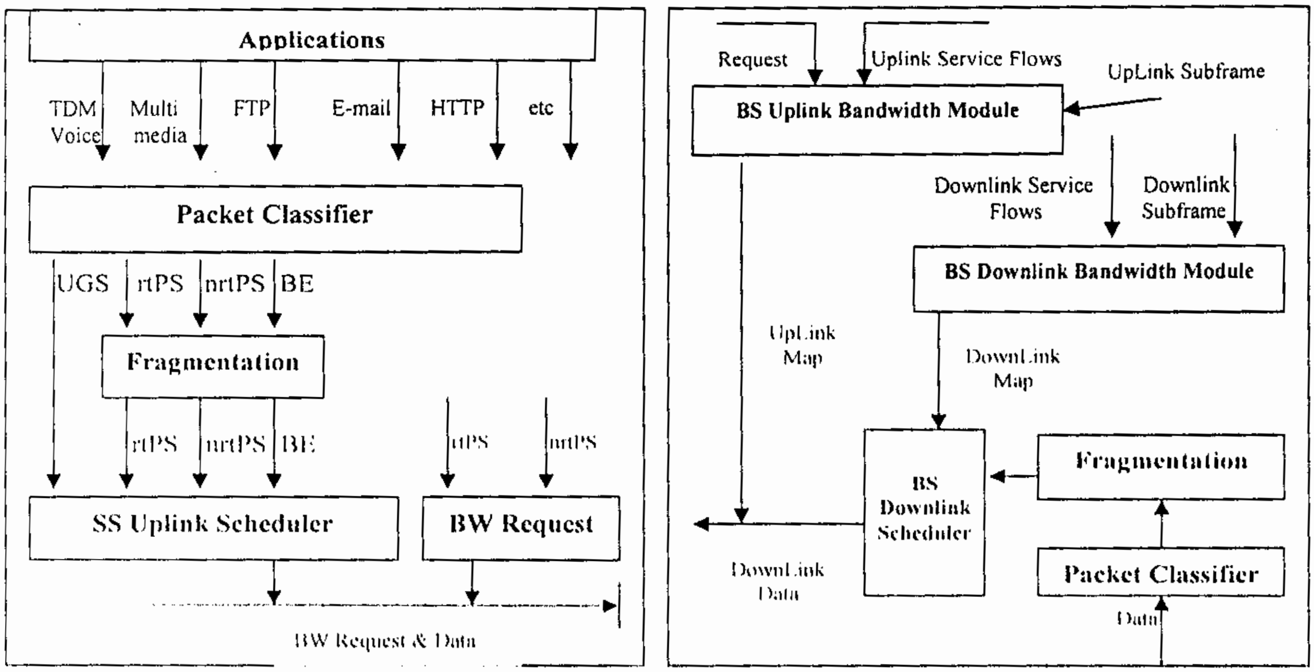

Fig 4.1: Scheduling Architecture for IEEE 802.16

4.2.1 BS Uplink Bandwidth Management Module

The IEEE 802.16 standard divides the frame into two sub frame that is uplink sub frame and downlink sub frame. This module as shown in fig 4.2 has responsibility for allocating bandwidth to each SS for uplink transmission. Bandwidth is allocated on per flow basis. This module performed its functionality only at BS (Base Station) side. It's main function to produce uplink map for all SS according to their bandwidth requirements to achieve excellent QoS for each flow. This module has little information about each uplink service flow and current state of each service flow like status of each queue.

It also keeps fairness among different flows and Subscriber Station under overloaded condition. It assures the delay guarantee to UGS and rtPS flows. But it can not assure delay guarantee to nrtPS and BE flows as we increase the number of flows. As the number of flows increasing nrtPS and BE have got little bandwidth because its priority despite that they got such bandwidth that can be enough for scheduling packets.

There is only information available to this BS module about service flow during connection setup and this information are Maximum Sustained Traffic Rate, Minimum Reserved Traffic Rate, Maximum Latency, Tolerated Jitter and Traffic priority. BS also knows about the arrival time of packets and bandwidth requirements that exchanged during bandwidth request packets. Bandwidth allocated to each SS by following way:

- Amount of bandwidth allocated to each SS in regular interval by the QoS parameters of connection associated with each connection.

- Amount of bandwidth requested by each SS for uplink transmission.
- Amount of bandwidth required by SSs UGS flow periodically.

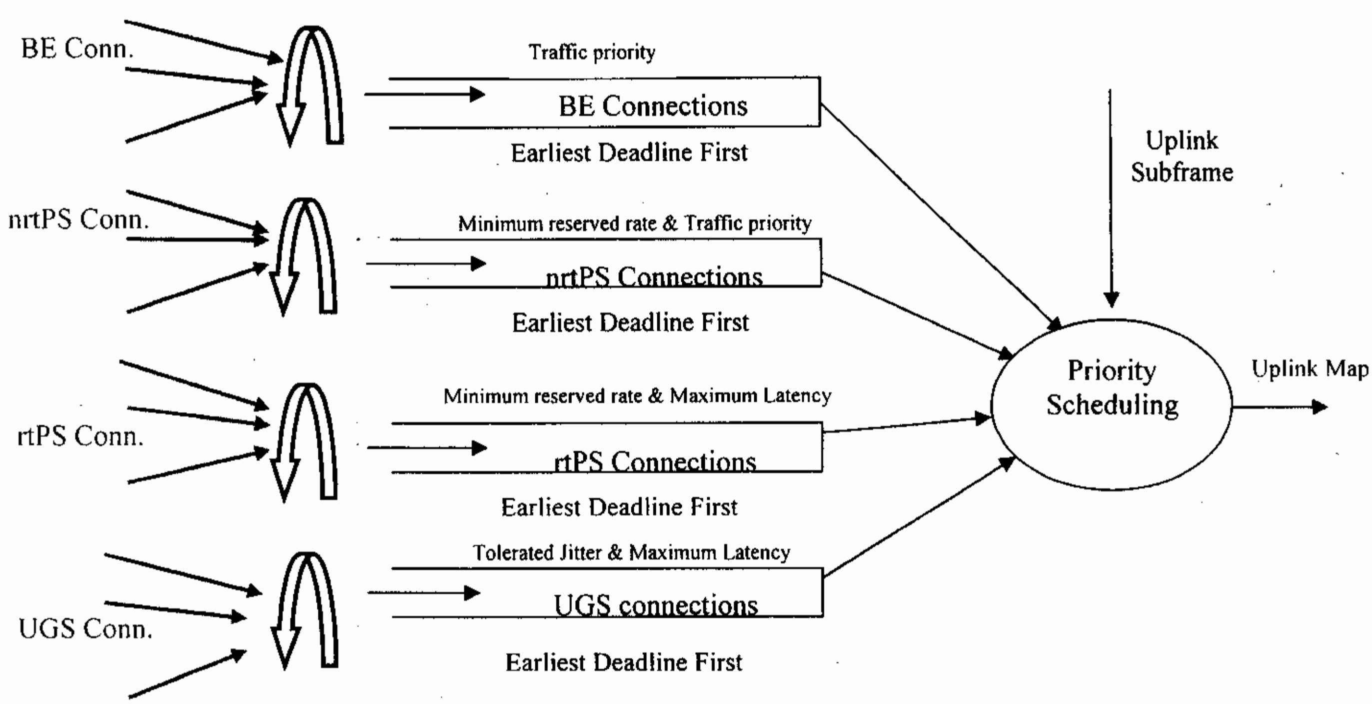

Figure 4.2: Uplink Bandwidth Scheduling

BS distributed uplink bandwidth among various SS by following strategy:

4.2.1.1 Bandwidth Allocation to UGS:

UGS supports real time data stream consisting of fixed sized data packet. Example of UGS is VoIP without silence suppression and T1. It is delay-sensitive and less jitter tolerance, so there is need a periodically bandwidth to maintain QoS. UGS cannot tolerate delay but can tolerate packet loss. Whenever UGS uplink flow registered with BS there is QoS parameter negotiated during connection setup and these parameters are Maximum Sustained Rate, Minimum Reserved Traffic Rate (equal to Maximum Sustained Traffic Rate in case of UGS flow), Tolerated Jitter, Request Transmission Policy and Maximum Latency for UGS flow. For example a UGS flow negotiated parameters are in this form:

Maximum Sustained Rate = 64 kbps
Tolerated Jitter = 1 ms
Maximum Latency =10 ms
UGS SDU Size = 200 bytes

According to negotiated parameter it shows that a UGS flow should required 64 kilo bit per second, so BS divides these requirements equally into all frames that transmit during one second. If we take frame duration 5 millisecond that's mean there is 320 bits (40 bytes) required for each frame (it is calculated by dividing *Maximum Sustained Rate* by total frame) and if minimum packet size is 1600 bits (200 bytes) so there is quite difficult to schedule packet in this limited bandwidth, so bandwidth useless. So BS allocates bandwidth to each UGS scheduling service according to *SDU size* that is 200 bytes.

BS first fulfill the requirements of all UGS flows as above describe mechanism. If there are number of UGS flow increasing, then *Tolerated Jitter* parameter used as a ordering parameter. If two flows are equal in Tolerated Jitter so we take *Maximum Latency* to break a tie.

BS also tracks the incoming traffic of each UGS flow. If UGS packets are coming after regular interval then BS fulfill the requirements according to predefined interval and if there is any variation involved in incoming traffic then BS increase and decrease the *Grant interval* to fulfill the requirements of UGS flow. This variation a BS can analyze on basis of interarrival time of packets for UGS flow.

4.2.1.2 Bandwidth Allocation to rtPS:

rtPS supports real time data stream consisting of variable sized data packet. Example of rtPS is MPEG (Moving Picture Experts Group). It is also delay-sensitive and less jitter tolerance, so there is need a sufficient bandwidth to maintain QoS. rtPS cannot tolerate delay but can tolerate packet loss. Whenever rtPS downlink flow registered with BS there is QoS parameter negotiated during connection setup and these parameters are Maximum Sustained Traffic Rate, Minimum Reserved Traffic Rate, and Maximum Latency for rtPS flow. For example a rtPS flow negotiated parameters are in this form

Maximum Sustained Rate = 1 Mbps
Minimum Reserved Rate = 512 kbps
Maximum Latency = 50 ms

According to negotiated parameter it shows that rtPS flow should required Minimum Reserved Rate 512 kilo bit per second, BS fulfills the requirements of rtPS flow whenever number of packets queued in its queue. So BS provides unicast polling request to each rtPS flows to meet the QoS Parameter.

After fulfilling the requirements of all UGS flows, BS first fulfill the BW request of different SS for rtPS flows. The parameter *Maximum Latency* used as an ordering parameter. If two flows are equal in Maximum Latency so we given priority to that flow that have less *Maximum Sustained Rate* and *Minimum Reserved Traffic Rate*.

4.2.1.3 Bandwidth Allocation to nrtPS:

nrtPS support delay-tolerant data streams consisting of variable data packets. Example of nrtPS is FTP (File Transfer Protocol). nrtPS can tolerate larger delay but cannot tolerate packet loss. Whenever this uplink flow registered with BS there is QoS parameter negotiated during connection setup and these parameters are Maximum Sustained Traffic Rate, Minimum Reserved Traffic Rate, Traffic Priority and Maximum Latency for nrtPS flow. For example a nrtPS flow negotiated parameters are in this form:

Maximum Sustained Rate = 1 Mbps
Minimum Reserved Rate = 512 kbps
Traffic Priority = 1

According to negotiated parameter it shows that a nrtPS flow should required Minimum Reserved Rate 512 kilo bit per second, BS fulfills the requirements of nrtPS flow whenever number of packets queued in its queue. So BS provides unicast request opportunities to each nrtPS flows to meet the QoS Parameter.

After fulfilling the requirements of all UGS flows, BS first fulfill the bandwidth request of rtPS flows. And then fulfill the requirements of all nrtPS flows as above describe mechanism. BS fulfills the requested bandwidth from among different SS for nrtPS flow and *Traffic Priority* parameter used as a ordering parameter. If two flows are equal in Traffic priority so we given priority to that flow that have allocate less *Maximum Sustained Rate* and *Minimum Reserved Traffic Rate* as compared to other nrtPS flows.

4.2.1.4 Bandwidth Allocation to BE:

BE supports data streams for which no minimum service level is required and therefore handle it on space available basis. Example of BE is HTTP (Hyper Text Transfer Protocol) Traffic. BE can tolerate larger delay but cannot tolerate packet loss. Whenever BE uplink flow registered with BS there is QoS parameter negotiated during connection setup and these parameters are Maximum Sustained Traffic Rate, Minimum Reserved Traffic Rate, and Maximum Latency for BE flow. For example a UGS flow negotiated parameters are in this form.

Maximum Sustained Rate = 1 Mbps
Traffic Priority = 1

According to negotiated parameter it shows that a BE flow Maximum Sustained Rate 1 mega bit per second, so BS try to best to fulfills the requirements of BE flow to also achieve the best QoS flow. After fulfilling the requirements of all UGS, rtPS, and nrtPS flows and then fulfill the requirements of all BE flows by allocating bandwidth according to *Maximum Sustained Rate* and Last Polling Time. The parameter *Traffic Priority* used as an ordering parameter. If two flows are equal in Traffic priority so we given priority to that flow that have allocate less bandwidth as compared to other and that is calculated from *Maximum Sustained Rate*.

4.2.2 SS Uplink Scheduler

Uplink scheduling has two main process, first process performed at BS side where it allocate bandwidth to each service flow that associated with a specific SS and BS granted bandwidth on GPSS basis. Second process starts at SS side as shown in fig 4.3, where SS responsibility to schedule packets from respective queue of UGS, rtPS, nrtPS and BE.

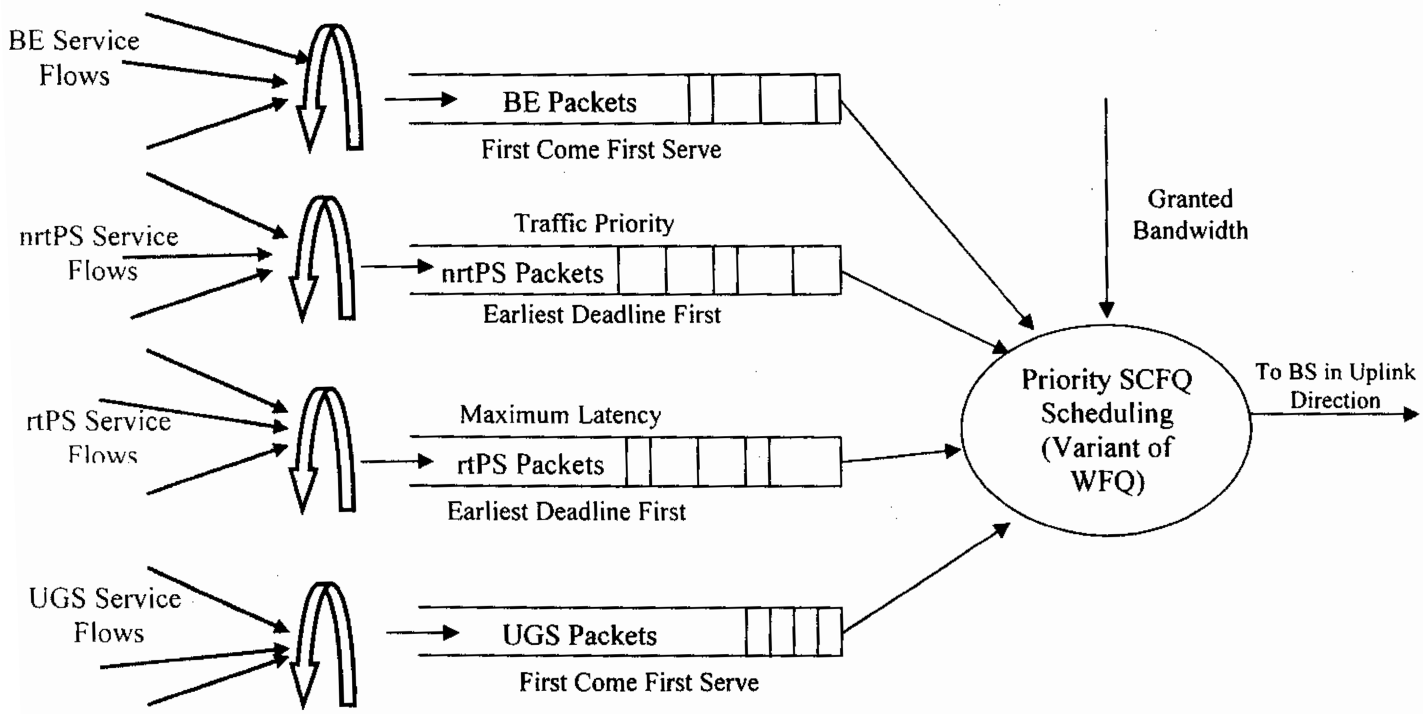

Fig 4.3: SS Uplink Scheduler

SS distributed uplink bandwidth among various flows by following strategy:

Firstly we schedule UGS packets because there is fixed bandwidth allocation from BS side and it is delay tolerant traffic. UGS packets are queued by first come first mechanism.

We use a variant of WFQ scheduling algorithm that is Self Clocked Fair Queuing (SCFQ) to schedule rtPS, nrtPS and BE packets. To avoid the costly computation of round number in WFQ, we used SCFQ. The worst case latency in this case could be

$$P_{max}/\Phi i + (N-1)\, P_{max}/R$$

rtPS Packets are queued by Earlier Deadline First mechanism and ordering parameter is Maximum Latency. nrtPS Packets are queued by Earlier Deadline First mechanism and ordering parameter is Maximum Latency and Traffic Priority. BE Packets are queued by First come First Serve mechanism.

A weight associated with each flow according to their priority and weight is calculated dynamically. Weight is calculated by the size of the queue and a constant (weight) priority associated with this flow and bandwidth distributed among flows by calculated weight.

A priority associated with each flow such like that:

P_rtPS = 6, P_nrtPS = 2, P_BE = 1
Allocated Bandwidth = B
Q_rtPS = R, Q_nrtPS = N Q_BE = E

A flow can get its proportion by following way:

rtPS_Bandwidth = (P_rtPS * Q_rtPS) / (Q_rtPS + Q_nrtPS + Q_BE) * B
nrtPS_Bandwidth = (P_nrtPS * Q_nrtPS) / (Q_rtPS + Q_nrtPS + Q_BE) * B
BE_Bandwidth = (P_BE * Q_BE) / (Q_rtPS + Q_nrtPS + Q_BE) * B

We are not using WFQ which is also known as packet by packet GPS (PGPS) by following reason:

- Due to iterated deletion problem.
- A WFQ scheduling algorithm has to update its round number on every packet arrival and departure. It has to do a fairly complex computation after every few microsecond.

SCFQ scheduling algorithm first find the finish number of each packet before its enque into respective queue and it also update the round number after each arrival and departure of packets. A formula used for calculated finish number given as

$$F_i^k = \max\,[F_i^{k-1}, CF] + L_i^k/\Phi_i$$

$\mathbf{F_i^k}$ = Finish time of packet k of flow i
$\mathbf{F_i^{k-1}}$ = Finish time of Last packet of flow i
CF = Finish number of the packet currently being served
$\mathbf{L_i^k}$ = Length of a packet k of flow i
$\mathbf{\Phi_i}$ = Round number of flow i

4.2.3 BS Downlink Bandwidth Management Module

This module as shown in fig 4.4 has responsibility for allocating bandwidth to each SS for downlink transmission. Bandwidth is allocated on per flow basis. This module performed its functionality at BS (Base Station) side. It's main responsibility to produce downlink map for all SS according to their bandwidth requirements. This module has all information about each service flow and current state of each service flow like status of each queue. To allocate downlink bandwidth to among different SS is easy then uplink bandwidth allocation because BS has all information for each scheduling service flow.

It's also responsibility to keeps fairness among different flows and Subscriber Station under overloaded condition. It assures the delay guarantee to UGS and rtPS flows. But it can not assure delay guarantee to nrtPS and BE flows as we increase the number of flows. As the number of flows increasing nrtPS and BE have got little bandwidth because its priority despite that they got such bandwidth that can be enough for scheduling packets.

Amount of bandwidth allocated to each SS in regular interval by the QoS parameters of a connection that associated with each connection.

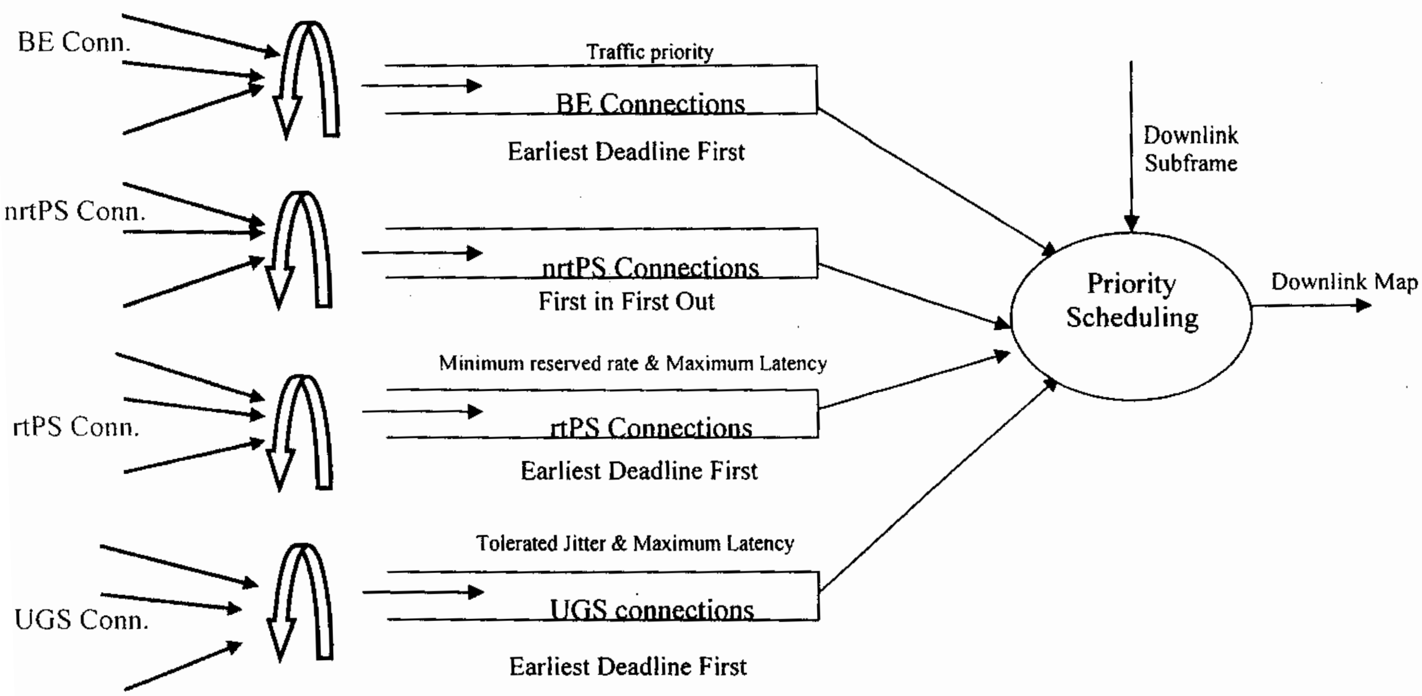

Fig 4.4: Downlink Bandwidth Management

BS distributed uplink bandwidth among various SS by following strategy:

4.2.3.1 Bandwidth Allocation to UGS:

UGS supports real time data stream consisting of fixed sized data packet. Example of UGS is VoIP without silence suppression and T1. It is delay-sensitive and less jitter tolerance, so there is need a periodically bandwidth to maintain QoS. UGS cannot tolerate delay but can tolerate packet loss. Whenever UGS downlink flow registered with BS there is QoS parameter negotiated during connection setup and these parameters are Maximum Sustained Rate, Minimum Reserved Traffic Rate (equal to Maximum Sustained Traffic Rate in case of UGS flow), Tolerated Jitter, Request Transmission Policy and Maximum Latency for UGS flow. For example a UGS flow negotiated parameters are in this form

Maximum Sustained Rate = 64 kbps
Tolerated Jitter = 1 ms
Maximum Latency =10 ms

According to negotiated parameter it shows that a UGS flow should required 64 kilo bit per second, so BS periodically assigned bandwidth to this flow to maintain the QoS. BS fulfills the requirements of UGS flow whenever a UGS packet queued in its queue. BS allocates bandwidth to flow according to packet size.

