Dynamic Bandwidth Scheduling and Burst Construction Algorithm for Downlink in (4G) Mobile WiMAX Networks

by

Jaskirat Singh

Supervisor: Dr. Hassan Naser

A thesis submitted to the

Faculty of Graduate Studies and Research

in partial fulfillment of the requirements for the degree of

Master of Science in

Electrical and Computer Engineering

Faculty of Engineering

Department of Electrical and Computer Engineering

Lakehead University

2013

Thunder Bay, Ontario, Canada 2013

Abstract

Advanced wireless systems, also called fourth generation (4G) wireless systems, such as Mobile Worldwide interoperability for Microwave Access (WiMAX), are developed to provide broadband wireless access in true sense. Therefore, it becomes mandatory for such kind of systems to provide Quality of Service (QoS) support for wide range of applications. In such types of systems, wireless base stations are responsible for distributing proper amount of bandwidth among different mobile users, thus satisfying a user's QoS requirements. The task of distributing proper amount of bandwidth rests upon a scheduling algorithm, typically executed at the base station.

2G and 3G wireless systems are able to provide only voice, low data rate, and delay insensitive services, such as Web browsing. This is due to the lack of development in digital modulation and multiple access schemes, which are two major aspects of physical layer of these systems. Digital modulation is used to combat with location-dependent channel errors which get introduced in the data transmitted by base station on a wireless channel to a mobile station. Hence, different locations of every mobile station in a cell coverage area require different modulation and coding schemes for error-free transmission. Link adaptation is a technique that makes the use of variable modulation and coding schemes possible, according to varying location of mobile stations. This technique is used by 4G systems to achieve error free transmissions. 2G and 3G systems are not capable of achieving error-free transmissions in many cases due to significantly fewer or no choice of modulation and coding schemes for different locations of mobile stations. In such cases, most of the time, wireless channel is either error-prone or error-free for mobile station.

Scheduling algorithms developed for 2G and 3G systems focussed on providing long term average rate requirements of users, which are satisfied at the expense of zero transmission for mobile users experiencing bad or error prone channel. This approach was adopted to achieve efficient use of wireless channel capacity. This was the best approach adopted by majority of scheduling algorithms because delay sensitive applications were not supported in such systems and hence bounded delay was not a matter of concern. Hence, the majority of the algorithms focussed on providing long term average rate requirements while maximizing cell throughput. This helped in making efficient use of wireless channel capacity at the expense of zero transmission for mobile users experiencing bad channel and compromising delay performance.

These approaches, however, will not be suitable for 4G systems as such systems support wide range of applications ranging from delay-insensitive to highly delay-sensitive. Hence in this thesis, a dynamic bandwidth scheduling algorithm called Leaky Bucket Token Bank (LBTB) is proposed. This algorithm exploits some advanced features of 4G systems, like link adaptation and multiple access scheme, to achieve long term average rate requirements for delay-insensitive applications and bounded delay for delay-sensitive applications.

Advanced features of 4G systems also bring more challenges. One such challenge is Orthogonal Frequency Division Multiple Access (OFDMA), a multiple access scheme deployed in 4G systems. In OFDMA, scheduled data for different mobile stations is packed into bursts and mapped to a two dimensional structure of time and frequency, called OFDMA frame. It has been observed that the way bursts are mapped to OFDMA frame affects the wakeup time of mobile stations receiving data and therefore causes power consumption. Wakeup time is the time duration in OFDMA frame for which the mobile station becomes active. Since OFDMA frame is a limited and precious radio resource, the efficient use of such radio resource is necessary. Efficient use requires that the wastage of such radio resource be minimized. Hence in this thesis, a burst construction algorithm called Burst Construction for Fairness in Power (BCFP) is also proposed. The algorithm attempts to achieve fairness in power consumption of different mobile stations by affecting their wakeup time. It also attempts to minimize wastage of radio resources.

For comparing the performance of joint proposed algorithms (LBTB+BCFP), the proposed burst construction algorithm (BCFP) is joined to the two other existing scheduling algorithms namely: Token Bank Fair Queuing (TBFQ) and Adaptive Token Bank Fair Queuing (ATBFQ). TBFQ is an algorithm developed for 3G wireless networks whereas, ATBFQ is an extension to the TBFQ and is developed for 4G wireless networks. Therefore, the performance of the proposed algorithms jointly together (LBTB+BCFP) is compared with the joint TBFQ and proposed burst construction algorithm (TBFQ+BCFP), as well as joint ATBFQ and proposed burst construction algorithm (ATBFQ+BCFP). We compare the performance in terms of average queuing delay, average cell throughput, packet loss, fairness among different mobile users, fairness in average wakeup times (average power consumption), and fraction of radio resources wasted. The performance of proposed burst construction algorithm (BCFP) is also compared with Round Robin algorithm in terms of fairness in average power consumption as well as fraction of radio resources wasted, for varying number of users.