BS first fulfill the requirements of all UGS flows as above describe mechanism. If there are number of UGS flow increasing, then *Tolerated Jitter* parameter used as a ordering parameter. If two flows are equal in Tolerated Jitter so we take *Maximum Latency* to break a tie. Another things that consider during ordering flows is that a flow that more closer to Maximum Latency but it cross a deadline where packets of a flow cannot reach another side so its best way to drop packets despite allocating bandwidth.

4.2.3.2 Bandwidth Allocation to rtPS:

rtPS supports real time data stream consisting of variable sized data packet. Example of rtPS is MPEG (Moving Picture Experts Group). It is also delay-sensitive and less jitter tolerance, so there is need a sufficient bandwidth to maintain QoS. rtPS cannot tolerate delay but can tolerate packet loss. Whenever rtPS downlink flow registered with BS there is QoS parameter negotiated during connection setup and these parameters are Maximum Sustained Traffic Rate, Minimum Reserved Traffic Rate, and Maximum Latency for rtPS flow. For example a rtPS flow negotiated parameters are in this form

Maximum Sustained Rate = 1 Mbps
Minimum Reserved Rate = 512 kbps
Maximum Latency = 50 ms

According to negotiated parameter it shows that a rtPS flow should required 64 kilo bit per second, so BS fulfills the requirements of rtPS flow whenever number of rtPS packet queued in its queue. BS allocates bandwidth to flow according to packet size.

After fulfilling the requirements of all UGS flows, BS fulfill the requirement of rtPS flows as above describe mechanism. *Maximum Latency* parameter used as a ordering parameter. If two flows are equal in Maximum Latency so we given priority to that flow that have allocate less bandwidth and it calculated from *Maximum Sustained Rate* and *Minimum Reserved Traffic Rate*.

4.2.3.3 Bandwidth Allocation to nrtPS:

nrtPS support delay-tolerant data streams consisting of variable data packets. Example of nrtPS is FTP (File Transfer Protocol). nrtPS can tolerate larger delay but cannot tolerate packet loss. Whenever this uplink flow registered with BS there is QoS parameter negotiated during connection setup and these parameters are Maximum Sustained Traffic Rate, Minimum Reserved Traffic Rate, Traffic Priority and Maximum Latency for nrtPS flow. For example a nrtPS flow negotiated parameters are in this form

Maximum Sustained Rate = 1 Mbps
Minimum Reserved Rate = 512 kbps
Traffic Priority = 1

According to negotiated parameter it shows that a nrtPS flow should required 128 kilo bit per second, so BS fulfills the requirements of nrtPS flow whenever number of nrtPS packet queued in its queue. BS allocates bandwidth to flow according to packet size.

After fulfilling the requirements of all UGS flows, BS fulfill the requirement of rtPS and then fulfill the requirements of all nrtPS flows as above describe mechanism. The parameter *Traffic Priority* used as an ordering parameter. If two flows are equal in Traffic priority so we given priority to that flow that have allocate less bandwidth as compared to other and that is calculated from *Maximum Sustained Rate* and *Minimum Reserved Traffic Rate*.

4.2.3.4 Bandwidth Allocation to BE:

BE supports data streams for which no minimum service level is required and therefore handle it on space available basis. Example of BE is HTTP (Hyper Text Transfer Protocol) Traffic. BE can tolerate larger delay but cannot tolerate packet loss. Whenever BE uplink flow registered with BS there is QoS parameter negotiated during connection setup and these parameters are Maximum Sustained Traffic Rate, Minimum Reserved Traffic Rate, and Maximum Latency for BE flow. For example a UGS flow negotiated parameters are in this form.

Maximum Sustained Rate = 1 Mbps And Traffic Priority = 1

According to negotiated parameter it shows that a BE flow should required 64 kilo bit per second, so BS fulfills the requirements of BE flow whenever number of BE packet queued in its queue. BS allocates bandwidth to flow according to packet size.

After fulfilling the requirements of all UGS, rtPS, nrtPS flows and then fulfill the requirements of all BE flows as above describe mechanism. In second stage BS allocate downlink bandwidth to all SS. Service flows that have not get bandwidth in current frame got in next.

4.2.4 BS Downlink Scheduler

Downlink scheduling as shown in fig 4.5 has two main process, first process performed at BS side where it allocate bandwidth to each service flow that associated with a specific SS and BS granted bandwidth on GPSS basis. Second process also starts at BS side where BS Downlink Scheduler responsibility to schedule packets from respective queue of UGS, rtPS, nrtPS and BE according to allocate bandwidth. We use a variant of WFQ scheduling algorithm that is Self Clocked Fair Queuing (SCFQ). We are not using WFQ which is also known as packet by packet GPS (PGPS) by following reason

- Due to iterated deletion problem.
- A WFQ scheduling algorithm has to update its round number on every packet arrival and departure. It has to do a fairly complex computation after every few microsecond.

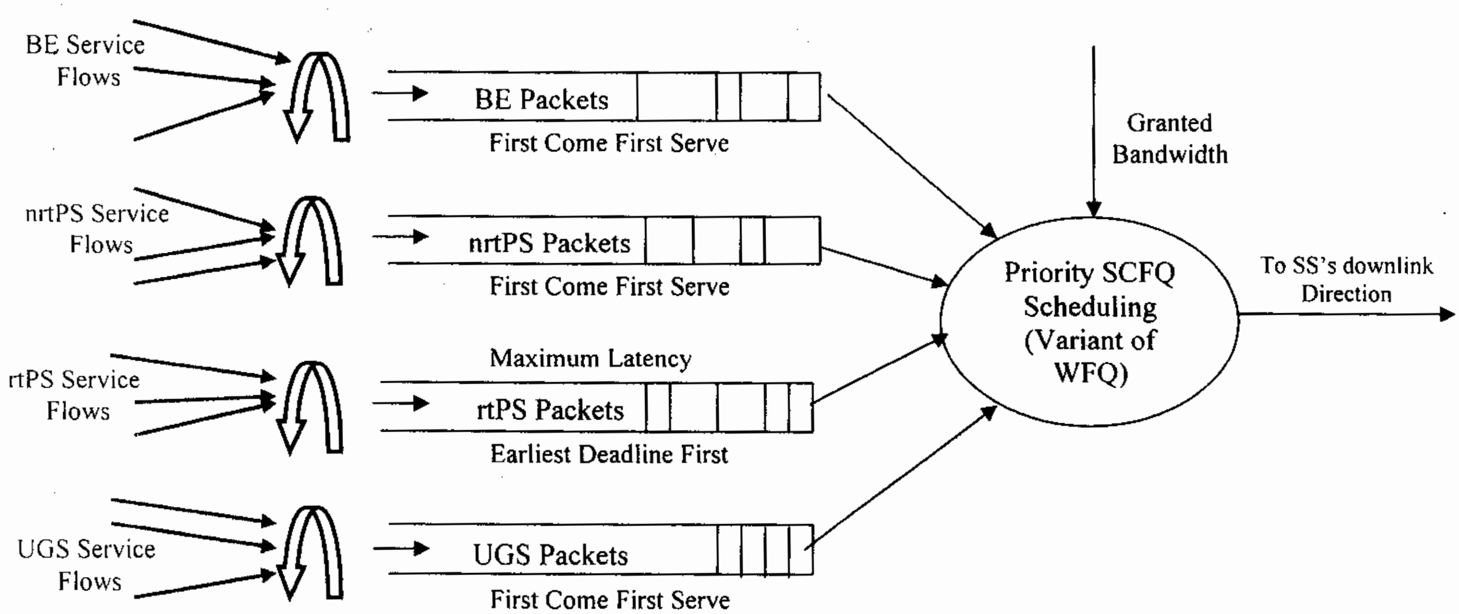

Fig 4.5: BS Downlink Scheduler

BS distributed uplink bandwidth among various flows by following strategy:

Firstly we schedule UGS packets because there is fixed bandwidth allocation so that it can meet QoS efficiently.

After scheduling UGS packets, the remaining bandwidth allocated among rtPS, nrtPS and BE flows. These packets are schedule by SCFQ scheduling algorithm that find the finish number of each packets before its enque into respective queue and it also update the round number after each arrival and departure of packets. Weight is calculated by mechanism described in SS Uplink Scheduler.

4.2.5 Packet Ordering Module

As mentioned by the author [18] to allocate m bytes such that maximum number of packets can be transmitted. It is knapsack problem and that is NP–complete. So he provides a suboptimal algorithm.

1) Arrange the packets in increasing packets size.
2) Allocating bandwidth from the first packets.

It's complexity is O(M) hence we used in real time by the Subscriber Station. According to author theoretical concept we have designed such module that sorting the packets queue according to packets size.

It's key responsibility of this module to overcome the inefficient use of bandwidth. There is lot of packets in a queue that can be schedule under available bandwidth but packets in front of queue is size greater than available bandwidth, so we re-order the packets in ascending order with respect to its size. For example 1003,546,789,900,207 and etc after reordering it look like this 207,546, 789, 900 and 1003.

We cannot perform this functionality on UGS queue because its packet size constant. We also cannot perform this functionality on rtPS queue because its represent audio and video traffic so if there is packets that size greater than packets, so after every re-order queue it might be chance that it cannot be schedule after number of re-ordering then that packets are useless.

We can perform this functionality on nrtPS and BE flow because they can tolerate more delay than rtPS. In case of fragmentation enable this module functionality disable. This process performed before the functionality of BS downlink scheduler and SS uplink scheduler.

4.2.6 Fragmentation Module

This module main responsibility to allow efficient use of granted bandwidth relative to the QoS requirements of a connection. We are not fragmented the UGS connection because there is fixed allocation of bandwidth. This process performed before BS scheduler and SS scheduler to schedule the packets.

Its main task to fragment the packets that size is greater than fragment allocated size. In this case header overhead increases but it can be compromise with efficient utilization of bandwidth that allocates to each SS.

Chapter 5

NS-2 Implementation Details

5. NS-2 Implementation Details

In this chapter we discuss about the implementation details of our architecture. We have carried out our simulation, to analyze the performance of IEEE 802.16 scheduling architecture using NS-2 (version 2.29) [26] simulator. There are two patches available for NS-2 simulating IEEE 802.16 MAC Layer one is developed by NIST [27] and other is developed by Department of Computer Science and Information Engineering and Department of Electrical Engineering Chang Gung University, Taiwan [28]. Before discussing the implementation details we describe the NS-2 capability, its potential benefits and its limitation.

5.1 NS-2 Simulators

NS are an object oriented simulator, written in C++, with an OTcl interpreter as a front end. The simulator supports a class hierarchy in C++ (also called the compiled hierarchy in this document), and a similar class hierarchy within the OTcl interpreter (also called the interpreted hierarchy in this document). The two hierarchies are closely related to each other; from the user's perspective, there is a one-to-one correspondence between a class in the interpreted hierarchy and one in the compiled hierarchy. The root of this hierarchy is the class Tcl Object. Users create new simulator objects through the interpreter; these objects are instantiated within the interpreter, and are closely mirrored by a corresponding object in the compiled hierarchy.

NS uses two languages because simulator has two different kinds of things it needs to do. On one hand, detailed simulations of protocols require a systems programming language which can efficiently manipulate bytes, packet headers, and implement algorithms that run over large data sets. For these tasks run-time speed is important and turn-around time (run simulation, find bug, fix bug, recompile, re-run) is less important. On the other hand, a large part of network research involves slightly varying parameters or configurations, or quickly exploring a number of scenarios. In these cases, iteration time (change the model and re-run) is more important. Since configuration runs once (at the beginning of the simulation), run-time of this part of the task is less important. Ns provide an environment where we can simulate real network and analysis the behavior of different network parameters.

Having two languages raises the question of which language should be used for what purpose. OTcl used for configuration, setup, "one-time" stuff and if you can do what you want by manipulating existing C++ objects. C++ used for If you are doing anything that requires processing each packet of a flow and If you have to change the behavior of an existing C++ class in ways that weren't anticipated

It can be explained by example; links are OTcl objects that assemble delay, queuing, and possibly loss modules. If your experiment can be done with those pieces, it's easy for you. If instead you want do something fancier (a special queuing discipline or model of loss or new algorithm), then you'll need to create new C++ object.

There are a number of classes defined in ns–2. There are six classes that are more frequently used in ns: The Class Tcl contains the methods that C++ code will use to access the interpreter. The class TclObject is the base class for all simulator objects that are also mirrored in the compiled hierarchy. The class TclClass defines the interpreted class hierarchy, and the methods to permit the user to instantiate TclObjects. The class TclCommand is used to define simple global interpreter commands. The class EmbeddedTcl contains the methods to load higher level built-in commands that make configuring simulations easier. Finally, the class InstVar contains methods to access C++ member variables as OTcl instance variables. Figure 5.1 describes the simplified view from user perspective.

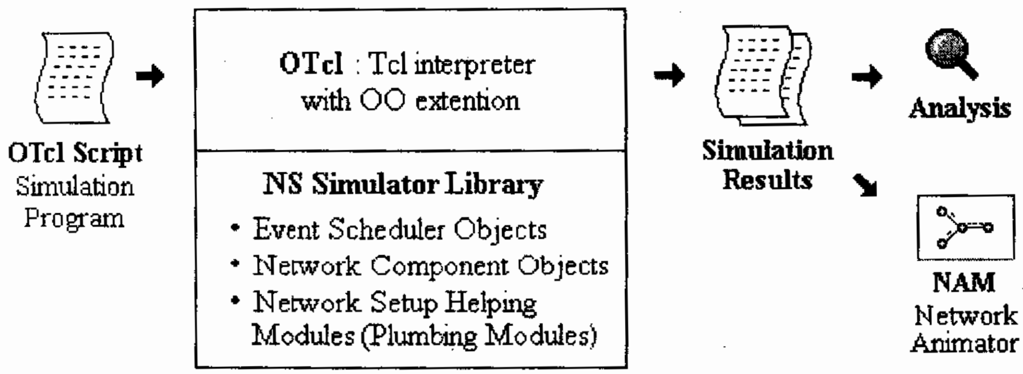

Figure 5.1: Simplified User's View of NS-2 [25]

5.1.1 TCL interpreter:

TclCL is the language used to provide a linkage between C++ and OTcl. Toolkit Command Language (Tcl/OTcl) scripts are written to set up/configure network topologies. TclCL provides linkage for class hierarchy, object instantiation, variable binding and command dispatching. OTcl is used for periodic or triggered events. The Event Scheduler and Basic network component objects is written and compiled with C+

These compiled objects are made available to the OTcl interpreter through an OTcl linkage that creates a matching OTcl object for each of the C++ objects and makes the control functions and the configurable variables specified by the C++ object act as member functions and member variables of the corresponding OTcl object. It is also possible to add member functions and variables to a C++ linked OTcl object as shown in fig 5.2.

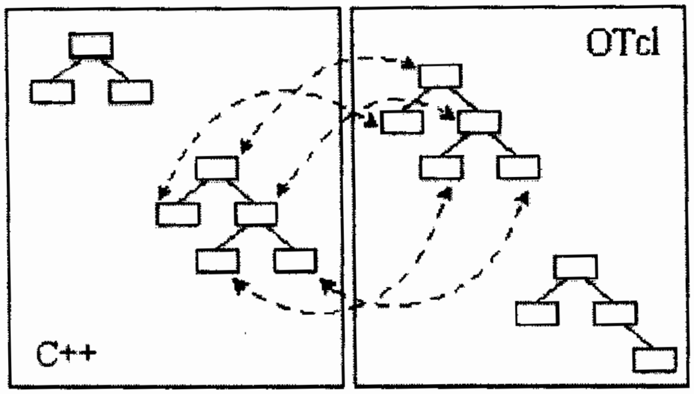

Figure 5.2: C++ and OTcl: The Duality [25]

5.1.2 Network Animator (NAM)

NAM, network animator, is used for visualization of network scenario. It provides visualization of

- Packet flows, different packets can be colored.
- Nodes' native packets queue.
- Packets which are dropped.

For wireless network simulation, NAM plays an important role because it can help that whether a node is within range of another node. NAM is very important to analysis the mobile nodes' movements during simulation.

Following OTcl procedures are used to set node attributes, they are methods of the class Node:

```
$node color [color]                    ; # sets color of node
$node shape [shape]                    ; # sets shape of node (circular by default)
$node label [label]                    ; # sets label on node
$node label-color [lcolor]             ; # sets color of label
$node label-at [ldirection]            ; # sets position of label
$node add-mark [name] [color] [shape]  ; # adds a mark to node
$node delete-mark [name]               ; # deletes mark from node
```

Nam is a Tcl/TK based animation tool for viewing network simulation traces and real world packet trace data. The design theory behind NAM was to create an animator that is able to read large animation data sets and be extensible enough so that it could be used indifferent network visualization situations. Under this constraint NAM was designed to read simple animation event commands from a large trace file. In order to handle large animation data sets a minimum amount of information is kept in memory. Event commands are kept in the file and reread from the file whenever necessary.

The first step to use NAM is to produce the trace file. The trace file contains topology information, e.g., nodes, links, as well as packet traces. Usually, the trace file is generated by ns. During an ns simulation, user can produce topology configurations, layout information, and packet traces using tracing events in ns. However any application can generate a NAM trace file. When the trace file is generated, it is ready to be animated by NAM. Upon startup, NAM will read the trace file, create topology, pop up a window, do layout if necessary, and then pause at time 0. Through its user interface, NAM provides control over many aspects of animation.

5.1.2.1 Nam Command Line Options

nam [-g <geometry>] [-t <graphInput>] [-i <interval>] [-j <startup time>]
[-k <intial socket port number>] [-N <application name>] [-c <cache size>]
[-f <configuration file>] [-r initial animation rate]
[-a] [-p] [-S]
[<tracefile(s)>]

Command Line Options

-g Specify geometry of the window upon startup.
-t Instruct nam to use tk graph, and specify input file nam for tk graph.
-i [Information for this option may not be accurate] Specify rate (real) milliseconds as the screen update rate. The default rate is
-N Specify the application name of this nam instance. This application name may later be used in peer synchronization.
-c The maximum size of the cache used to store 'active' objects when doing animating in reverse.
-f Name of the initialization files to be loaded during startup. In this file, user can define functions which will be called in the trace
-a Create a separate instance of nam.
-p Print out nam trace file format.
-S Enable synchronous X behavior so it is easier for graphics debugging. For UNIX system running X only.

5.1.2.2 User Interface

Starting up nam will first create the nam console window as shown in fig 5.3. You can have multiple animations running under the same nam instance. At the top of all nam windows is a menu bar. For the nam console there are 'File' and 'Help' menus. Under the 'File' there is a 'New' command for creating a ns topology using the nam editor (under construction) , an 'Open' command which allows you to open existing trace files, a 'WinList' command that popup a window will the names of all currently opened trace files, and a 'Quit' command which exits nam. The 'Help' menu contains a very limited popup help screen and a command to show version and copyright information.
Once a tracefile has been loaded into nam (either by using the 'Open' menu command or by specifying the tracefile on the command line) an animation window will appear. It has

a 'Save layout' command which will save the current network layout to a file and a 'Print' command which will print the current network layout.

The 'Views' menu has 4 buttons as shown in fig 5.3:

- New view button: Creates a new view of the same animation. User can scroll and zoom on the new view. All views will be animated synchronously.
- Show monitors checkbox: If checked, will show a pane at the lower half of window, where monitors will be displayed.
- Show auto layout checkbox: If checked, will show a pane at the lower half of window, which contains input boxes and a button for automatic layout adjustments. This box will not be enabled when using link orientation layouts.
- Show annotation checkbox: If checked, will show a list box at the lower half of window, which will be used to list annotations in the ascending order of time.
- Below the menu bar, there is a control bar containing 6 buttons, a label, and a small scrollbar (scale). They can be clicked in any order. We will explain them from left to right.
- Button 1 («) - Rewind. When clicked, animation time will go back at the rate of 25 times the current screen update rate.
- Button 2 (<) - Backward play. When clicked, animation will be played backward with time decreasing.
- Button 3 (square) - Stop. When clicked, animation will pause.
- Button 4 (>) - Forward play. When clicked, animation will be played forward with time increasing.
- Button 5 (») - Fast Forward. When clicked, animation time will go forward at the rate of 25 times the current screen update rate.
- Button 6 (Chevron logo) - Close current animation window.

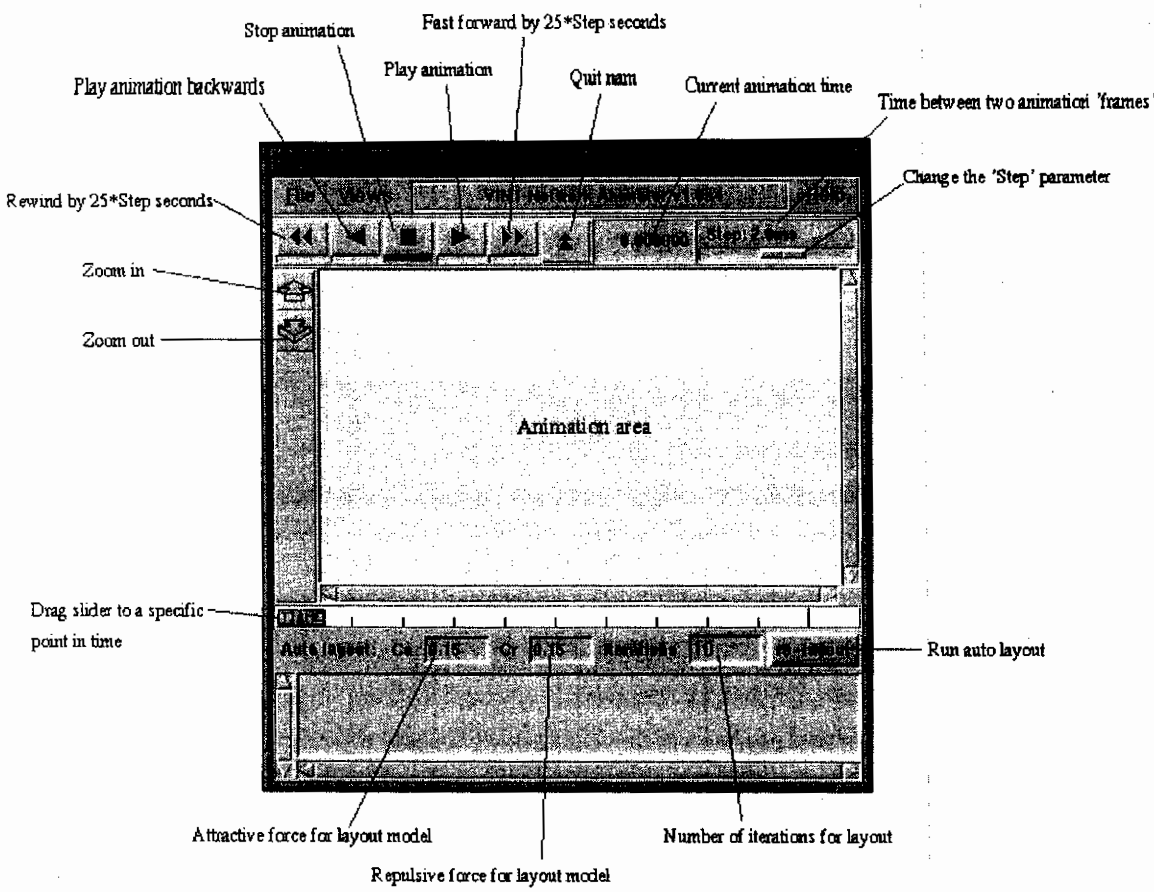

Figure 5.3: NS-2 User interface [25]

Time label - Show the current animation time (i.e., simulation time as in the trace file). Rate Slider - Controls the screen update rate (animation granularity). The current rate is displayed in the label above the slider.

Below the first control bar, there is Main Display, which contains a tool bar and a main view pane with two panning scroll bars. All new views created by menu command 'Views/New view' will have these three components. The tool bar contains two zoom buttons. The button with an up arrow zooms in, the button with a down arrows zooms out. The two scroll bars are used to pan the main animation view. Clicking the left button on any of the objects in the main view pane will pop up a information window. For packet and agent objects, there is a 'monitor' button in the popup window. Clicking that button will bring out the monitor pane (if it is not already there), and add a monitor to the object. For link objects, there will be a 'Graph' button. Clicking on that button will bring up another popup window, where users can select between drawing a bandwidth utilization graph or drawing a link loss graph of one simplex edge of the duplex link.

Below the user interface objects we have discussed so far, there may or may not be a Monitor pane, depending on whether the checkbox 'Views/Show monitors' is set. (The default is unset). All monitors will be shown in this pane. A monitor looks like a big button in the pane. Currently only packets and agents may have monitors.

A packet monitor shows the size, id, and sent time. When the packet reaches its destination, the monitor will still be there, but will say that the packet is invisible. An agent monitor shows the name of the agent, and if there are any variable traces associated with this agent, they will be shown there as well.

Below the monitor pane (or in its place if the monitor pane isn't there), there is a Time Slider. It looks like a scaled ruler, with a tag 'TIME' which can be dragged along the ruler. It is used to set the current animation time. As you drag the 'TIME' tag, current animation time will be displayed in the time label in the control bar above. The left edge of the slider represents the earliest event time in the trace file and the right edge represents the last event time. Clicking left button on the ruler (not on the tag) has the same effect as Rewind or Fast Forward, depending on the clicking position. The Automatic Layout Pane may be visible or hidden. If visible, it is below the time slider. It has three input boxes and one relay out button. The labeled input boxes let user adjust two automatic layout constants, and the number of iterations during next layout. When user press ENTER in any of the input boxes, or click the 'relayout' button, that number of iterations will be performed. Refer to the AUTOMATIC LAYOUT section for details of usage. The bottom component of the nam window is a Annotation List box, where annotations are displayed. Anannotation is a (time, string) pair, which describes a event occurring at that time. Refer to ns (1) for functions to generate annotations. Double-clicking on an annotation in the listbox will bring nam to the time when that annotation is recorded. When the pointer is within the listbox, clicking the right button will stop the animation and bring up a popup menu with 3 options: Add, Delete, Info. 'Add' will bring up a dialog box with a text input to add a new annotation entry which has the current animation time.

5.1.3 Trace Data Analyzers

There are number of ways to analyze trace file produces from simulation. There are following four ways that are mostly follow to analyze the trace file.

1) **XGraph**

   It is an X-Windows application that includes:

   - Interactive plotting and graphing
   - Animation and derivatives

   To use XGraph in NS-2 the executable can be called within a TCL Script. This will then load a graph displaying the information visually displaying the information of the trace file produced from the simulation.

2) **TraceGraph**

   It is a trace file analyzer that runs under Windows, Linux and UNIX systems and requires Matlab 6.0 or higher. TraceGraph supports the following trace file formats.

- Wired
- Satellite
- Wireless (old and new trace)

3) Awk scripts with Microsoft word

It is shell scripting language that extracts data from trace file according to the requirement of user and arranges these extracted data. Then Microsoft Excel plotted graph according to data. It can support any trace format.

4) User built–in code

It is a method where a user builds its own code to extract and compute data and show in graphical format. And this code developed in any language just like c++ and java. It can support any trace format.

5.2 Characteristics of NS-2

NS-2 implements the following features

1) Router queue Management Techniques DropTail, RED, CBQ,
2) Multicasting
3) Simulation of wireless networks
   - Developed by Sun Microsystems + UC Berkeley (Daedalus Project)
   - Terrestrial (cellular, adhoc, GPRS, WLAN, BLUETOOTH), satellite
   - IEEE 802.11 can be simulated, Mobile-IP, and adhoc protocols such as DSR, TORA, DSDV and AODV.
4) Traffic Source Behaviour- www, CBR, VBR
5) Transport Agents- UDP/TCP
6) Routing
7) Packet flow
8) Network Topology
9) Applications- Telnet, FTP, Ping
10). Tracing Packets on all links/specific links

5.3 Operating Systems for NS-2

Ns can be used on the following platforms:

- UNIX (Free BSD, SunOS, Solaris).
- Linux (RedHat 9, Enterprise Edition, FEDORA 4)
- Microsoft Windows

However for windows, Cygwing emulator is required for ns. The most favorable operating system for ns is Linux/Unix operating system.

5.4 Potential Benefits

1) *Economy and ease of installation* are important factors while using ns-2 simulations. Because physical simulation demands lot of capital and hard work.
2) *Speed* is also an important factor, forces us to ns-2. Because physical simulation is very time consuming. Also modifications in ns-2 are easier and faster than actual scenario.
3) *Less space* is required as compared to physical networks. Because in physical networks, one have to put a lot of machines, power cables and other network components while in simulation one have to only installed simulator on a machine.
4) *Open source and free* software: There are also other simulators like OPNET, which is very expensive. The research version of OPNET costs more than Rs. 320000. While NS-2 is freely available on Internet.

5.5 Limitations

1) NS-2 offers above mentioned exciting features but it is very difficult to work in NS-2 for new user.
2) Ns–2 is memory extensive simulator, so there is lots of problem arises during simulated large network and as the number of nodes are increasing processing time also increasing.
3) We have considerable confidence in ns, ns is not a polished and finished product, but the result of an ongoing effort of research and development.
4) Bugs in the ns–2 software are still being discovered and corrected.
5) Users of ns are responsible for verifying for themselves that their simulations are not invalidated by bugs.
6) Patience to debug NS source code when needed.
7) More complex simulations may need modification to NS source code.
8) Debugging process are complicated so there is quite knowledge of c++ and Otcl required.

5.6 IEEE 802.16 Patch Details

Many protocol modules have been implemented in the ns-2, the IEEE 802.16 broadband wireless access networks (BWANs) or WiMAX module has also contributed by Chang Gung University, Kweishan, Taoyuan, Taiwan [17]. They design and implementation of the WiMAX module based on the IEEE 802.16 standard with the point-to-multipoint (PMP) mode for the ns-2 as shown in fig 5.4. This module comprises fundamental functions of the service-specific convergence sublayer (CS), the MAC common part sublayer (CPS), and the PHY layer.

The 802.16-based WiMAX module named as the Mac802 16 class is in accordance with the specifications of the IEEE 802.16-2004 standard [1] and based on the ns-2 version 2.29 [11]. All modules are designed by using object oriented programming language C++

and modeled as several classes. The relationship between the WiMAX module and legacy ns-2 modules is based on the original network component stack of the ns-2 as shown in Fig 2. It illustrates the type of objects for the traffic generating agent (TGA), the link layer (LL), the interface queue (IFQ), the MAC layer (WiMAX module), and the PHY layer (Channel). First, the TGA is considered simply as an application level traffic generator that generates VoIP, MPEG, FTP, HTTP traffic, and so on. These traffic are classified into five different types of service, the UGS, rtPS, ertPS, nrtPS, and BE, each with its own priority. All packets will be transferred to different types of priority queues according to their service types by using CS layer SFID-CID mapping mechanism. The data packets in these queues are treated as MSDUs and will be selected to pass into the WiMAX module in a round robin manner. While the WiMAX module in the SS receives the MSDUs from the Queue object, the MAC management component will initiate the ranging process to enter the WiMAX system or to transmit the MSDUs according to the scheduled time obtained from UL-MAP. Once the process has been successfully finished in the MAC layer, the Network Interface will add a propagation delay and broadcast in the air interface. The Channel object we used is the WirelessPhy class.

The WiMAX module also receives packets from the air interface passed from other nodes, and then it determines whether the packet is a control packet or not. If the packet is a control packet, the MAC management object will take corresponding procedures according to the control packet. If not, the packet will be passed to LL object after the defragmentation process. Finally, the TGA will receive the packets from the LL object. The BS and SS are recognized by its corresponding numbers, which are specified in the OTcl object.

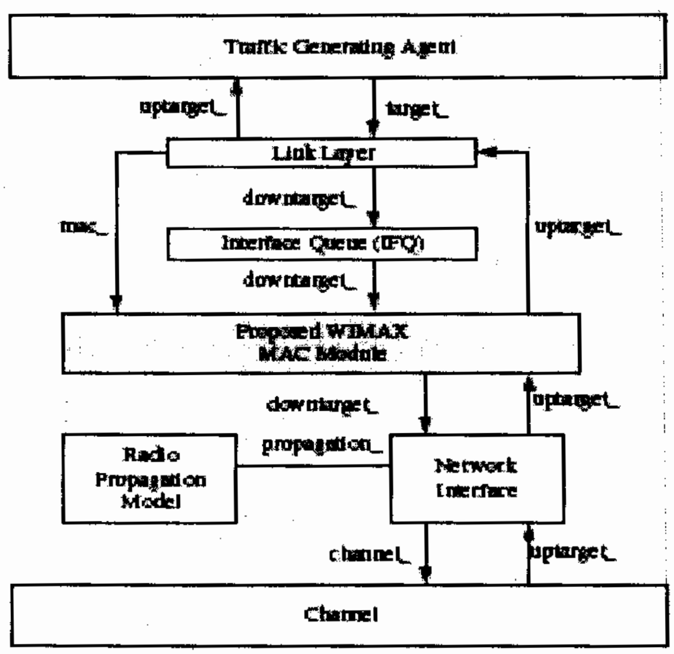

Figure 5.4: The relationship between the WiMAX module and legacy ns-2 modules [17]

The components of the WiMAX module are as the following.

5.6.1. The CS Sublayer

The CS sublayer has two major functions: 1) transforming the IP address (from the upper layer) into several SFIDs or the reverse transformation (from SFID to IP address), 2) recording the mapping between a SFID and a transport CID (TCID). These functions enable the MAC layer to keep the essential information of the upper layer SDUs about their QoS parameters and destination addresses.

5.6.1.1 IP-SFID mapping

The SDUs, which come from the upper layers, will contain the corresponding destination addresses and service types. An IP-SFID mapping function should record and classify the characteristics of the requesting packets for future IP-MAC mapping. A SFID is used for either the DL transmission with QoS parameters reference or the UL transmission with IP lookup.

5.6.1.2 SFID-TCID mapping

The SFID-TCID mapping is a main function of the CS sublayer for SFID to TCID mapping, which defines the QoS class of the service flow associated with the connection. For the UL traffic, the SS will send a bandwidth request header with the primary CID to the BS for data transmission by invoking the BandwidthRequest() function if it does not obtain a TCID. The SS can add, change, or delete its obtained bandwidth via bandwidth management messages: dynamic service addition, change, and deletion (DSA, DSC, and DSD) later. For the DL traffic (from Internet), the insert SFID() function (in the BS) will determine whether the SS obtained a TCID. If not, this function will generate an unused TCID for the SS or transfer the MSDU into the corresponding QoS queue. Since the connection of the WiMAX is bi-direction and each direction has at least five priorities (UGS, rtPS, ertPS,nrtPS, and BE), a SS may use all services (ten SFIDs including five for UL and five for DL) during its usage time. Therefore, The length of the SFID and TCID are 32-bit and 16-bit (2m+1–0xFEFE) long, where m is a variable depending on the setting of the operator, respectively. After the mapping operation, the SFID-TCID mapping will be recorded in both sides of the BS and the SS.

5.6.2 The MAC Sublayer

The MAC CPS sublayer is the main part of the MAC and maintains the MAC operations and management messages of the system. The management messages such as DCD, UCD, DL-MAP, UL-MAP, DSA, DSC, DSD, RNG-REQ, RNG-RSP, and so forth are generated in this sublayer. The main body of the MAC CPS is constructed by a Mac802 16 class, which contains several independent functions such as Ranging(), Fragmentation(), BandwidthRequest(), and so forth. The detailed functions are described as follows.

5.6.2.1 Ranging

The initial ranging process is the first step in our module for a SS to enter the network. First, a new SS has to scan for the DL channel and establish synchronization with the BS. After synchronizing with the BS, the SS will obtain transmit parameters from the UCD message, which is periodically generated by the BS, to recognize the channel information for transmission. While an unregistered SS receives a packet from the Queue object, it will start the ranging process to notice the BS when entering the system. The SS sends the RNG-REQ management message in the ranging interval which is defined in the UL-MAP issued from the BS with quadrature phase shift keying (QPSK) 1/2 coding rate modulation for contending the entry of the system. The entering process follows the random backoff mechanism with an initial backoff countdown interval of (0,CWmin− 1) where the CWmin is the minimum contention window size and is equal to 32. At ranging period, the backoff time is uniformly chosen in the range (0,CWmin − 1). After each unsuccessful transmission, the CWmin is doubled up to a maximum value CWmax = 2mCWmin. The CWmax value is set to 1024 as defined in the standard. The SS uses CID value of zero to send RNG-REQ and starts a timer to wait for the RNG-RSP message from the BS. These processes are operated in two functions rng req() and rng rsp(). If the SS receives the RNG-RSP before the timer expiration, the ranging process is successful. Otherwise, the SS will select a new backoff window size for a new ranging process. The collision detection of the RNG-REQ is set by the receiver timer. If more than one RNG-REQ message is sent within the time interval, these RNG-REQs are treated as collision. Otherwise, this message is successful. When the BS receives a RNG-REQ message from the SS, it will decide to let the SS join the networks or not. After determination, the BS will reply a RNG-RSP message following the DL MAP among the next several superframes. The RNG-RSP contains an unique basic CID and a primary CID to the SS for future communications.

5.6.2.2 MAC management

Five kinds of messages, DCD, UCD, DL-MAP, UL-MAP, and bandwidth request (BR), are used in this function. Each message has its own management message type and they can be discriminated by each other. The DCD includes the management message type, downlink channel ID, TLV encoding information for the overall channel, and the downlink burst profile. The DCD channel encoding is composed of the TLV specific, which includes all the channel information, such as the DL burst profile (may appear more than once), the frame duration, PHY type, power adjustment rule, channel number, the transmit/receive transition gap (TTG) and the receive/transmit transition gap (RTG), the frequency of the downlink center frequency, the BSID, the frame duration code, and the frame number.

The important point of the DCD is the downlink burst profile, which includes the DIUC (in order to map to the DL-MAP) and the TLV encoded information. In the TLV encoded information DCD burst profile, FEC code type, DIUC mandatory exit threshold, as well as the DIUC minimum entry threshold. The FEC code type can indicate the modulation type of the burst. The DIUC mandatory exit threshold will define the range of the CINR and indicate the DIUC that can no longer be used, and where this change to a more robust DIUC is required. Similarly, the DIUC minimum entry threshold is the minimum requirement for CINR in order to start using this DIUC.

The UCD includes management message type, ranging backoff start, ranging backoff end, request backoff start, request backoff end, and the TLV encoding information for the overall channel. The significance of UCD is the TLV encoding information for the overall channel. It constructs the uplink burst profile. Same as the downlink burst profile, the uplink burst profile also contains the FEC code type and modulation type. The ranging data ratio is also included in the uplink burst profile, which means the reducing factor between the power used for this burst and power used for CDMA ranging. The last TLV encoding information in uplink burst profile is the normalized C/N override. This is a list of numbers, where each number is encoded by one nibble and interpreted as a signed integer. All of the MAC messages mentioned above are triggered by specific timers.

First, all traffic flows are generated by the traffic generating agent as shown in Fig. 2. These data flows will be treated as the basic packet object defined in the ns-2. Then these packets will come to the Mac802 16 through the interfacequeue (IFQ) and be treated as the MSDUs. Oncethe MSDU comes, the insert SFID() is invoked in order to classify the MSDUs into several groups, such as UGS, rtPS, ertPS, nrtPS, and BE. If the SFID is set as active, then the MSDU will be transferred into the queue labeled as a TCID number; otherwise, the insert SFID() will assign a TCID and set the corresponding SFID as active.

After the service flow classification, these MSDUs will be transferred into their corresponding queue and be held to be served. In this stage, either it is in the DL or the UL channel, the BS has to manage the bandwidth by invoking the bandwidth management function Bandwidth Management(), which plays the call admission control (CAC) mechanism of each SS in the UL and the DL bandwidth management. In the implementation of the module, the CAC mechanism follows the first-in-first-served (FIFS) basis to admit the coming requests. If the bandwidth is enough for serving the request, this request will be allowed to enter the system; otherwise, it will be denied by CAC immediately.

The SS may request to perform bandwidth request with BS by using BandwidthRequest() and the related parameters, e.g., CID, type, encryption control (EC), and header type (HT), and so on. The HT field of the MAC header is set to one for indication of a BR. The SS will calculate its required bandwidth and set the BR field (19 bits) to a corresponding bandwidth (1–524287 bytes). Afterward the SS will continue to observe the upcoming UL-MAPs to check whether its request is successful or not. The bandwidth request process follows the random backoff approach as described in ranging process. Once the requested bandwidth is admitted by the BS, the SS can invoke GrantManagementSubhdr() for future grant management if SS needs more bandwidth. The GrantManagementSubhdr() will generate a subheader for an indication of piggyback request, poll-me, or slip indicator of the active TCID.

5.6.2.3 Priority queue

In the BS or SS, the packets that come from the upper layer will be prior delivered to Priority(). According to the TCID and its service type: UGS(5), rtPS(4), ertPS(3), nrtPS(2), BE(1), the Priority() will make a corresponding priority classification. The Priority() generates an exclusive queue to store these packets based on its TCID. Finally, packets will be treated as MSDUs and be segmented by different queue function, e.g. UGS Q(), rtPS Q(), and BE Q(), etc. Notice that different TCIDs will refer to a same classification if their service types are same, namely, a classification may contain several queues with unique TCIDs at the same time.

5.6.2.4 Scheduler

The Scheduler() function is in charge of selecting queued MSDUs according to the admitted bandwidth. The selection policy of the scheduler in the designed module uses the weighted Round-Robin method. To begin with, in the DL, we associate one percentage parameter with each classification as q5, q4, q3, q2, q1, which is corresponding to the UGS, rtPS, ertPS, nrtPS, and BE, respectively. In the first round, the expected serving quantity of each classification is calculated as BT type = min(Rtype,Btotal ∗ qi), i □ {1, . . . , 5} and _5 i=1 qi ≤ 1, where Rtype represents the total amount of requested type services and B total represents the total available bandwidth of the system. The parameters {q5, q4, q3, q2, q1} are variables and can be regulated by any simulation need.

In the second round, the Scheduler() will serve the remaining, unserved services in priority order. If all remaining services in priority i are served, the Scheduler() will serve the next priority i + 1 and so on. This process will be repeated until whole available bandwidths are exhausted or remaining required services are served. The adopted strategy is used to guarantee that lower priority traffic can still obtain a minimum bandwidth for transmission if the traffic load is extremely heavy. We emphasize that the scheduling algorithm or police is not mandatory in the standard specifications. In other words, this implies that researchers or engineers can design their own Scheduler() function according to their specific purposes or usages to substitute for this one. In addition, the ARQ function is an optional subject matter and is not implemented in our module.