Acknowledgements

First and foremost, I would like to express my sincerest appreciation to my thesis Supervisor, Dr. Hassan Naser, especially for his patience and then his guidance in achieving this work. I also like to thank him for his encouragement during the phases, when I felt down or faced a crisis. I would also like to thank all the other reviewers whose advice helped me in making all proper corrections, which improved the quality of my thesis.

In addition, I am very thankful to one of my lab mates and a good friend, Bona Ater for sitting patiently with me and helping me debugging the simulations. He is a real expert in any kind of simulation development. I am also very much thankful to one more of my good friends, Rositsa Gergova, for her assistance in improving my writing. It was not possible for me without her help to come out with such a properly written thesis.

Last but not the least; I would like to recognize my family for their continual encouragement and understanding and without whose support this work would not have been possible. I appreciate their patience and support, especially during the phase when I was under time extension.

Dedications

First and foremost, I would like to dedicate first proposed work of my research to one of my closest friends, Late Pankaj Upreti. He has and will always remain one of my mentors, from whom I have learned to be more confident, humble, gained a never say die attitude, and to find positive, even in negative situations.

Secondly, I would like to dedicate second proposed work of my research to one of my good friends, Rositsa Gergova. Her tender friendship and support has always gave a secured feeling of having a friend for whom help I can always ask.

Finally, I would like to dedicate whole thesis to my family, especially to my father, who has become one of my mentors and will always be. He has always encouraged me to pursue what I enjoy, with as much perfection as possible. I have learned from him to become patient, humble, firm on my decisions, and consistently positive.

Contents

List of Figures

List of Tables

List of Acronyms

2G	Second Generation
3G	Third Generation
3G+	Beyond 3G
4G	Fourth Generation
ATBFQ	Adaptive Token Bank Fair Queuing
BCFP	Burst Construction for Fairness in Power
BE	Best Effort
CBQ	Class Based Queuing
CBR	Constant Bit Rate
CID	Connection Identifier
CIF-Q	Channel condition Independent packet Fair Queuing
CoS	Class of Service
CSDPS	Channel State Dependent Packet Scheduling
CS-WFQ	Channel State independent Wireless Fair Queuing
ertPS	extended real time Polling Service
FDD	Frequency Division Duplex
GPS	Generalized Processor Sharing
IWFQ	Idealized Wireless Fair Queuing
LBTB	Leaky Bucket Token Bank
MAC	Medium Access Control
MATS	Mapping with Appropriate Truncation and Sort

MRFQ	Multi Rate wireless Fair Queuing
MRTR	Minimum Reserved Traffic Rate
MSTR	Maximum Sustained Traffic Rate
nrtPS	non real time Polling Service
OCSA	One Column Stripping with non-increasing Area
OFDMA	Orthogonal Frequency Division Multiple Access
PGPS	Packet by packet GPS
QAM	Quadrature Amplitude Modulation
QoS	Quality of Service
QPSK	Quadrature Phase Shift Keying
rtPS	real time Polling Service
SBFA	Server Based Fairness Approach
SCFQ	Self Clocked Fair Queuing
SINR	Signal to Interference and Noise Ratio
SLA	Service Level Agreement
STR	Sustained Traffic Rate
TBFQ	Token Bank Fair Queuing
TDD	Time Division Duplex
UGS	Unsolicited Grant Service
WF^2Q	Worst-case Fair Weighted Fair Queuing
WFFQ	Wireless Fluid Fair Queuing
WFQ	Weighted Fair Queuing
WiMAX	Worldwide interoperability for Microwave Access

List of Symbols

K	Number of bits per data symbol
T_u	Duration of data time
T_g	Duration of guard time
T_s	Total duration of symbol time
S_{total}	Total OFDMA time symbols in Mobile WiMAX frame
$N_{subchannels}$	Number of subchannels in OFDMA resource matrix
N_{RB}	Total number of RBs in data region of OFDMA resource matrix
$b_{z,RB}$	Number of bits carried by a resource block for a user in zone z
$r_{z,RB}$	Data rate of a resource block for a user in zone z
$N_{z,RB}$	Number of resource blocks out of total resource blocks reserved for all the users in zone z
C	Total varying capacity of wireless channel in a scheduling round
D_{ic}	Average bucket depth of CoS c of user i
M_{ic}	Minimum bucket depth of CoS c of user i
P_{ic}	Maximum bucket depth of CoS c of user i
B_{ic}	Average rate of CoS c of user i
A_{ic}	Amount of bytes scheduled for CoS c of user i
$s_{ic,n}$	Amount of bytes scheduled for CoS c of user i in $n-th$ scheduling round
E_{ic}	Maximum amount of extra tokens that can be granted to lagging flow of CoS c of user i