5.6.2.5 DL-MAP/UL-MAP

The DL-MAP and UL-MAP are periodically generated to announce the information of the arrangement of the DL and UL periods in the superframe. These two messages are handled by DLmapHandler() and ULmapHandler(). There are management message types, PHY synchronization field, DCD count, base station ID, and DL-MAP IEs (IE: information element) in the DL-MAP message. The important part in the DL-MAP are the DL-MAP IE()s, which are composed by the type-length-value (TLV) encoding. Each DLMAP IE is generated by DL MAP IE() and each DL-MAP IE contains the downlink interval usage code (DIUC), the CID, the number of CIDs assigned for this IE, the

OFDMA symbol and subchannel offset, the number of the OFDMA symbols, and the number of the subchannels.

The structure of UL-MAP is similar to the DL-MAP, but the differences between them are the uplink channel ID and the allocation start time. The uplink channel ID is the identifier of the uplink channel to which this message refers. The allocation start time is the effective start time of the uplink allocation defined by the UL-MAP. The UL-MAP IE structure is also similar to the DL-MAP IE. The capability of the DL-MAP and the UL-MAP will decide the time domain and the frequency domain in the frame space.

5.6.2.6 Fragmentation/Packing

The packet fragmentation or packing process is executed by PDU Generator() function. This function grabs the MSDUs from QoS queues (UGS, rtPS, ertPS, nrtPS, and BE) and produces MPDUs depending on the command of the Scheduler(). It will generate the generic MAC header for each data payload. Fragmentation is the process that divides a MSDU into one or more MPDUs. If packing is turned on for a connection, the MAC may pack multiple MSDUs into a single MPDU. In this module, the input MSDU will be fragmented or packed depending on the length of the MPDU. Due to the reason of simplicity, we set the length of each MPDU fixed. Once the fragmentation or packing process is proceeded, the corresponding subheader will also be given to each MSDU contained in a MPDU.

After data fragmentation and packing, the scheduler will invoke Transmit Data() for data transmission. The treatment of the transmission will be various depending on whether it happened in the SS or BS. In the SS, the scheduler concatenates the MPDUs into one burst transmission according to the arrangement of the UL-MAP. In contrast to SS, the BS concatenates the MPDUs into one burst transmission according to the DL-MAP, which is generated by the scheduler. This function is periodically triggered in each superframe to decide to transmit burst data to/from DL or UL of the BS or SSs.

The Assembler() function is a process function of defragmenting and unpacking received data burst. It will read the MAC header to see whether this MPDU is fragmented or packed. Furthermore, the subheaders information such as FC in MPDU will also be read to recover an original MSDU.

5.6.2.7 802.16 timer class

The 802.16 Timer class inherits the Handler class with three important functions:

- The start() is used to trigger the timer to start.
- The stop() is used to stop the timer if the event happened before expiration.
- The handle() is used to trigger event while time runs out.

These Timer classes play an important role of the system such as sequencing the events between the BS and SSs in WiMAX networks. For instance, each superframe starts from the DL-MAPTimer and then it triggers the following timers, UL-MAPTimer, DCDPeriodTimer, and UCDPeriodTimer iteratively. Another kind of timer is used to count down the given specific time before the expected time expires or an event is encountered, e.g., the SS starts a RNG-REQ timer to wait for the RNG-RSP from the BS. All timer intervals defined in the WiMAX module are referred to the IEEE 802.16 standard.

5.7 Our Implementation

Base Station and Subscriber Stations are initialized along with other Mac Layer Parameters. Subscriber Stations performed initial ranging process and send ranging request to BS and wait for response. After successfully ranging process each SS have a unique ID. SS establish uplink connections with BS and each connection get a unique SFID (Service Flow ID). BS establishes downlink connection with BS and each connection get a unique SFID (Service Flow ID). After successful establishment, BS and SS's exchange data packet and management packets.

Subscriber Station Side Implementation

On receiving data packet from upper layer, Mac layer queue it according to respective connection queue. Before queuing the data packets there is function that calculate the WFQ packet finish time from current WFQ virtual time and WFQ finish time. A bandwidth allocated to SS is distributed among different connection and SS scheduler take the responsibility for distribution. Scheduling algorithm WFQ implement to distribute bandwidths. Earliest deadline first (EDF) and First in First out (FIFO) used for ordering packets according to meet their deadline. For UGS and Best Effort used FIFO and for rtPS and nrtPS used EDF principle. After successful scheduling packets, MAC header and sub header (if necessary) add to all packets and packets handed over to PHY layer. Fig 5.5 describes the graphically representation of SS implementation.

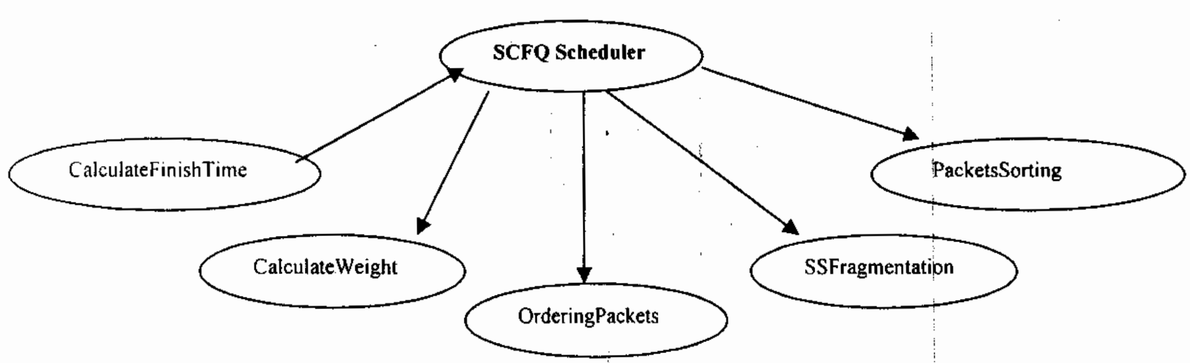

Figure 5.5: Subscriber Station Scheduler Functions

CalculateFinishTime: It's main function to calculate finish time of each packets from WFQ finish time and WFQ virtual time
CalculateWeight: It's main function to calculate weights for rtPS, nrtPS and BE to schedule packets according to weights from scheduling services. These weights are calculated from queue size and the priority associated with each scheduling service.
OrderingPackets: its main function to order packets according to ordering principle and ordering parameters.
SSFragmentation: It's main function to Fragments the packets according to specified size
PacketsSorting: It's main function to sort out packets according to descending order whenever there is need.

Base Station Side Implementation

Base station responsibility to produce uplink map, downlink map and schedule packets towards downlink direction. On receiving data packet from upper layer, Mac layer queue it according to respective connection queue. A bandwidth allocated to BS for downlink direction is distributed among different connection and BS scheduler take the responsibility for distribution. Scheduling algorithm WFQ implement to distribute bandwidths. Earliest deadline first (EDF) and First in First out (FIFO) used for ordering packets according to meet their deadline. For UGS and Best Effort used FIFO and for rtPS and nrtPS used EDF principle. After successful scheduling packets, MAC header and sub header (if necessary) add to all packets and packets handed over to PHY layer. Fig 5.6 and 5.7 describe the graphically representation of BS implementation.

Base station also fulfills the bandwidth requirement of different scheduling service and constant bandwidth requirement of UGS. These function also performed at BS side:

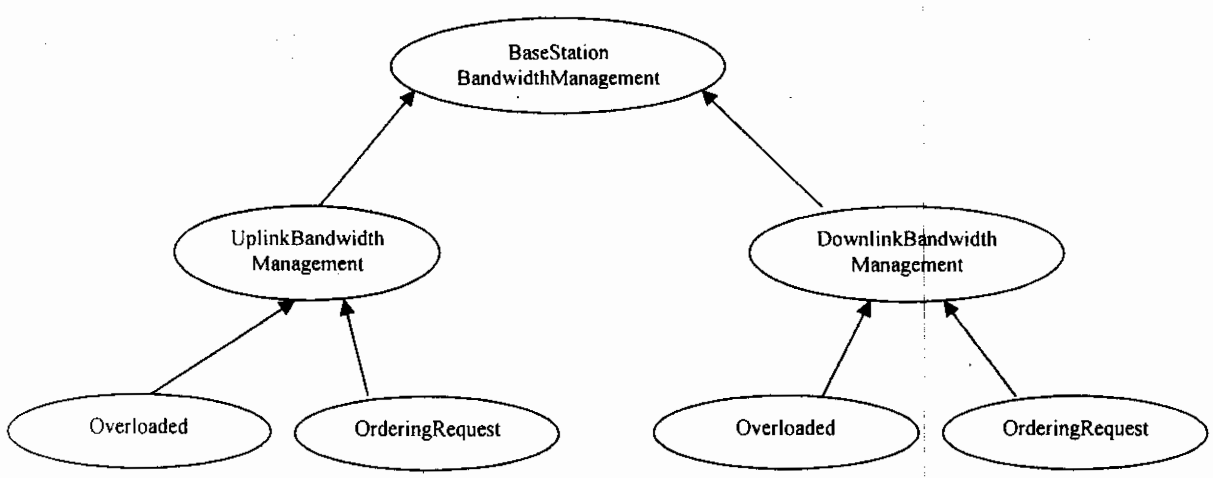

Figure 5.6: Base Station Bandwidth Management Functions

UplinkBandwidthManagement: It's main function to allocate bandwidth to different service flows and granted bandwidth to subscriber station.
DownlinBandwidthManagement: It's main function to allocate bandwidth to different service flows and granted bandwidth to Base Station Scheduler.
OrderingRequest: It's main function to order different request according to scheduling parameters and ordering principle to efficiently utilized bandwidth and meet QoS.
Overloaded: It's main function to keep fairness among different flows and Subscriber Station under overloaded condition. It also assures the bandwidth guarantee to UGS and rtPS flows. But it can not assure delay guarantee to UGS and rtPS flows as we increase the number of flows.

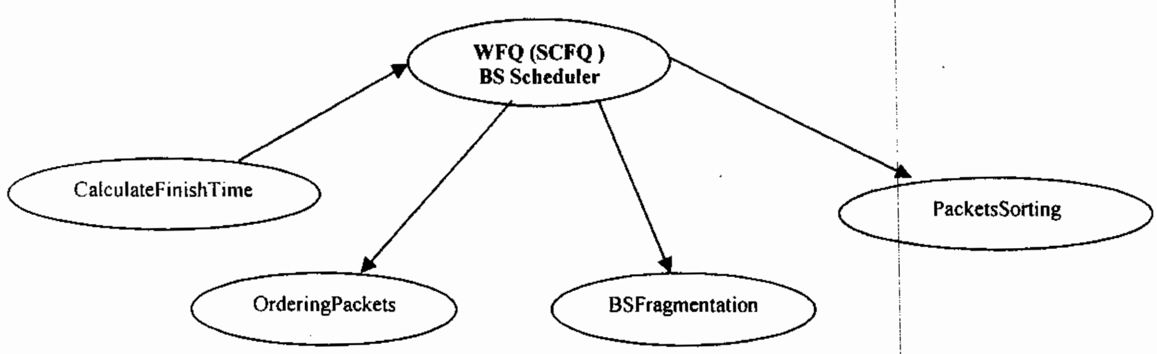

Figure 5.7: Base Station Scheduler Functions

CalculateFinishTime: It's main function to calculate finish time of each packets from WFQ finish time and WFQ virtual time
OrderingPackets: It's main function to order packets according to ordering principle and ordering parameters.
BSFragmentation: It's main function to Fragments the packets according to specified size
PacketsSorting: It's main function to sort out packets according to descending order whenever there is need.

Chapter 6

Results

6. Results

In this chapter we describe our simulation topology, simulation assumption and simulation results which show the effectiveness of our proposed scheme and achieve our stated goals that describe in previous chapter. We have done number of simulations to show the performance of our proposed scheme. We evaluate the following in our proposed scheme:

1) Effect of mean delay due to increase in number of uplink flow and downlink flow.
2) Effect of delay variation due to increase in number of uplink flow and downlink.
3) Effect of average throughput due to increase in number of uplink flow and downlink flow.
4) Effect of Fairness Index of flows due to increase in number of uplink flow and downlink flow and evaluates the effect of Fairness Index due to increase in number of Subscriber Station.
5) Evaluate the delay, jitter, and throughput of UGS, rtPS, nrtPS and BE flows.
6) Evaluate the packet delivery ratio and bandwidth utilization per Subscriber Station.

6.1 Assumptions

We have take number of assumption in our implementation of our proposed scheduling architecture in ns-2 simulator [26] and module developed by [17]. These are the following assumptions:

We are not implementing any admission control mechanism because our main focus to design such scheduling architecture that incorporate Quality of Service Parameters (QoS). We take an assumption a good admission control mechanism supported our scheduling architecture.

We are not implementing any security mechanism such like MAC Privacy sublayer because there is need a separate study for it.

We have used the following specifications to map different flows classes:

There are four kinds of service flows: UGS, rtPs, nrtPS, and BE which are all generated from the traffic generating agent in both the SS and the BS. The Internet traffic is treated as the DL traffic to the SSs; on the contrary, the UL traffic is the traffic from the SSs to the Internet.

UGS: Constant Bit Rate (CBR) traffic is used for UGS flow. Our CBR packet size is 220 Bytes with 64 Kbps constant rate.

rtPS: For real time services that generate variable size data packets is used for rtPS flow. Each connection of rtPS occupies 1 Mbps data rate and the data length follows the uniform distribution model by setting Uniform(200,980) (between 200 bytes–980 bytes) and time interval Uniform(-0.5,0.5).

nrtPS: Variable Bit Rate (VBR) traffic is used for nrtPS flow. Each connection occupies a mean data arrival rate 512 kbps and data length follows the uniform distribution model by setting Uniform (200, 1000) (between 200 bytes–1000 bytes) and time interval is 0.01.

BE: Data stream is used for BE flow. Each connection occupies a mean data arrival rate 512 kbps and data length follows the uniform distribution model by setting Uniform (200, 1000) (between 200 bytes–1000 bytes) and time interval is 0.01.

6.2 Performance Metrics

We have chosen Delay, Throughput, Jitter, Fairness Index and packet loss over IEEE 802.16 network as a performance metrics.

6.2.1 What is Delay?

The average time taken by the data packet to reach the intended destinations, here we considered Average End-to-End delay. This include delay occurred due to different reasons like queuing delay, propagation delay, processing delay etc. it is an important parameter for delay sensitive application like multimedia application. It is also very important for application where data is processed online.

$$Mean\ Delay = \frac{\sum_{i=1}^{n} Delay\ Of\ Packet}{n}$$

Delay of Packet: time from the packet is transmitted to the time the packet is received.

Delay of Packet = Propagation delay + Queuing delay + Transmission delay

Propagation delay = distance / signal propagation speed
Queuing delay = depend on the network load
Transmission delay = Size / Bandwidth

6.2.2 What is Jitter?

The variation in packet delay is sometimes called "jitter". This term, however, creates confusion because it is used in different ways by different groups of people. "Jitter" commonly has two meanings: The first meaning is the variation of a signal with respect to some clock signal, where the arrival time of the signal is expected to coincide with the arrival of the clock signal. This meaning is used with reference to synchronous signals and might be used to measure the quality of circuit emulation. The second meaning has to do with the variation of a metric (e.g., delay) with respect to some reference metric (e.g., average delay or minimum delay).This meaning is frequently used by computer scientists and frequently (but not always) refers to variation in delay.

Jitter is a variation (somewhat random) of the latency from packet to packet. Jitter is most often observed when packets traverse multiple hops from source to destination. Jitter is also considering the variation in the latency of packets at the destination. If the jitter value is high the performance in some time-sensitive applications, such as voice over IP, might get affected.

There are several ways of measuring jitter based on parameters being taken into account while measuring. Following method we are used to measuring jitter.

Jitter is calculated as the change in the difference in the arrival time of the packet.

$$\Delta \text{arrival}_n = |\text{Arrival}_n - \text{Arrival}_{n-1}|$$

where, n is the current packet.

$$\text{Jitter}_n = |\Delta\text{arrival}_n - \Delta\text{arrival}_{n-1}|$$

where, n is the current packet.

6.2.3 What is Throughput?

The amount of data transferred from one place to another or processed in a specified amount of time. Data transfer rates for disk drives and network are measured in terms of throughput. Typically, throughputs are measured in kbps, Mbps and Gbps. We usually think of throughput as ***measured performance***. Implementation inefficiencies may cause the achievable bit rate to be less than the bandwidth for which the networks was designed. Throughput is measured by following equation.

Throughput = Transfer Size / Transfer time

6.2.4 What is Packet Delivery Ratio?

Packet delivery ratio is equal to the number of packets received on destination divided by number of packet send on source during a specified time. Packet delivery ratio can be measured by following equation.

*Packet Delivery Ratio = (no. of received packets / no. of send packets) * 100*

6.2.5 What is Fairness Index?

To quantitatively measure the fairness of the bandwidth among flows, fairness index f is used [24].

$$f(x_1, x_2, \ldots, x_n) = (\sum_{i=1}^{n} x_i)^2 / n * \sum_{i=1}^{n} x_i^2$$

Here x_i is the throughput of the ith flow, e.g. the amount of data that has been successfully transferred from the sender to the target in each flow, n is the number of throughputs. The closer fairness index is to the value 1, the better (more equally) the bandwidth is utilized during the traffic flows.

6.3 Simulation Setup

The simulation topology as shown in fig 6.1 consists of one Base Station and number of Subscriber station. During simulation the number of SS is constant.

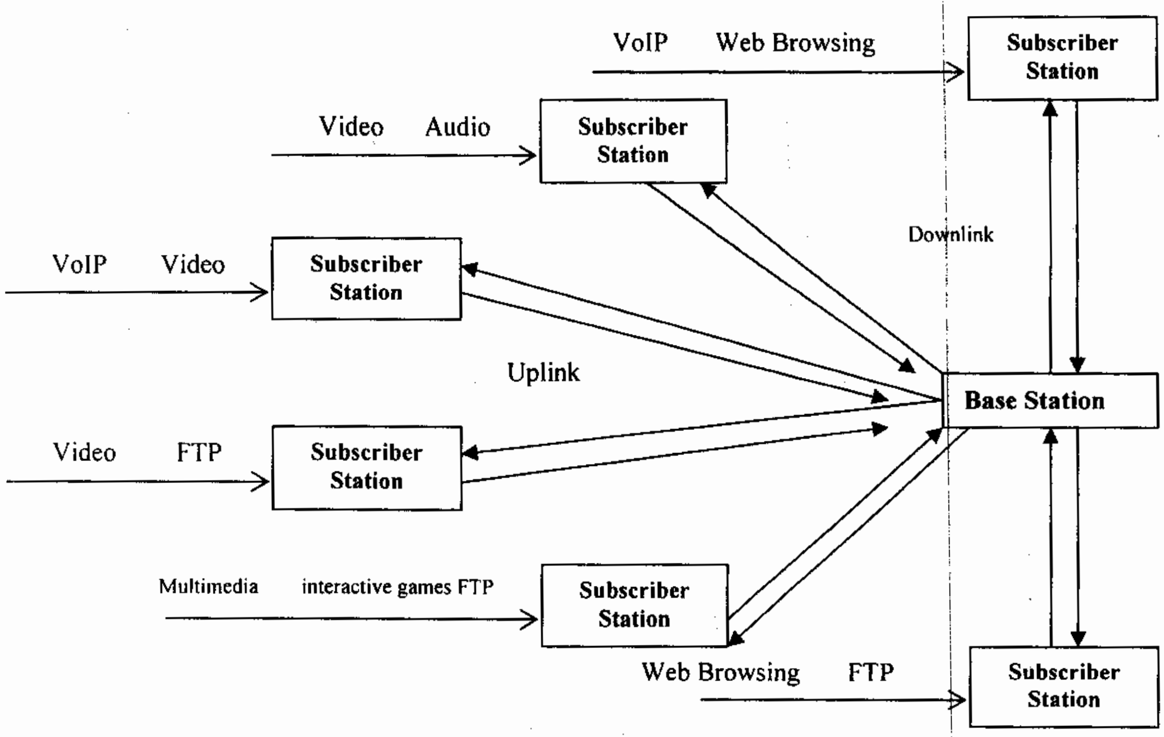

Figure 6.1: Simulation Setup

The simulation environment, shown in Fig 6.2, is set one serving BS to 10 SSs concurrently within a 1000m×1000m square.

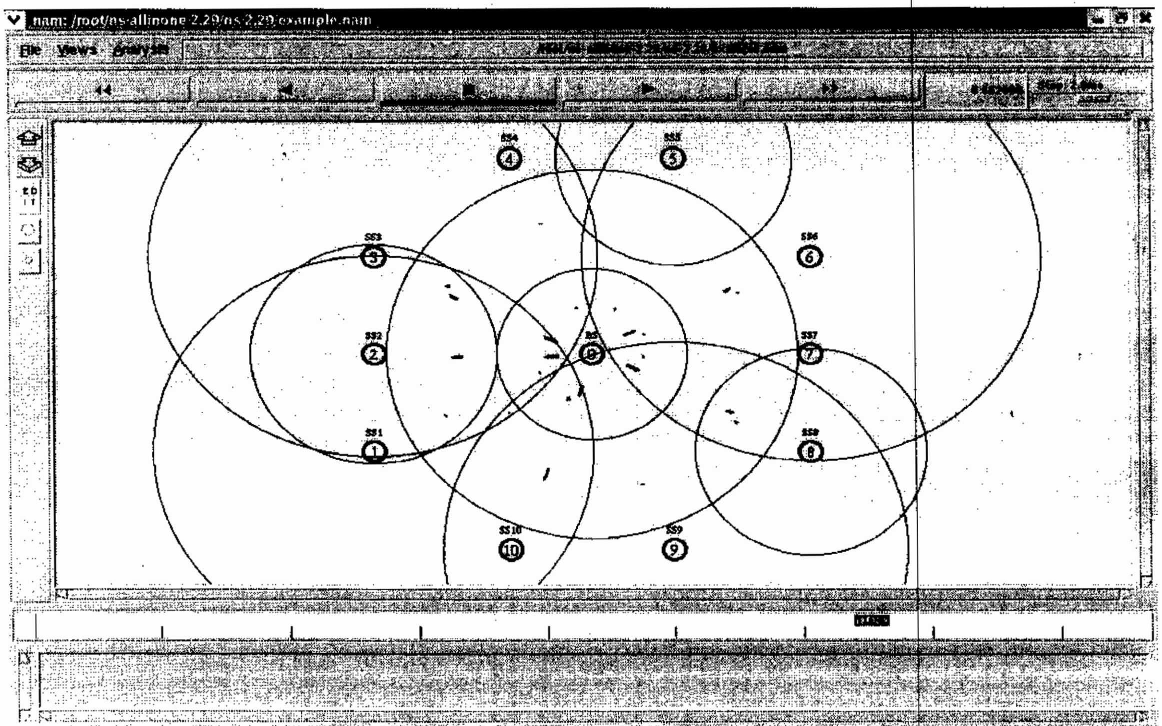

Fig 6.2: Simulation Environment in ns–2

6.4 Simulation Parameters

Table 6.1 describes simulation parameter for carry simulation in ns-2.

Table 6.1: Simulation Parameters

Parameters	Value
Spectrum	5.0 GHz
Bandwidth	20 MHz
Data Rate	Upto 74 Mbps
Downlink/Uplink ratio	3:2
OFDMA Symbol per Frame	49
OFDMA Symbol per Frame (data portion)	48
No. of subchannels	30
Ranging opp. per Frame	12 OFDMA symbols
Max. no. of ranging retry	10
Bandwidth request opp. per frame	12 OFDMA symbols
Max. no. of bandwidth req. retry	10
Basic CIDs	1–1000
Primary CIDs	1001–2000
Transport/secondary Mgt. CIDs	2001–65278
Broadcast CID	65535
SFID range	1–4294967295

6.5 Simulation Results

We have performed number of experiments to show the effectiveness and performance of our proposed architecture. We have run number of simulations with different subscriber station and each simulation, a SS has same number of uplink flows and downlink flows and with same QoS Parameter. Each simulation runs for 40 second.

6.5.1 Mean Mac Delay Analysis

We have measured Mean Mac delay of each subscriber station and Mean Mac delay of uplink and downlink flows across all subscriber station.

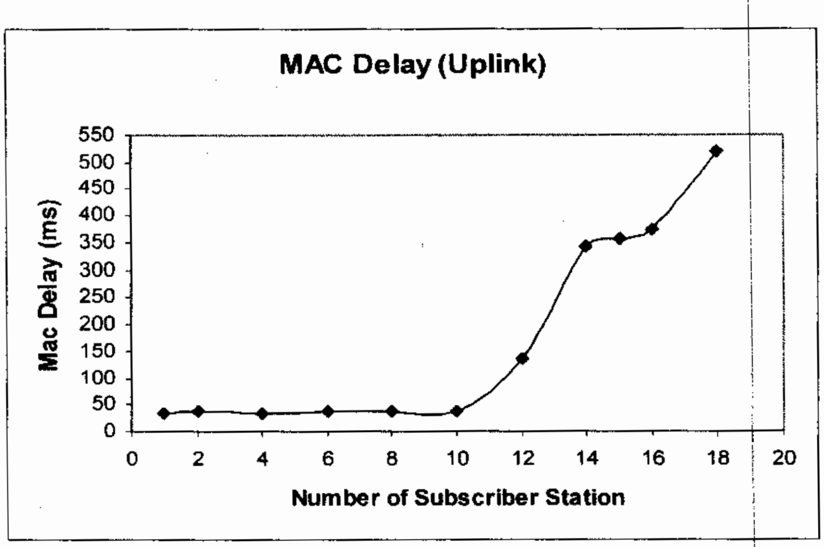

Fig 6.3: Mean Mac Delay Vs Number of Subscriber Station (Uplink)

Fig 6.3 shows the Mean Mac delay of each Subscriber Station in uplink direction. It is analyzed that mean delay increases as number of SS increases. Mean delay likely to be constant whenever there is enough bandwidth to schedule packets across all SS. As system is overloaded number of subscriber station then queue time increases so mean mac delay also increases. After 12 subscribers station there is rapid increase in mean delay.

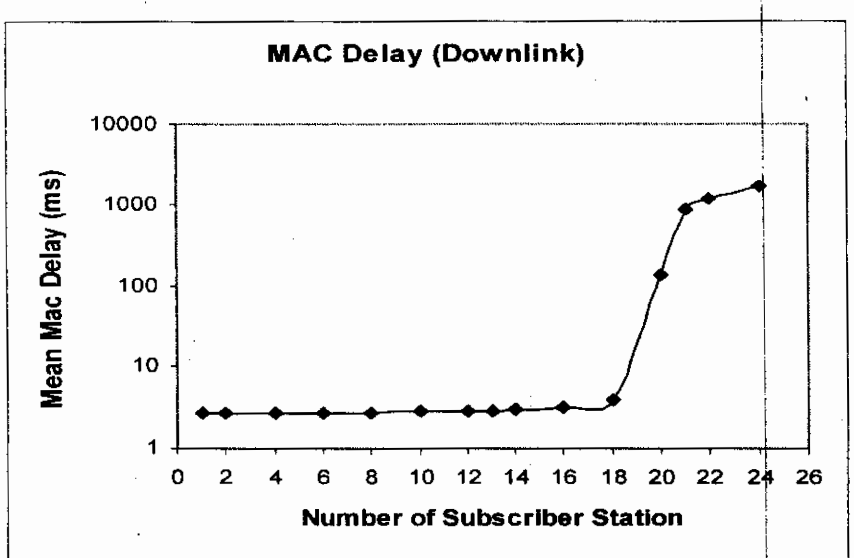

Fig 6.4: Mean Mac Delay Vs Number of Subscriber Station (Downlink)

Fig 6.4 shows the Mean Mac delay of each Subscriber Station in downlink direction. It is analyzed that mean delay increases as number of SS increases. Mean delay likely to be constant whenever there is enough bandwidth to schedule packets across all SS. As system is overloaded number of subscriber station then queue time increases so mean mac delay also increases. After 18 subscribers station there is rapid increase in mean delay.

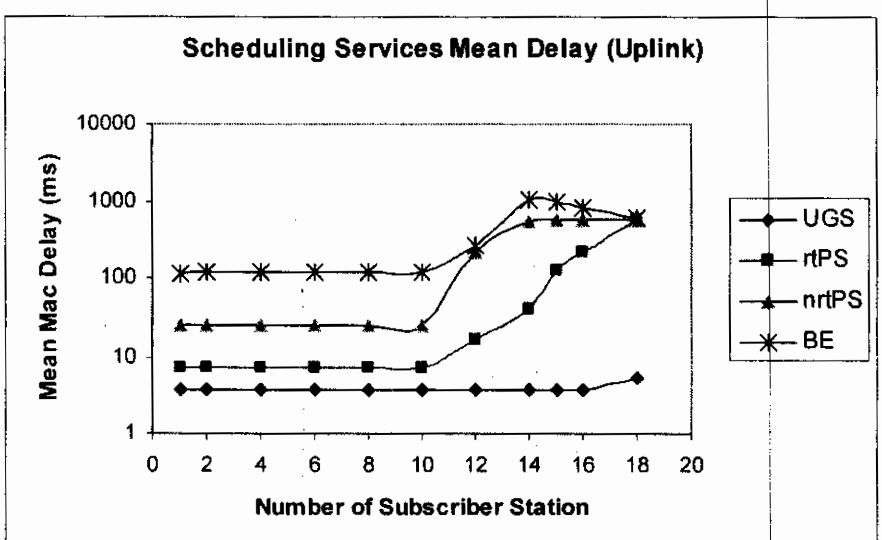

Fig 6.5: Mean Mac Delay of Scheduling Services (Uplink)

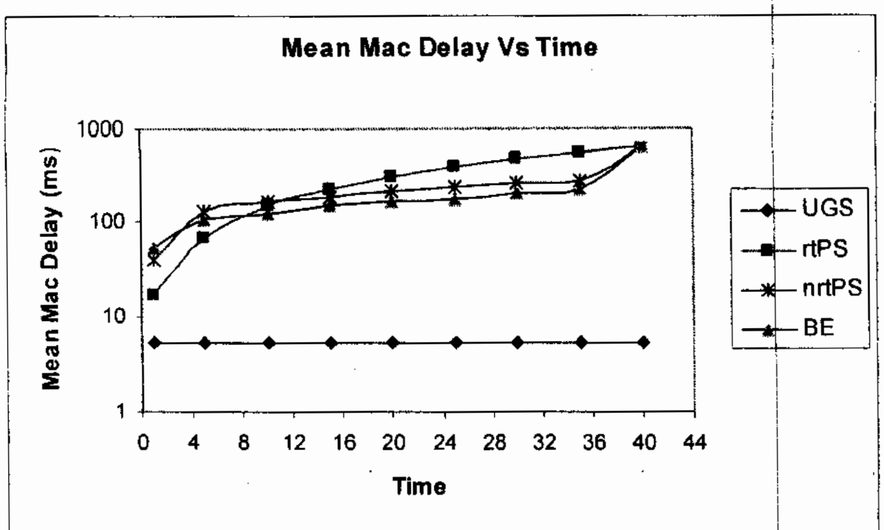

Fig 6.6: Mean Mac Delay Vs Time (Uplink)

Fig 6.5 shows the Mean Mac delay of Scheduling service flows of Subscriber Station in uplink direction. It is analyzed that mean delay of UGS flows merely same across all SS as the number of SS increasing due to constant bandwidth allocation. Mean delay of rtPS flows increase gradually after overloaded condition because it is delay tolerant traffic. Mean delay of nrtPS and BE flows increasing sharply with respect to subscriber station. At 18 subscriber station, mean delay of nrtPS and BE same due to Packet ordering by size and packet delivery ratio is also decreasing. Fig 6.6 shows the mean mac delay of scheduling services with respect to time. It shows that mean delay of UGS flows remain constant through simulation time. Mean mac delay of rtPS connection increases gradually with respect to time. nrtPS and BE flows show variation in mean mac delay through simulation time.

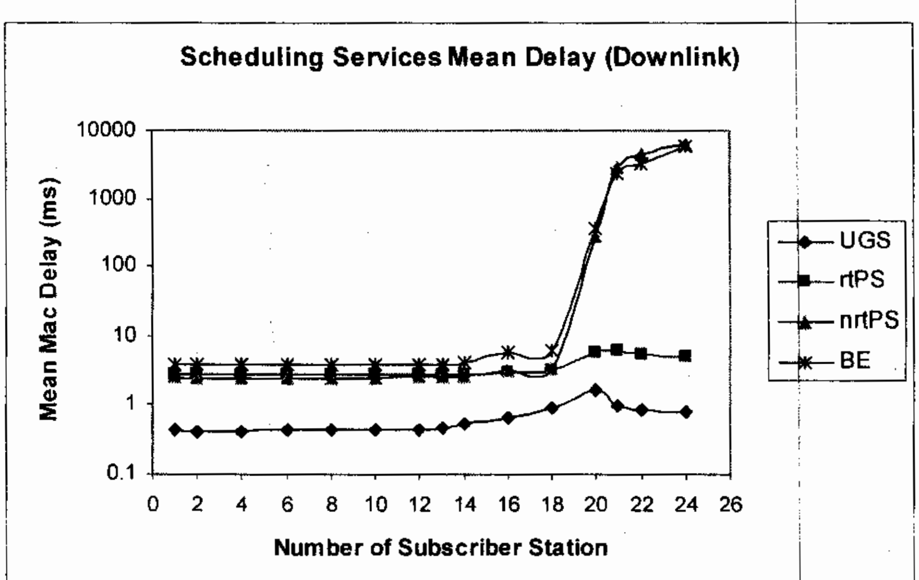

Fig 6.7: Mean Mac Delay of Scheduling Services (Downlink)

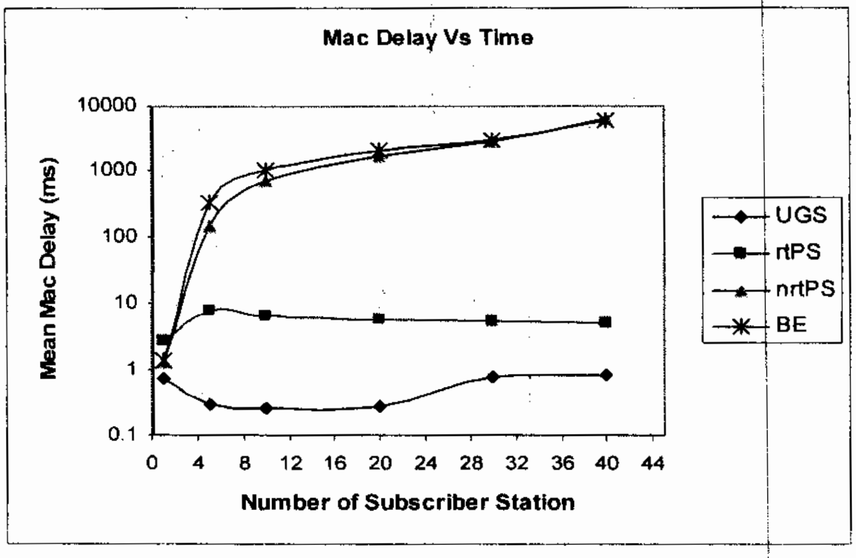

Fig 6.8: Mean Mac Delay Vs Time (Downlink)

Fig 6.7 shows the Mean Mac delay of Scheduling service flows of Subscriber Station in downlink direction. It is analyzed that mean delay of UGS flows merely same across all SS as the number of SS increasing due to constant bandwidth allocation. Mean delay of rtPS flows increase very slowly after overloaded condition because it is delay tolerant traffic. Mean delay of nrtPS and BE flows increasing sharply with increase in subscriber stations. Fig 6.8 shows the mean mac delay of scheduling services with respect to time. It shows that mean delay of UGS flows remain constant through simulation time. Mean mac delay of rtPS connection increases gradually with respect to time. nrtPS and BE flows mean mac delay increasing through simulation time.

6.5.2 Jitter Analysis

We have measured jitter of uplink and downlink flows across all subscriber station.

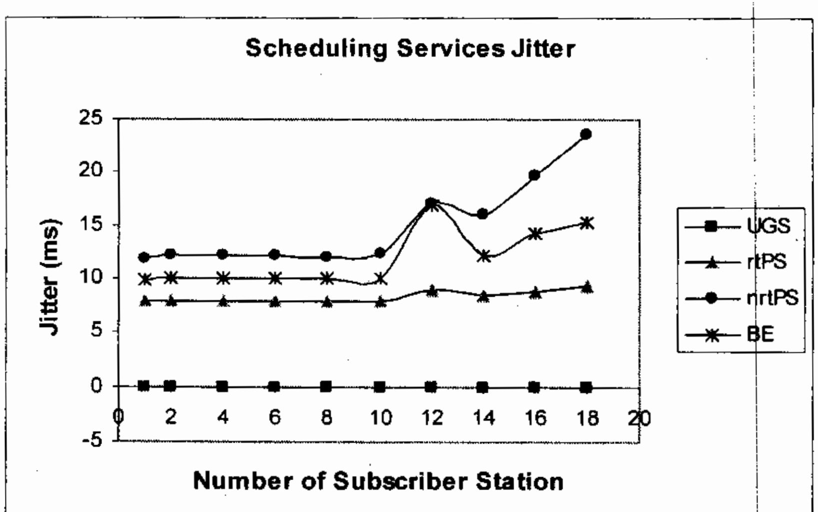

Fig 6.9: Jitter of Scheduling Services (Uplink)

Fig 6.9 shows the jitter of Scheduling service flows of Subscriber Station in uplink direction. It is analyzed that jitter of UGS flows same across all SS. UGS jitter is app. 0 ms. Jitter of rtPS flow is increasing slowly from 7 to 9 ms. Jitter of nrtPS and BE flows increasing from 10 to 16 ms when the number of SS increasing from 12 to 18.

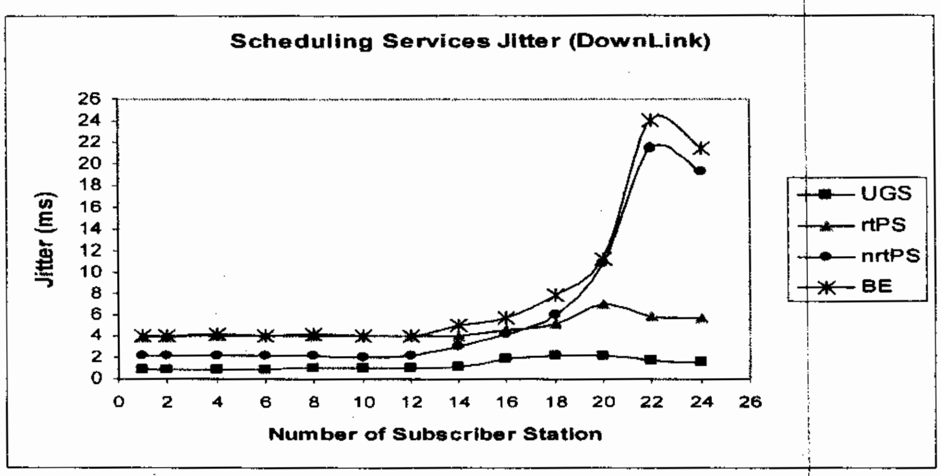

Fig 6.10: Jitter of Scheduling Services (Downlink)

Fig 6.10 shows the jitter of Scheduling service flows of Subscriber Station in uplink direction. It is analyzed that jitter of UGS flows is little fluctuate across all SS. UGS jitter is vary from 0.5 ms to 2 ms. Jitter of rtPS also vary little from 5 to 6 ms after 18 nodes. Jitter of nrtPS flows increasing from 10 to 20 ms and jitter BE flows increasing from 10 to 21 ms, when the number of SS increasing from 18 to 24. Jitter of nrtPS and BE flows are decreasing after 22 nodes due to packets sorting by size.

6.5.3 Packets Delivery Ratio

We have measured packet delivery ratio of uplink and downlink flows through all subscriber station.

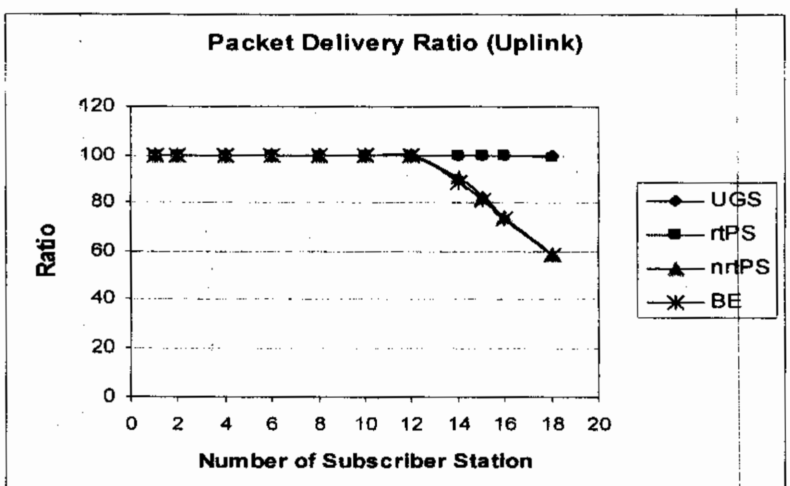

Fig 6.11: Packet Delivery Ratio (Uplink)

Fig 6.11 shows the packet delivery ratio of service flows of Subscriber Station in uplink direction. It shows that UGS and rtPS ratio is constant which 99.99 percentages through number of SS is increasing. Both traffics are delay tolerant so it is necessary to schedule packets as soon as possible to meet the delay guarantee. It is also analyzed that ratio for nrtPS and BE decreasing after 12 node.

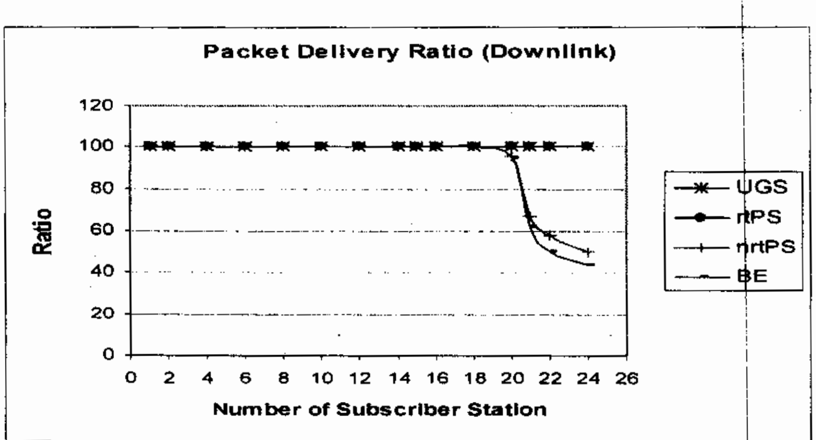

Fig 6.12: Packet Delivery Ratio (Downlink)

Fig 6.12 shows the packet delivery ratio of service flows of Subscriber Station in downlink direction. It shows that UGS and rtPS ratio is constant which 99.99 percentages through number of SS is increasing. These traffics are delay tolerant so it is necessary to schedule packets as soon as possible to meet the delay guarantee. It is also analyzed that ratio for nrtPS and BE decreasing after 20 node.

6.5.4 Throughput Analysis

We have measured uplink, downlink, scheduling flows and subscriber station throughput with respect to time.

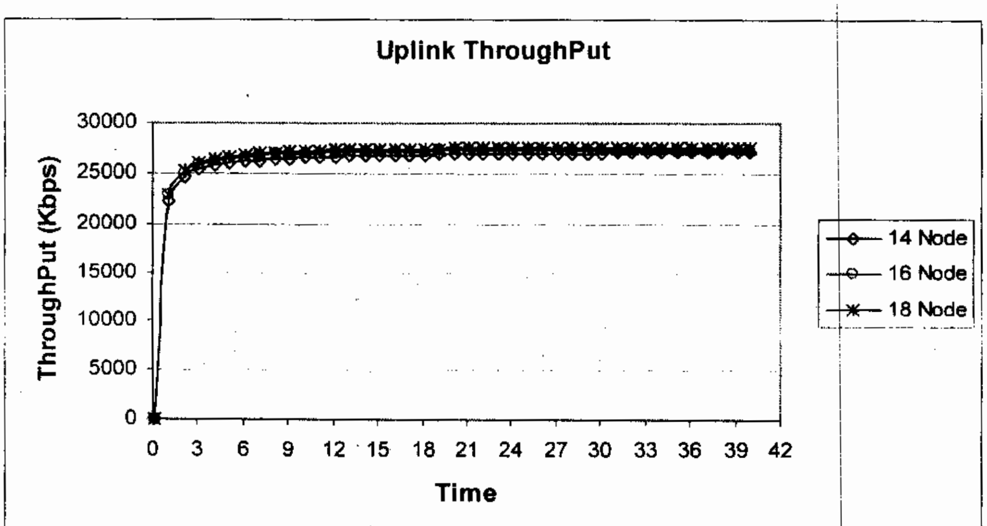

Fig 6.13: Uplink Throughput Vs Time (Uplink)

Fig 6.13 shows the uplink throughput with 14,16,18 SS through simulation time. There is slight difference between them with 14 SS maximum throughput is 27065 kbps, with 16 SS maximum throughput is 27331 kbps and for 18 nodes maximum throughput 27534 kbps.

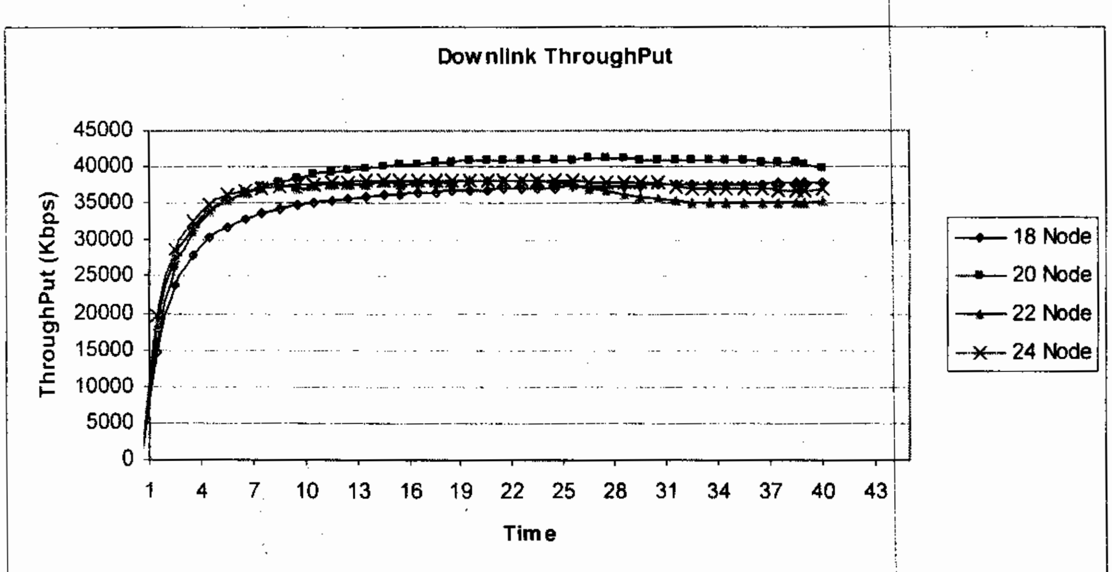

Fig 6.14: Downlink Throughput Vs Time (Downlink)

Fig 6.14 shows the downlink throughput with 18,20,22,24 SS through simulation time. Maximum throughput before system overloaded is 37730 kbps, and throughput decreasing with respect to number of SS and remain constant at 38000 kbps.

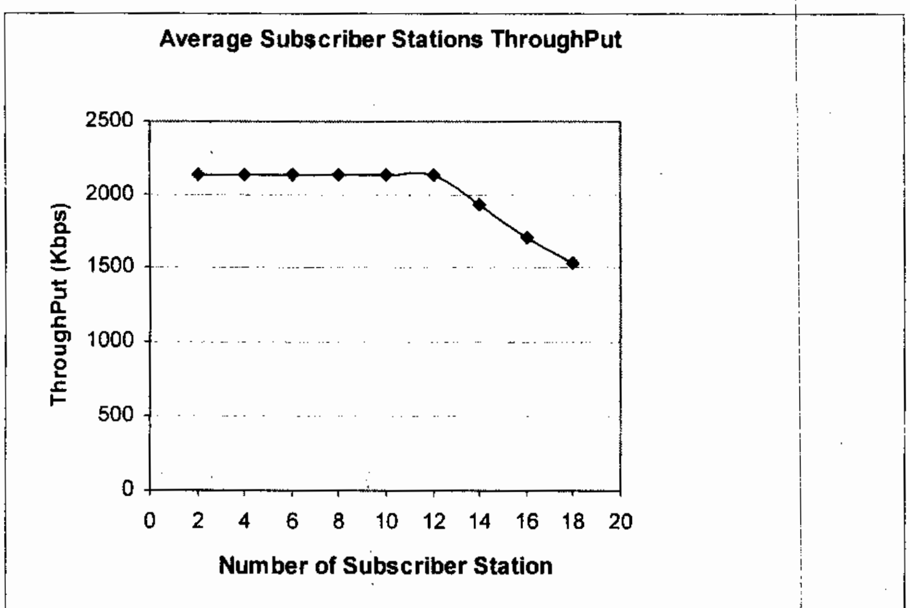

Fig 6.15: Subscriber Station Throughput

Fig 6.15 shows the Average subscriber station throughput with respect to number of subscriber station. Average subscriber station throughput gradually decreases as the number of SS increasing.

6.5.5 Fairness Index

We have measured fairness index as the number of subscriber station increasing.

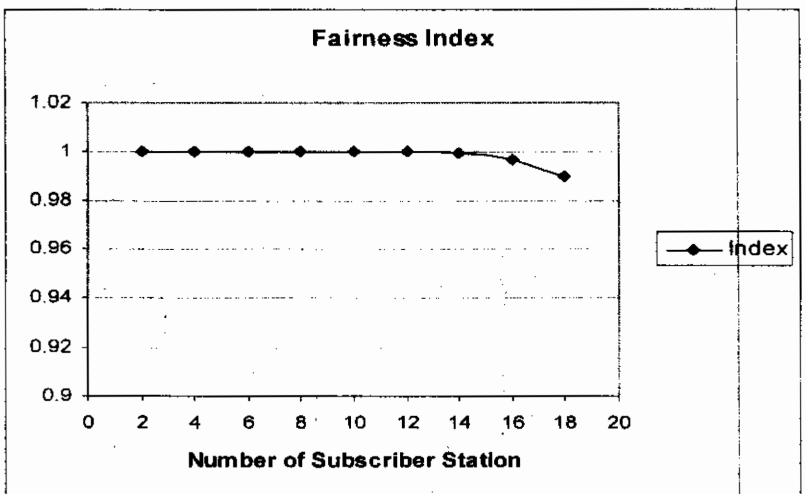

Fig 6.16: Fairness Index

Fig 6.16 shows the Fairness index across all SS. There is slightly decreasing from 1 to 0.99. It shows that bandwidth is distributed among SS equally and fairly according to scheduling flows that show the fairness and effectiveness of scheduling mechanism

6.5.6 Bandwidth Utilization

We have measured bandwidth utilization as the number of subscriber station increasing.

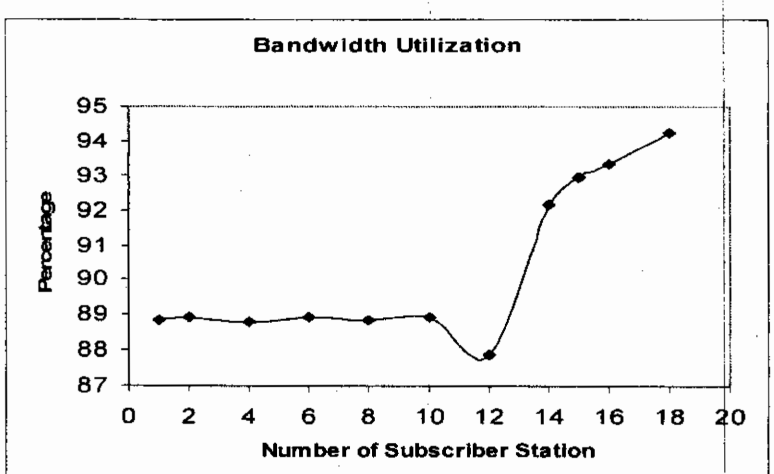

Fig 6.17: Bandwidth Utilization

Fig 6.17 shows the bandwidth utilization across all SS. Bandwidth utilization is 89 percentages. There is slight decrease of 1 percentage and then it increases as increases SS. Maximum bandwidth utilization at 18 subscriber stations is over 94.5 percentages.

6.5.7 Mean Mac Delay Analysis under Fragmentation

We have run number of simulations with different subscriber station under fragmentation is enabled at rtPS, nrtPS and BE flows and each simulation, a SS has same number of uplink flows and downlink flows and with same QoS Parameter. Each simulation runs for 40 second. We have measured Mean Mac delay of each subscriber station and Mean Mac delay of uplink and downlink flows under fragmentation across all subscriber station.

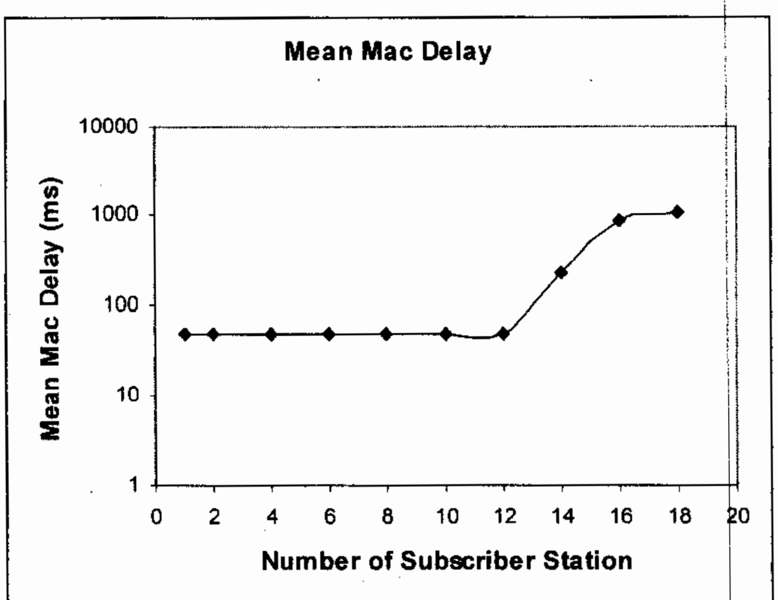

Fig 6.18: Mean Mac Delay under Fragmentation

Fig 6.18 shows the Mean Mac delay under fragmentation of each Subscriber Station in uplink direction. It is analyzed that mean delay increases as number of SS increases. Mean delay likely to be constant whenever there is enough bandwidth to schedule packets across all SS. As system is overloaded number of subscriber station then queue time increases so mean mac delay also increases. After 12 subscribers station there is gradually increase in mean delay.

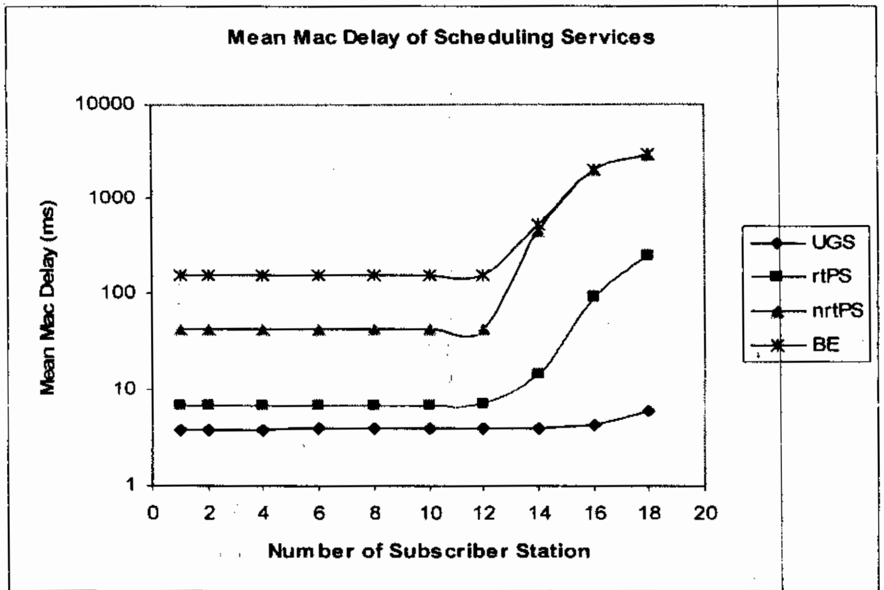

Fig 6.19: Mean Mac Delay of Scheduling Flows under Fragmentation

Fig 6.19 shows the Mean Mac delay under fragmentation of Scheduling service flows of Subscriber Station in uplink direction. It is analyzed that mean delay of UGS flows merely same across all SS as the number of SS increasing due to constant bandwidth allocation. Mean delay of rtPS flows increase gradually after overloaded condition because it is delay tolerant traffic. Mean delay of nrtPS and BE flows increasing sharply with respect to subscriber station. It also show that nrtPS and BE flows delay increase in upward direction there is not decline in nrtPS and BE curve because in fragmentation case packets sorting is disable.

6.5.8 Jitter Under Fragmentation

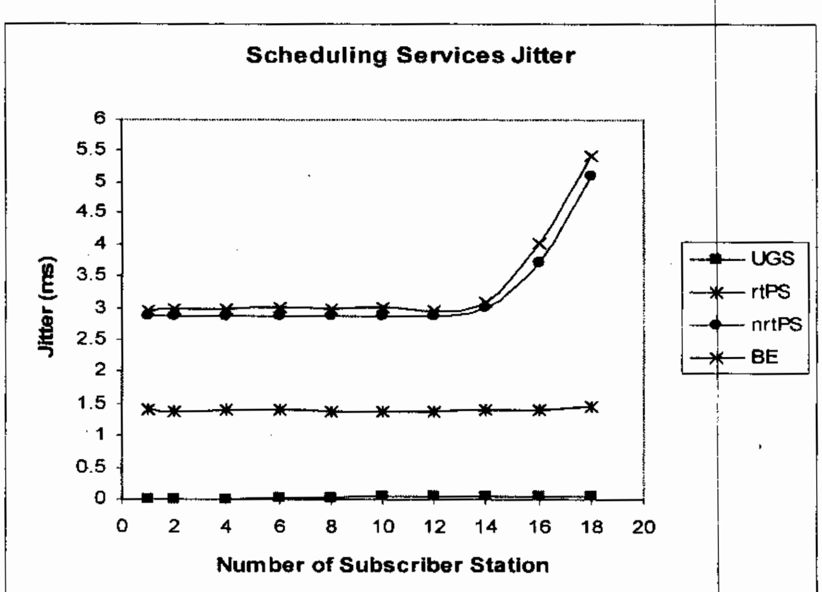

Fig 6.20: Jitter under Fragmentation

Fig 6.20 shows the jitter of Scheduling service flows under fragmentation of Subscriber Station in uplink direction. It is analyzed that jitter of UGS flows approximate same across all SS. Jitter of rtPS flow remains constant through simulation time that is app. 1.4 ms. Jitter of nrtPS and BE flows increasing from 3 to 5 ms when the number of SS increasing from 14 to 18.

6.5.9 Bandwidth Utilization Under Fragmentation

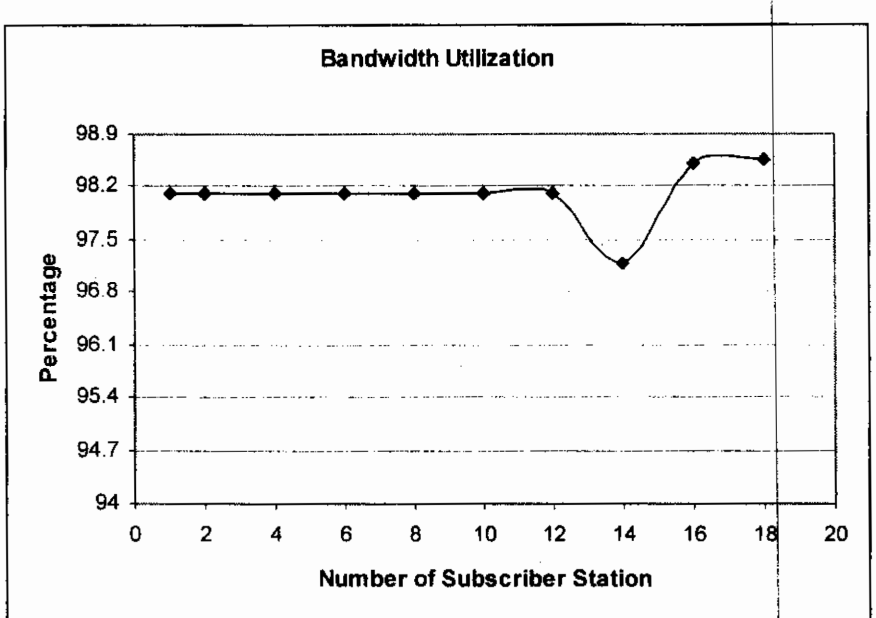

Fig 6.21: Bandwidth Utilization under Fragmentation

Fig 6.21 shows the bandwidth utilization under fragmentation across all SS. Bandwidth utilization is 98.1 percentages. There is slight decrease of 1 percentage and then it increases as increases SS. Maximum bandwidth utilization at 18 subscriber stations is over 98.55 percentages. This graph shows the importance of fragmentation when fragmentation is disable then bandwidth utilization is 89 percentages but in case fragmentation that is 98.1 that is 9 percentages greater. So fragmentation plays an important role in utilization of bandwidth and it also save the bandwidth from wastage.

6.6 Simulation Result for Variant Traffic

For variant traffic simulation, we have generated two types of UGS, two types of rtPS, one type of nrtPS and two type of BE service flow.

UGS_1: Constant Bit Rate (CBR) traffic is used for UGS flow. It represents the one of VoIP codec ITU G.722 and we model it for our simulation. Therefore our CBR packet size is 220 Bytes with 64 Kbps constant rate. Parameters associated with this flow are following:

Maximum Sustained Rate = 64 kbps
Tolerated Jitter = 1 ms
Maximum Latency =10 ms
UGS SDU Size = 200 bytes

UGS_2: Constant Bit Rate (CBR) traffic is used for second UGS flow. It also represents the one of VoIP codec ITU G.726 and we model it for our simulation. So therefore CBR packet size is 80 Bytes with 32 Kbps constant rate. Following parameters are associated with this flow:

Maximum Sustained Rate = 32 kbps
Tolerated Jitter = 1 ms
Maximum Latency =8 ms
UGS SDU Size = 80 bytes

rtPS_1: For real time services that generate variable size data packets is used for rtPS flow. It actually represents the MPEG-2 codec and follows such data rate that specifies for its best quality. Each connection of rtPS_1 occupies 1 Mbps data rate and the data length follows the uniform distribution model by setting Uniform(200,980) (between 200 bytes–980 bytes) and time interval Uniform(-0.5,0.5). Following parameters are associated with this rtPS_1 flow:

Maximum Sustained Rate = 1 Mbps
Minimum Reserved Rate = 512 kbps
Maximum Latency = 50 ms

rtPS_2: For real time services that generate variable size data packets is used for second rtPS flow. It also follows the video data rate that occupies for a video traffic. Every connection of rtPS_2 occupies 384 Kbps data rate and the data length follows the uniform distribution model by setting Uniform(200,980) (between 200 bytes–980 bytes) and time interval Uniform(-0.5,0.5). Parameters associated with this rtPS flows are following:

Maximum Sustained Rate = 424 Kbps
Minimum Reserved Rate = 384 Kbps
Maximum Latency = 30 ms

nrtPS: Variable Bit Rate (VBR) traffic is used for nrtPS flow. It actually represent the File transfer protocol application. Each connection occupies a mean data arrival rate 512 kbps and data length follows the uniform distribution model by setting Uniform (200, 1000) (between 200 bytes–1000 bytes) and time interval is 0.01. Service flow parameter associated with this flow:

Maximum Sustained Rate = 1 Mbps

Minimum Reserved Rate = 512 kbps
Traffic Priority = 1

BE_1: Data stream is used for BE_1 flow. All connection occupies a mean data arrival rate 124 kbps and data length follows the uniform distribution model by setting Uniform (200, 1000) (between 200 bytes–1000 bytes) and time interval is 0.01.

Maximum Sustained Rate = 124 Kbps
Traffic Priority = 3

BE_2: Data stream is used for BE_2 flow. Each connection occupies a mean data arrival rate 32 kbps and data length follows the uniform distribution model by setting Uniform (200, 1000) (between 200 bytes–1000 bytes) and time interval is 0.01. BE_2 flow has greater priority than BE_1. Parameters are associated with this flow.

Maximum Sustained Rate = 32 Kbps
Traffic Priority = 1

6.6.1 Delay under Variant Traffic

We have measured Mean Mac delay of each subscriber station and Mean Mac delay of uplink flows across all subscriber station.

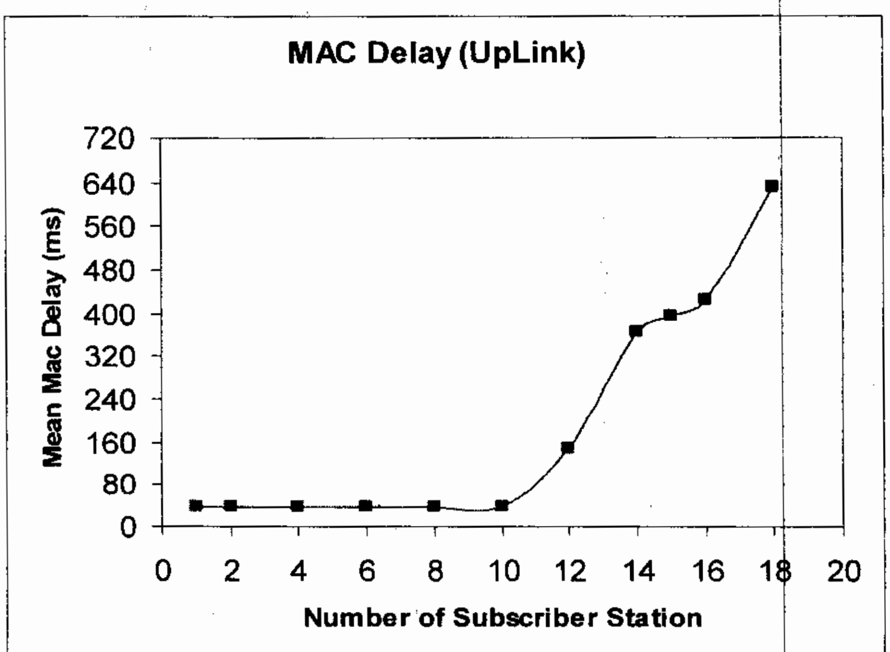

Fig 6.22: Mean Mac Delay under Variant Traffic

Fig 6.22 shows the Mean Mac delay of each Subscriber Station in uplink direction under variant traffic. It is analyzed that mean delay increases as number of SS increases. Mean delay likely to be constant whenever there is enough bandwidth to schedule packets across all SS. As system is overloaded due to number of subscriber station increase then queue time (waiting time) increases so mean mac delay also increases. After 12 subscribers station there is gradually increase in mean delay.

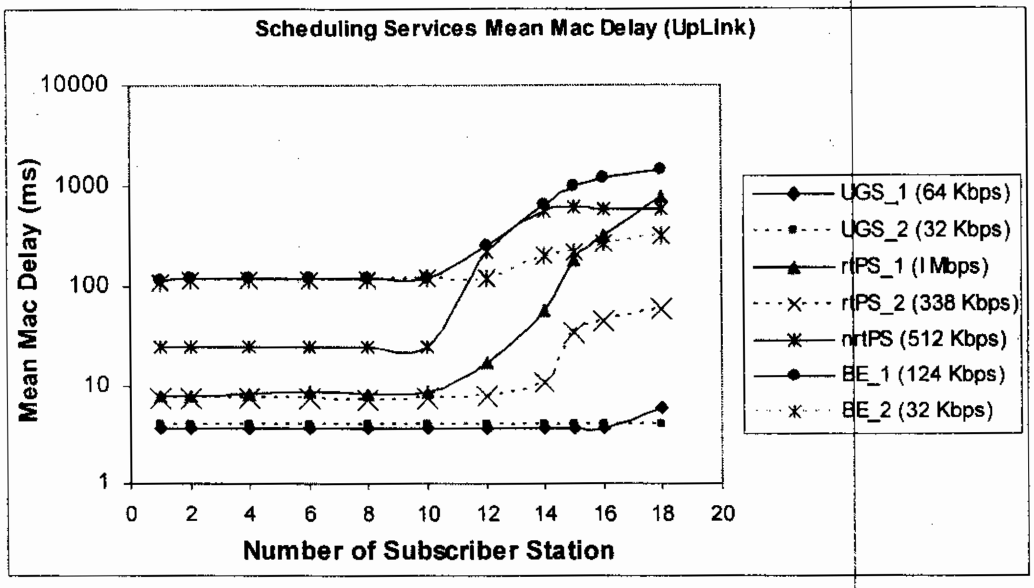

Fig 6.23: Mean Mac delay of Scheduling Service under Variant Traffic (Uplink)

Fig 6.23 shows the Mean Mac delay of Scheduling service flows of Subscriber Station in uplink direction. It is analyzed that mean delay of UGS_1 and UGS_2 flows merely same across all SS as the number of SS increasing due to constant bandwidth allocation. UGS_2 delay is same across all SS due to its less jitter as compared to UGS_1. Mean delay of rtPS_1 flow increase gradually after overloaded condition and delay of rtPS_2 slightly increase. Both rtPS traffic are delay tolerant, so whenever scheduling take place between these two flows, rtPS_2 packets have got bandwidth first so rtPS_2 delay is lesser than rtPS_1. The main reason rtPS_2 got bandwidth early because its *Maximum Latency* and *Minimum Reserved Traffic Rate* is less than rtPS_1. Mean delay of nrtPS flow increasing sharply with respect to number of subscriber station. After 14 SS it touches its peak point and remains stable for 3 to 4 SS and then decreases. This behavior of nrtPS due to packets reordering because whenever nrtPS got bandwidth smaller packets is scheduled first. Mean delay of BE_1 and BE_2 increases gradually after 12 subscriber stations and delay of BE_2 is not increasing as much as compared to BE_1 because BE_2 have higher *Priority* than BE_1.

6.6.2 Jitter under Variant Traffic

We have measured jitter of uplink flows across all subscriber station. It measured the delay variation for different services flow with respect to number of subscriber station.

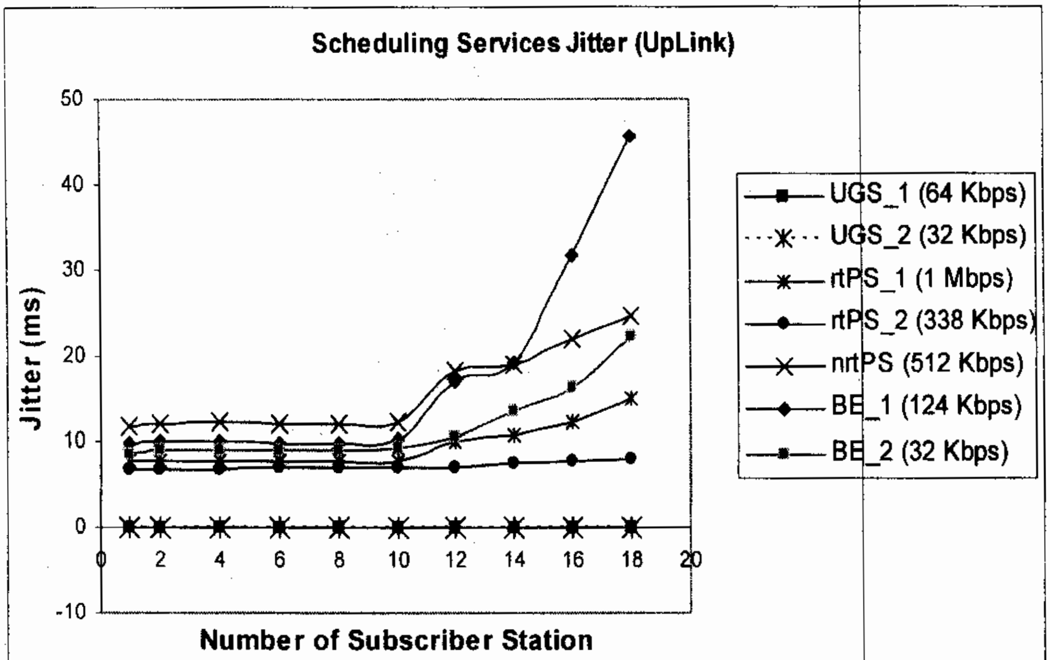

Fig 6.24: Jitter of Scheduling Service under Variant Traffic (Uplink)

Fig 6.24 shows the jitter of Scheduling service flows of Subscriber Station in uplink direction under variant traffic. It is analyzed that jitter of both UGS flows same across all SS that is near to 0.2 ms. Jitter of rtPS_1 flow is increasing slowly from 10 to 14 ms after 12 SS but as compared to rtPS_2 jitter remain same and there is slight increase from 7 to 8 after12 SS. Both UGS and rtPS flows are delay tolerant traffic and they cannot tolerate more jitter, so subscriber station scheduled their packets as soon as possible to achieve lesser delay and jitter. Jitter of nrtPS flow increasing gradually as the number of SS increasing from 12 to 18. Jitter of BE_1 flow increasing sharply as the number of SS increasing from 12 to 18 and jitter of BE_2 increases slowly.

6.6.3 Bandwidth Utilization under variant traffic

We have measured bandwidth utilization as the number of subscriber station increasing. Fig 6.25 shows the bandwidth utilization across all SS. Bandwidth utilization is above 90 percentages. As the number of SS increasing bandwidth utilization is also increasing. Maximum bandwidth utilization at 18 subscriber stations is over approximately 93.5 percentages.

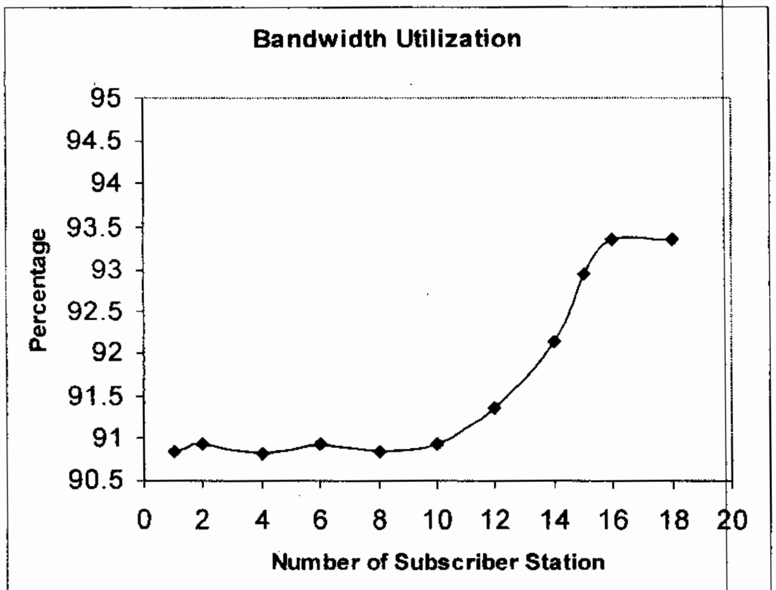

Fig 6.25: Bandwidth Utilization under Variant Traffic

6.7 Comparison with other scheduling architecture

We have compared our simulation result with one of the previously presented scheduling mechanism that is named "*Quality of Service Support in IEEE 802.16 Networks*" by Claudio Cicconetti, Luciano Lenzini, Enzo Mingozzi and Eklund and was published in IEEE Network, March/April 2006. Claudio Cicconetti et. al. in 2006, assess the performance of IEEE 802.16 in two of the most promising scenario that are residential and small and medium-sized enterprise (SME) envisaged by the Wimax forum [10]. In this paper, writers assume Weighted Round Robin (WRR) as an uplink scheduler and Deficit Round Robin (DRR) as a Downlink Scheduler. Figure 6.27 and 6.29 show the mean mac delay and jitter of different flows of above mentioned scheduling mechanism with respect to number of subscriber station. Figure 6.26 and 6.28 show the mean mac delay and jitter of different services flows of our architecture.

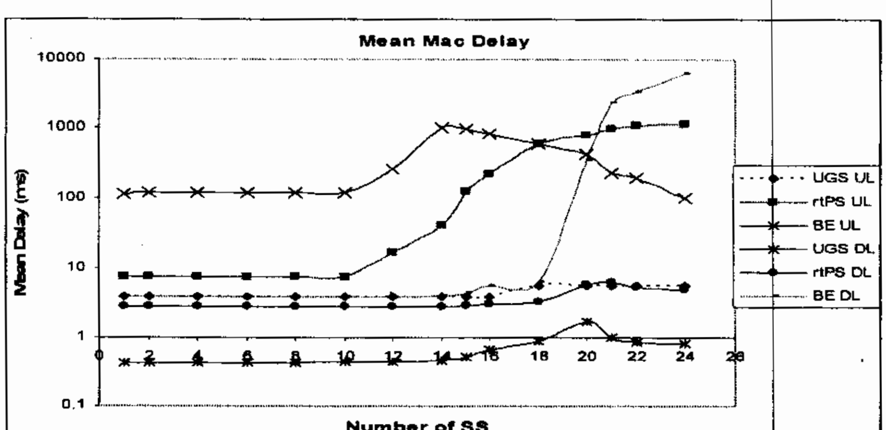

Fig 6.26: Mean Mac Delay of Scheduling Services (Uplink & Downlink)

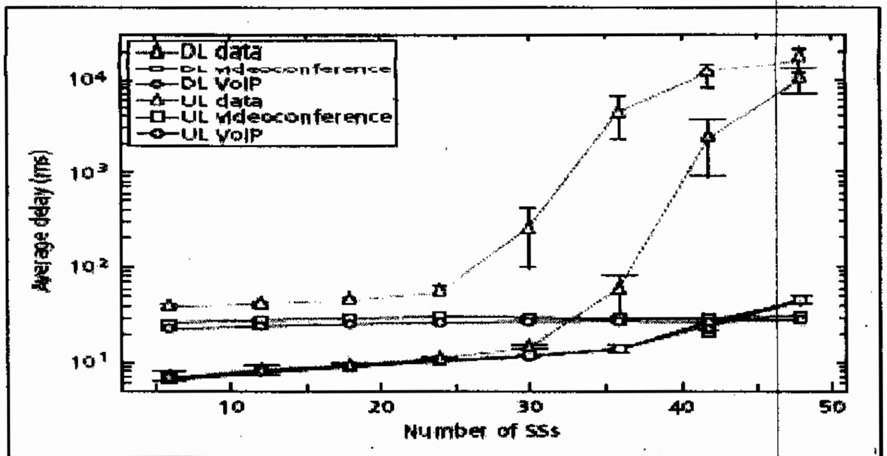

Fig 6.27: Average Delay Vs. Number of SS [10]

As fig 6.26 shows that mean delay of UGS flow remain constant and mean delay is 4 ms and it is compare with UL VoIP (represent UGS) of fig 6.27 that is also remain constant as the number of subscriber station increases and its mean delay is approximately 20 ms that is also acceptable in case of VoIP. Mean delay of UGS in downlink direction is remain constant but fig 6.27 shows that DL VoIP (represent UGS) increases slightly as number of SS increasing. Our UGS result shows better in both uplink and downlink direction as we compare with fig 6.27. It does not mean second one show bad result. Our result is better on basis of quality of service and performance.

As fig 6.26 shows that mean delay of rtPS toward uplink direction is increasing slightly after overloaded condition and it is compare with UL Videoconference (represent rtPS) of fig 6.27 that is remain constant as the number of subscriber station increases and its mean delay is approximately 20 ms. Mean delay of rtPS in downlink direction is remain constant but fig 6.27 shows that DL Videoconference (represent rtPS) increases slightly as number of SS increasing. It show that second one show better result in case of uplink direction and our result better in case of downlink. We can say that our scheduling architecture provides better support to QoS and performance and more scalable.

As fig 6.26 shows that mean delay of BE toward uplink and downlink direction is increasing sharply after overloading and it is compare with UL Data (represent BE) and DL Data (represent BE) of fig 6.27 which also increase sharply. It show that our simulated result better in case of uplink and downlink direction. We can say that our scheduling architecture provides bandwidth guarantees to BE flows to meets its QoS and that's bandwidth guarantees achieved by SCFQ scheduling algorithm under the presence of delay tolerant traffic and we can say that our scheduling architecture firstly fulfils the requirement of delay tolerant traffic and try its best to fulfill the bandwidth guarantees to bandwidth consumed traffic such as nrtPS and BE.

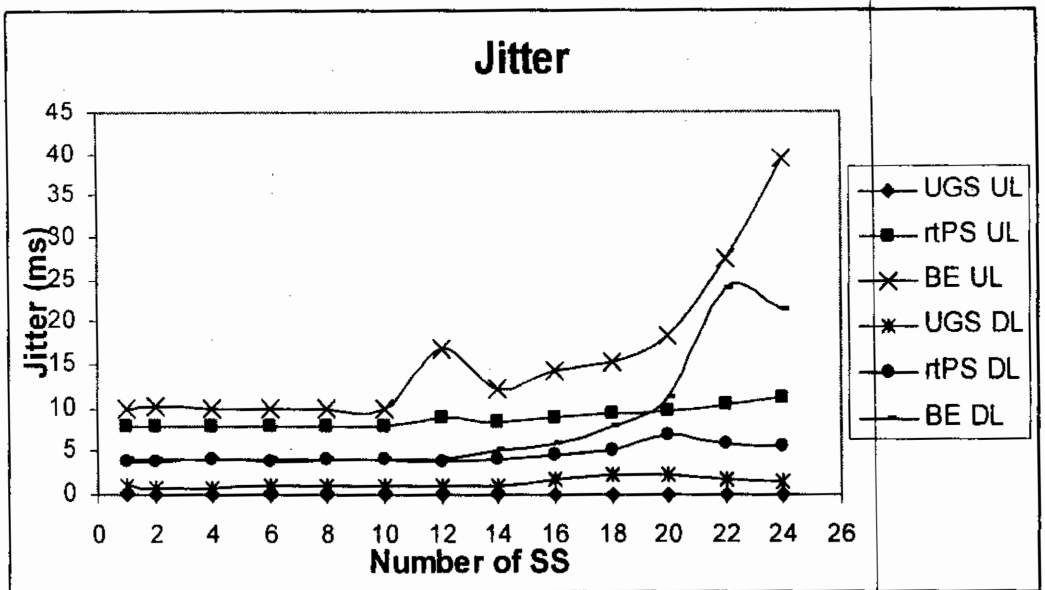

Fig 6.28: Jitter of Scheduling Services (Uplink & Downlink)

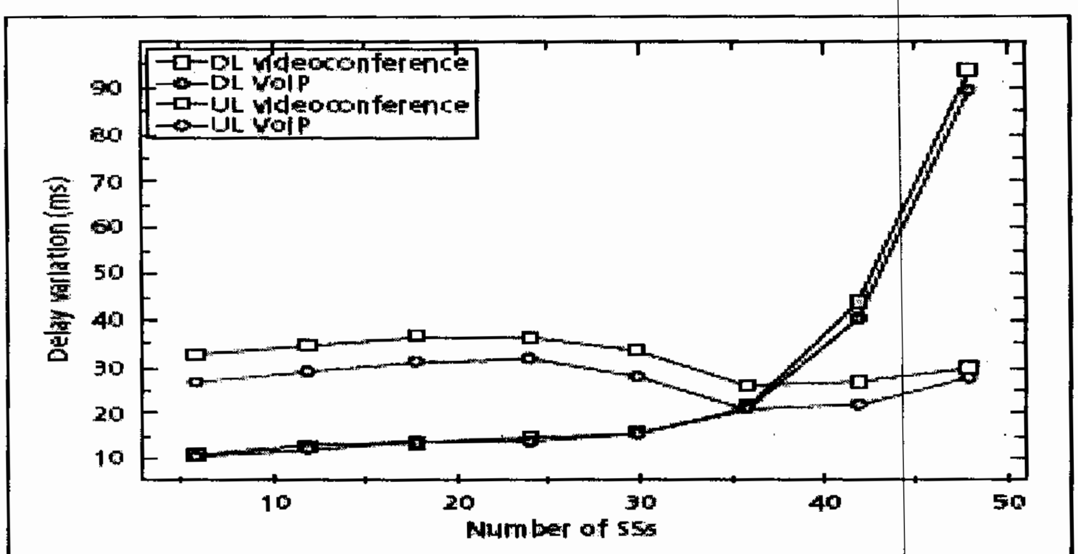

Fig 6.29: Delay Variation Vs. Number of SS [10]

Fig 6.28 show that jitter of UGS and rtPS little bit increases as the number of SS increases and it is compare with fig 6.29 which show that UL VoIP and UL Videoconference are slight change but in downlink direction it gradually increase after overloaded condition. As we can say that our scheduling architecture schedule packets as early as possible and try to achieve minimum delay and jitter under overloaded condition.

At the end we can say that our scheduling architecture show better result to achieve Quality of service for delay tolerant and bandwidth guarantees traffic. It also provides good performance under overloaded condition and tries to accommodate more service flows which show the scalability of the system. Another scheduling architecture [10] occupies a fixed data rate to achieve better quality of service which effects the newly enter service flows and in our case there is no fixed allocation except UGS which accommodates more service flows to some extent until existing flows show bad quality of service. So we can say that our scheduling architecture is fair and efficient.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

In this thesis we proposed a scheduling architecture IEEE 802.16 standard for uplink and downlink direction. Our proposed architecture support QoS requirement of all four uplink service flow that mentioned in IEEE 802.16 standard. We incorporate QoS parameter that associated with each scheduling service and schedule their packets according to these parameters. Our scheduling architecture includes parameters like maximum sustained rate, maximum latency, tolerated jitter, minimum reserved bandwidth, traffic priority, request transmission policy, burst size, SDU size and queue information for various applications. We use First in First out (FIFO), Earliest Deadline First (EDF) scheduling algorithm to schedule packets of a flow and used Self Clocked Fair Queuing (SCFQ) to schedule packets different flows to achieve QoS and efficient bandwidth utilization. We also associate weights with scheduling service to achieve fairness, that is calculated by queue information and priority associated with that flows. We concluded that Uplink scheduling is more difficult as compared to Downlink scheduling. Downlink scheduling is easy because BS has all the information about flow and updated queue information so we can easily schedule packets.

For uplink scheduling BS has only information that QoS parameter that negotiated during connection setup so its best way for BS to allocate bandwidth on basis of these parameter and SS allocate this bandwidth according to queue status and QoS parameters to achieve quality of service. So we concluded that to achieve quality of service, efficient utilization of bandwidth and fairness is the best way to design such architecture that incorporate all QoS parameters, so a flow can meets its delay and bandwidth guarantee. This approach can save a BS from lot of computation and databases management for each flow. For successful implementation a good admission control policy is also required.

Our simulation result show that bandwidth allocated to high priority flow such like UGS and rtPS as we increase number of flows, so their QoS guarantee are always meet. A lower priority flows does not affect QoS of high priority flows. Simulation result shows that fairness is maintained among flows and among subscriber station. Simulation result about fragmentation under this architecture show that fragmentation plays an important role in efficient bandwidth utilization.

Finally, we concluded that such architecture is best for varied quality of service requirement of different flows in uplink and downlink direction.

7.2 Future Work

Our work can be extended in following dimensions.

Scheduling Algorithms: we have used SCFQ as a scheduling algorithm, so a comparative study can be done on different scheduling algorithm and find out which one is the best.

Admission Control: A good admission control policy can be integrated with this architecture and combined performance study can be carried out.

Packing, Concatenation and Contention: These features can be added to current architecture and evaluate their performance.

Fragmentation: A separate study can be carried out on fragment size, which fragment size is best for variable size traffic, so fragment header overhead is tolerable and bandwidth utilization is also maximize.

Dynamic Uplink and Downlink subframe Allocation: A dynamic scheme can be developed for distribution of uplink and downlink subframe transmission on basis of traffic load without compromise of uplink and downlink flows quality of service.

An extensive simulation study with complex scenario also required to check efficiency of this architecture.

We can also implement this architecture on Mesh mode and IEEE 802.16e mobile standard and evaluate the result.

References

[1] M. Hawa and D. W. Petr, "Quality of service scheduling in cable and broadband wireless access systems," 10th IEEE International Workshop on Quality of Service, May 2002, pp. 247–255.

[2] Supriya Maheshwari, "An Efficient QoS Scheduling Architecture for IEEE 802.16 Wireless MANs," Masters Thesis, IIT Bombay, 2005.

[3] Guosong Chu, Deng Wang, and Shunliang Mei. "A QoS architecture for the MAC Protocol of IEEE 802.16 BWA System." IEEE International Conference on Communications Circuits and System and West Sino Expositions, vol.1, pp.435–439, China, 2002.

[4] Jianfeng Chen, Wenhua Jiao, Qian Guo, "Providing integrated QoS control for IEEE 802.16 broadband wireless access systems," Vehicular Technology Conference, 2005. VTC-2005-Fall. 2005 IEEE 62nd, Volume 2.

[5] L. F. M. de Moraes, P. D. Maciel Jr., "Analysis and Evaluation of a New MAC Protocol for Broadband Wireless Access," International Conference on Wireless Networks, Communications, and Mobile Computing - WirelessCom 2005, Maui, Hawaii, USA. June of 2005.

[6] Aura Ganz and Kitti Wongthavarawat, "IEEE 802.16 based last mile broadband wireless military networks with quality of service support." IEEE Milcom 2003, vol.2 pp.779–784.

[7] Howon Lee, Taesoo Kwon, Dong-Ho Cho, "'An efficient uplink scheduling algorithm for VoIP services in IEEE 802.16 BWA systems," IEEE Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th, Volume 5.

[8] Sung-Min Oh, Jae-Hyun Kim, "The analysis of the optimal contention period for broadband wireless access network," Pervasive Computing and Communications Workshops, 2005.

[9] Abhishek Maheshwari , "Implementation and Evaluation of a MAC Scheduling Architecture for IEEE 802.16 WirelessMANs," Masters Thesis, IIT Kanpur, 2006

[10] Claudio Cicconetti, Luciano Lenzini, Enzo Mingozzi and Eklund, "Quality of Service Support in IEEE 802.16 Networks" IEEE Network, March/April 2006

[11] G.S. Paschos, I. Papapanagiotou, C.G. Argyropoulos and S.A. Kotsopoulos, "A Heuristic Strategy for IEEE 802.16 WiMAX scheduler for Quality of Service" 45th Congress FITCE 2006, 30 August - 2 September 2006, Athens, Greeces

[12] Seungwoon Kim and Ikjun Yeom. "TCP-aware Uplink Scheduling for IEEE 802.16" IEEE COMMUNICATIONS LETTERS, 2006

[13] Victor Rangell, Javier Ortiz and Javier Gomez. "Performance Analysis of QoS Scheduling in Broadband IEEE 802.16 Based Networks" OPNETWORK 2006 technology conference, August 2006, Washington D.C.s

[14] Xiaojing Meng "An Efficient scheduling for Diverse QOS Requirements in WiMAX" Master Thesis, Waterloo, Ontario, Canada, 2007.

[15] Claudio Cicconetti, Alessandro Erta, Luciano Lenzini, and Enzo Mingozzi "Performance Evaluation of the IEEE 802.16 MAC for QoS Support" IEEE Transaction on Mobile Computing, Vol. 6, No. 1, January 2007

[16] IEEE 802.16-2004, IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Fixed Broadband Wireless Access Systems, IEEE, Oct. 1,2004.

[17] Jenhui Chen, Chih-Chieh Wang, Frank Chee-Da Tsai, Chiang-Wei Chang, Syao-Syuan Liu, Jhenjhong Guo, Wei-Jen Lien, Jui-Hsiang Sum, and Chih-Hsin Hung "The Design and Implementation of WIMAX Module for ns–2 Simulator" ACM International Conference Proceeding Series; Vol. 202 Proceeding from the 2006 workshop on ns-2: Pisa, Italy 2006

[18] Vandana Singh and Vinod Sharma "Efficient and Fair Scheduling of Uplink and Downlink in IEEE 802.16 OFDMA Networks" in Wireless Communications and Networking Conference, 2006. WCNC 2006. IEEE, Volume: 2, pp 984-990

[19] Demers A,Keshav S and Shenker S "Analysis and Simulation of a Fair Queuing Algorithm" in Proc. ACM SIGCOMM'89, 1989, pp 3–12.

[20] Parekh,A.K.and Gallager "A generalized processor sharing approach to flow control in integrated services networks the singlenode case" IEEE/ACM Trans. Networking, vol. 1, pp 334–357, June 1993.

[21] Maria Adamou, Sanjeev Khanna, Insup Lee, Insik Shin and Shiyu Zhou "Fair Real-time Traffic Scheduling over A Wireless LAN" Real-Time Systems Symposium, 2001. (RTSS 2001). Proceedings. 22nd IEEE Volume , Issue , 3-6 Dec. 2001, pps 279 - 288

[22] http//www.wimaxforum.com

[23] http//www.wikipedia.com

[24] R.Jain "The Art of Computer Systems Performance Analysis: Technique for Experimental Design, Measurement, Simulation and Modelling", John Wiley and Sons, Inc., 1991, s. 35-37

[25] http://www.isi.edu/nsnam/ns/ns-tutorial/ucb-tutorial.html 5th UCB/LBNL Network Simulator (NS): June 1999 Tutorial

[26] Network Simulator, NS–2, http://www.isi.edu

[27] National Institute of Science and Technology: http://www.nist.edu

[28] http://ndsl.csie.cgu.edu.tw

[29] www.cs.unt.edu/~rdantu/FALL_2005_WIREESS_NETWORKS/wimax.ppt

Appendix A

List of Abbreviations

BE:	Best Effort Service Flows
BPSK:	Binary Phase Shift Keying
BS:	Base Station
BWA:	Broadband Wireless Access
CBR:	Constant Bit Rate
CID:	Connection Identifier
CPS:	Common Part Sublayer
CS:	Convergence Sublayer
DL:	Downlink
DSL:	Digital Subscriber Line
EDF:	Earliest Deadline First
FCFS:	First Come First Serve
FIFO:	First In First Out
FTP:	File Transfer Protocol
GPC:	Grant per Connection
GPS:	Generalized Processor Sharing
GPSS:	Grant per Subscriber Station
HTTP:	Hyper Text Transfer Protocol
IEEE:	Institute of Electrical and Electronics Engineering
MAC:	Medium Access Control Layer
MPDU:	Mac Protocol Data Unit
MPEG:	Moving Picture Expert Group
MSDU:	Mac Service Data Unit
nrtPS:	Non Real-Time Polling Service Flows
NAM:	Network Animator
NP:	Non-Deterministic Problem
NS2:	Network Simulator 2
OFDM:	Orthogonal Frequency Division Multiplexing
OFDMA:	Orthogonal Frequency Division Multiple Access
OTcl:	Object–Oriented Transcripting Language
PGPS:	Packet Generalized Processor Sharing
PDU:	Protocol Data Unit
PMP:	Point to Multi-point
QAM:	Quarter Amplitude Modulation
QoS:	Quality of Service
QPSK:	Quaternary phase shift keying
rtPS:	Real-Time Polling Service Flows
SCFQ:	Self Clocked Fair Queuing
SDU:	Service Data Unit
SFID:	Service Flow Identifier
SS:	Subscriber Station

TCL:	Transcripting Language
TDMA:	Time Division Multiple Access
UGS:	Unsolicited Grant Service Flows
VBR:	Variable Bit Rate
UL:	Uplink
VoIP:	Voice over IP
WFQ:	Weighted Fair Queuing
WiMax:	Worldwide Interoperability for Microwave Access
WirelessMAN:	Wireless Metropolitan Area Network
WLAN:	Wireless Local Area Network

Appendix B

Wireless Technologies

Wireless technologies collection from global voice and data networks to infrared light and radio frequency technologies optimized for short-range wireless connections. Devices are commonly used for wireless networking include portable computers, desktop computers, handheld computers, PDAs, cellular phones, pen-based computers and pagers. Wireless technologies have evolved substantially over the past few years and, depending on their range, can be classified in different ways.

Wireless Wide Area Network

This network is designed to enable users to access the Internet via a wireless wide area network (WWAN) access card. Data rates are very fast as compared with the data speeds of mobile telecommunications technology, and their range is also extensive. Cellular networks based on CDMA are good examples of WWAN.

Wireless Local Area Network

This kind of network is designed to enable users to access the Internet in localized hotspots via a wireless local area network (WLAN) access card and a PDA or laptop. Data speeds are relatively fast as compared with the data speed of mobile telecommunication technology. WLAN range is limited. Wi-Fi is the most widespread and popular WLAN technology.

Wireless Personal Area Network

This type of network is designed to enable the users to access the Internet via a wireless personal area network (WPAN) access card and a PDA or laptop. Data speeds are very fast as compared with the data rates of mobile telecommunications technology and their range is very limited.

Wireless Metropolitan Area Network

WMAN network is designed to enable the users to access the Internet and multimedia services via a wireless metropolitan area network (WMAN). Data rates are very fast as compared with the data rates of mobile telecommunication technology as well as other wireless network, and their range is also extensive. The charter of the WMAN working group is to 'develop a standard for a cognitive radio-based air interface for use by license-exempt devices on a non-interfering basis in spectrum that is allocated to the TV Broadcast Service'. WiMax is good example of WMAN network.

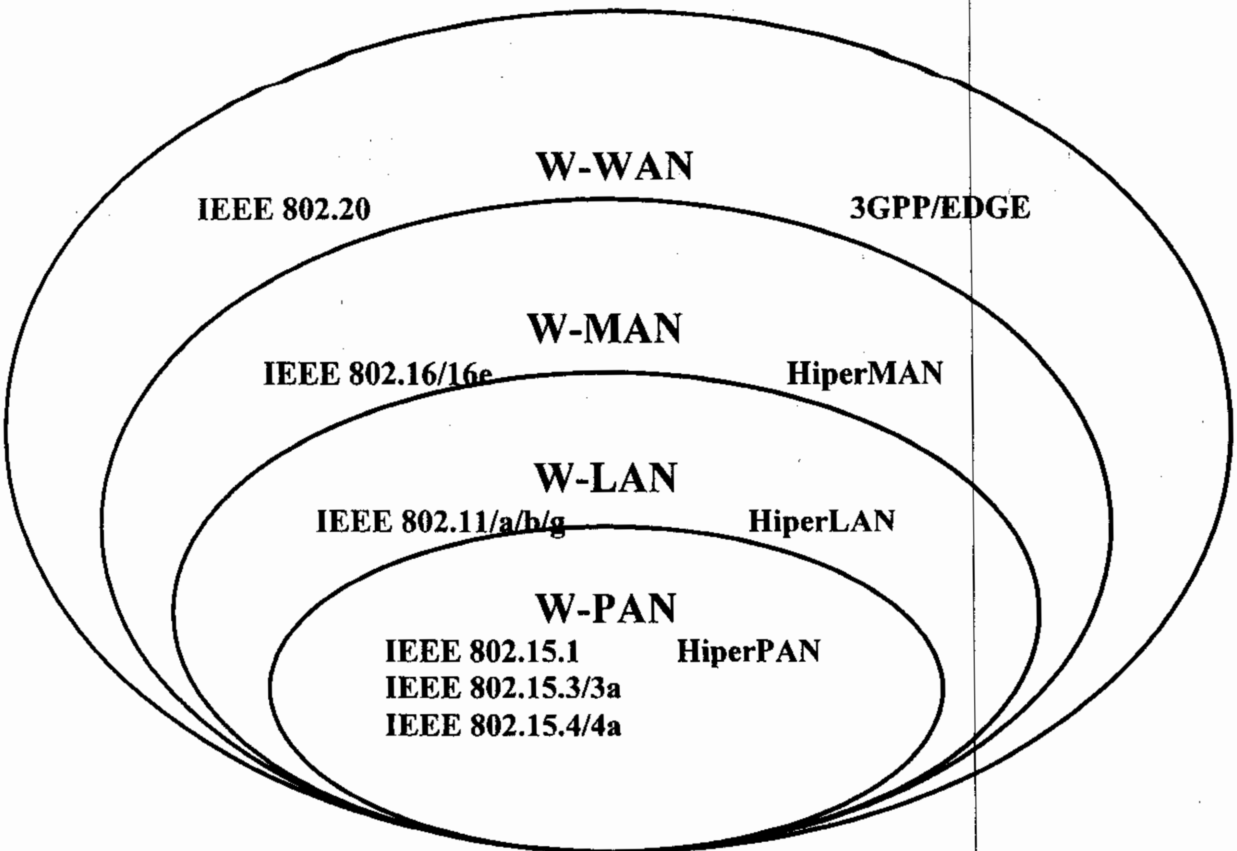

Fig B1: Global Wireless Standard [22]

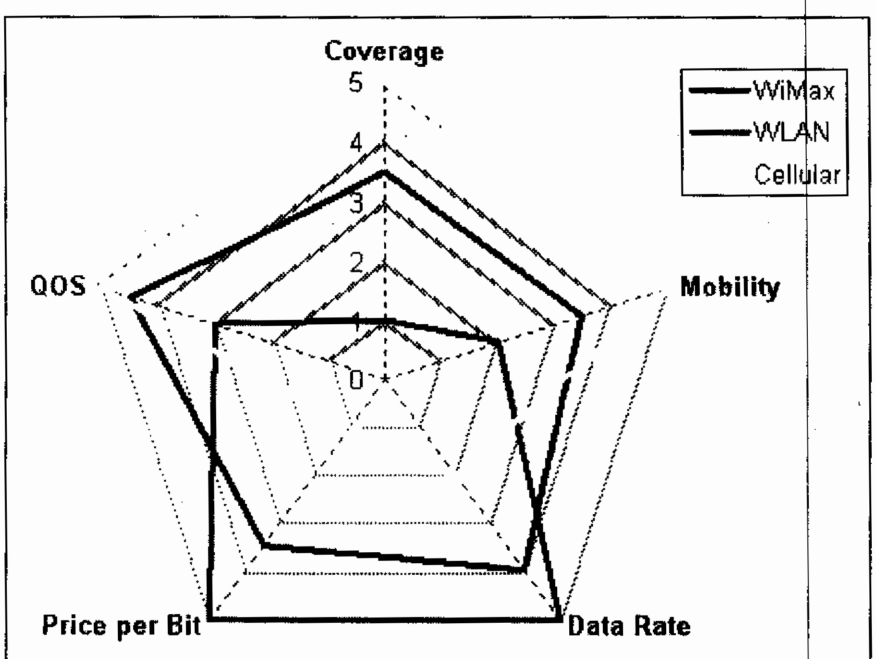

Fig B2: WiMax vs. WLAN and Cellular [29]

WiMax as Last/First Mile Tecnology

WiMAX is:

1. A wireless technology optimized for the delivery of IP centric services over a wide area network.
2. A certification that denotes interoperability of equipment built to the IEEE 802.16 or compatible standard. The IEEE 802.16 Working Group develops standards that address two types of usage models: a fixed usage model (IEEE 802.16-2004) and a portable usage model (802.16 REV E, scheduled for ratification in 2005).
3. A scaleable wireless platform for constructing alternative and complementary broadband networks.

The air interface standard, IEEE 802.16 is a specification for fixed broadband wireless metropolitan access networks (MANs) that use a point-to-multipoint architecture. Published on 8 April 2002, this standard describes the use of bandwidth between the 10 and 66 GHz licensed and between the 2 and 11 GHz unlicensed frequency and it also defines a MAC layer that supports multiple physical layer specifications. 802.16 support very high data rates in both uplink to and downlink from a base station. This standard covers a distance of 30 miles, in order to handle such services as VoIP and data.

WiMAX is a standardized wireless version of Ethernet and it consider as an alternative to wire technologies (such as cable modems, DSL and T1/E1 links) to provide broadband access to users. This application is called wireless "*last/first mile*" broadband because the transmission distances involved are typically of this order, and the engineering problem is to bridge the final gap between the user premises and the telecom or service provider's of main network. WiMAX is the Worldwide Microwave Interoperability Forum, a non-profit industrial body dedicated to promoting the adoption of this technology and ensuring that different vendors' products will interoperate. The 802.16 standard is large, complicated and evolving year by year, and offers many options and extensions, so interoperability is a major issue that must be addressed. In particular, one extension known as 802.16a became the focus of much industry attention because it is the easiest and most useful to implement. It is likely that when people talk loosely of WiMAX they are referring to the technology for fixed wireless specified by 802.16a and its later version 802.16d.

WiMAX can satisfy a variety of access needs. Potential applications include extending broadband access capabilities to bring them closer to user premises, filling gaps in cable, DSL and T1 services, Wi-Fi and cellular backhaul, providing last-100 m access from fiber to the curb and giving service providers another cost-effective option for supporting broadband services (Figure B3). WiMAX can support very high bandwidth solutions where large spectrum deployments. It can power existing infrastructure, keeping costs down while delivering the bandwidth needed to support a full range of high-value, multimedia services. WiMAX can also help service providers to meet many of the challenges they face due to increasing customer demands without discarding their existing infrastructure investments because it has the ability to seamlessly interoperate across various network types.

WiMAX can provide wide area coverage and quality of service capabilities for applications ranging from real-time delay-sensitive voice-over-IP (VoIP) to real-time streaming video and non-real-time downloads, ensuring that users obtain the performance they expect for all types of communications. WiMAX is proposed to serve as the next step in the evolution of 3G mobile phones, via a potential combination of WiMAX and CDMA standards called 4G.

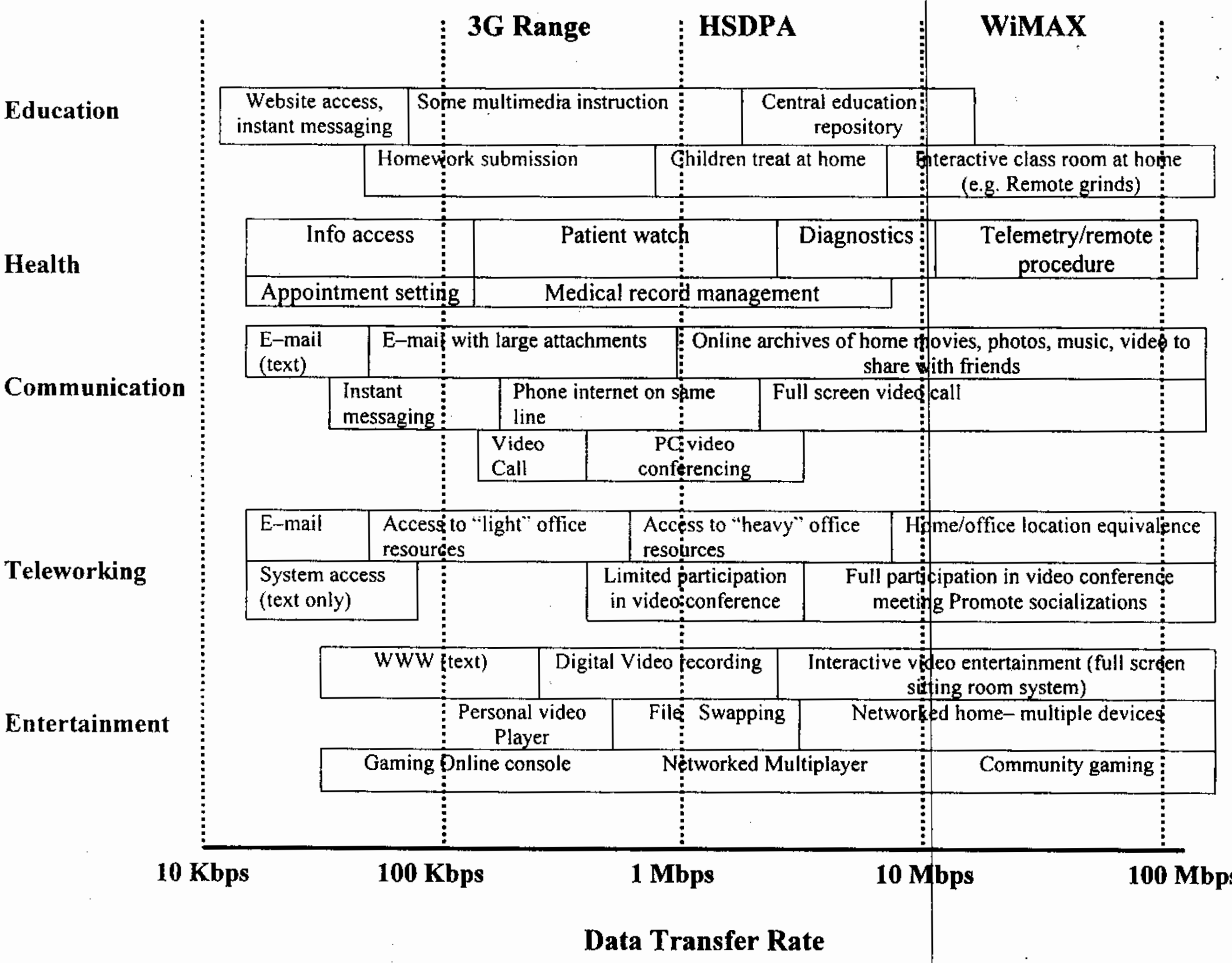

Fig B3: Application using BWA [22]

IEEE 802.16 Family

This technology aims to provide fixed broadband wireless access to residential and small business applications, as well as enable Internet access in countries without any existing wired infrastructure in place. Standardization efforts are also underway for the 802.16e version that attempts to provide mobility to the end user in a MAN environment. The WiMAX forum is a non-profit association formed in 2003 by equipment and component suppliers to promote the adoption of IEEE 802.16 compliant equipment by operators of broadband wireless access systems.

Table B1: Overview of the different variants within the 802.16 standard. [22]

	802.16	802.16a	802.16REVd or 802.16-2004	802.16e
Approved	Dec. 2001	Jan. 2003	July 2004	App. July 2005
Spectrum	10 - 66 GHz	< 11 GHz	< 11 GHz	2 - 6 GHz
Propagation	LOS	NLOS	NLOS	NLOS
Modulation	QPSK, 16QAM и 64QAM	OFDM 256, OFDMA + 802.16	OFDM 256, OFDMA + 802.16	OFDM 256, OFDMA + 802.16
Speed	32 – 134 Mbps	1 – 75 Mbps	Like 802.16a	Up to 15 Mbps
Mobility	No	No	No	Yes, with roaming
Channel bandwidth	20, 25 and 28 MHz	Variable from 1,25 up to 20 MHz	Like 802.16a	> 5 MHz
Cell size	1 - 5 km	5 – 8 km, max. is 50 km with directional antenna	Like 802.16a	1 – 5 km
Terminal		External with external antenna	External with internal antenna	PC card

Table B2:Comparison of WiMAX, WiFi and 3G technology [29]

	WiFi 802.11g	WiMAX 802.16-2004*	WiMAX 802.16e	CDMA2000 1x EV-DO	WCDMA / UMTS
Approximate maximum reach	100 meters	8kms	5kms	12kms	12kms
Approximate maximum throughput	54 Mbps	75 Mbps (20 MHz band)	30Mbps (10 MHz band)	2.4 Mbps (higher for EV-DV)	2Mbps (10+ Mbps for HSDPA)
Typical Frequency bands	2.4 GHz	2-11 GHz	2-6 GHz	400,800,900, 1700, 1800, 1900 2100 MHz	1800, 1900, 2100 MHz
Availability	Now	Ratified in June 2004, products in 2005	Expected ratification in Q3 2005, products in 2006	Now	Now
Application	Wireless LAN	Fixed Wireless Broadband (eg-DSL alternative)	Portable Wireless Broadband	Mobile Wireless Broadband	Mobile Wireless Broadband