$\alpha_{ic}E_{ic}$	Amount of extra tokens that is granted to a lagging flow of CoS c of user i
u_{ic}	The amount of bytes scheduled for a lagging flow of CoS c of user i
X_{ic}	The maximum amount of tokens by which a leading flow of CoS c of user i can be penalized
$\beta_{ic}X_{ic}$	The amount of tokens by which a leading flow of CoS c of user i can be penalized
d_{ic}	The amount of bytes scheduled for a leading flow of CoS c of user i
$B_{c,min}$	The lowest average rate among the individual CoS c sub-group within leading and lagging group
r_{lowest}	The lowest data rate of a resource block among the data rates of a resource block carrying scheduled data of leading and lagging flows
Q_{ic}	Queue length of CoS c of user i
C_r	Remaining tokens in the system
β_{ic}	Dynamic fraction of penalized bytes
α_{ic}	Dynamic fraction of excess bytes
n_{lead}	Number of backlogged flows in leading group
n_{lag}	Number of backlogged flows in lagging group
s_{jn}	Amount of scheduled bytes of user j in $n-th$ scheduling round
RB_{jn}	Number of resource blocks in a burst of user j in $n-th$ scheduling round
L_i	The number of empty resource blocks in strip # i
v_{jn}	The number of strips on horizontal time axis occupied by the burst of user j in $n-th$ scheduling round
h_{jn}	Burst delay of user j in $n-th$ scheduling round
a_{jn}	Area occupied by rectangular shape of burst of user j in $n-th$ scheduling round
o_{jn}	Over-allocated resource blocks in the area occupied by absolute rectangular shape of burst of user j in $n-th$ scheduling round
T_{jw}	Average wakeup time of user j averaged over past scheduling rounds

ρ	Location parameter for a random variable which is distributed by Pareto Distribution
β	Shape parameter for a random variable which is distributed by Pareto Distribution
T_{ON}	Average value of ON period
T_{OFF}	Average value of OFF period
β_{OFF}	Shape parameter for OFF period which is distributed by Pareto Distribution
β_{ON}	Shape parameter for ON period which is distributed by Pareto Distribution
ρ_{OFF}	Location parameter for OFF period which is distributed by Pareto Distribution
ρ_{ON}	Location parameter for ON period which is distributed by Pareto Distribution
$f_X(x)$	Probability Distribution Function
$E[X]$	Expected value of a random variable X
$F_X(x)$	CDF of a probability distribution
S_{avg}	Average SINR reported by a user in a coverage area
D	Distance between base station and a user in a coverage area

Chapter 1:

Introduction

As more smart devices are being introduced in the current market, the demand for different Quality of Service (QoS) needs for a multitude of applications is also increasing. Some of the most popular applications dominant in current 4G wireless systems include Web Browsing, Live Streaming and Mobile TV. These applications are becoming popular and feasible because of the several advanced features and enhancements made in the physical layer of 4G systems. Some of these enhancements that are necessary to be introduced for this thesis will be discussed in the next chapter. It will be concluded that inclusion of such enhancements has made highly delay-sensitive applications feasible. This has made such systems the primary choice of users to satisfy all of their needs. Hence, the number of users in 4G wireless networks are increasing at a very fast pace resulting in a huge load on 4G networks. This is posing a big challenge to satisfy different QoS requirements for the ever-increasing number of users simultaneously. One of the latest reports of Worldwide interoperability for Microwave Access (WiMAX) Forum [32] has shown that Mobile WiMAX subscriptions were 10 million in 2010, and they are expected to reach 130 million in 2014, globally.

To face this challenge, there is a need to develop a dynamic bandwidth scheduling algorithm which will be aware of the enhancements made in the physical layer. It should distribute bandwidth dynamically among different users based on several factors such as QoS, location of user with respect to the wireless base station, fairness, and service level agreements [34].

Since most of the smart devices today are battery operated, the power consumption of such devices becomes a major concern. One of the enhancements made in 4G systems is Orthogonal Frequency Division Multiple Access (OFDMA) [35]. In OFDMA, data for several mobile stations is transmitted simultaneously and their data get mapped in the form of bursts to OFDMA frame. Each burst is composed of one or more elementary units called *resource blocks*. A constraint of *absolute rectangular shape* of a burst is set by IEEE 802.16e-2009 PHY specification [33]. According to this constraint, bursts of different users should always occupy the space in OFDMA frame in the form of rectangular area with equal lengths and breadths. There are fixed number of resource blocks in OFDMA frame. Together, they constitute wireless channel capacity and therefore, a very valuable radio resource [35]. Therefore, minimization of wastage of resource blocks is necessary. The resource blocks can get wasted due to two possibilities: first, there are not enough left-over resource blocks for the user's burst to map. In such a case, the wasted resource blocks are called *unoccupied resource blocks*. Second, the absolute rectangular shape of a burst can also consist of some extra left-over resource blocks. In such a case, the wasted resource blocks are called *over-allocated resource blocks*. The wasted resource blocks in both possibilities contribute to resource blocks wastage. It will be concluded later that in downlink the wakeup time of mobile stations in a cell is affected by the way in which resource blocks are packed into bursts for mobile stations [3, 5]. It is then safe to say that power consumption of mobile stations in a cell is affected by the burst mapping of mobile stations [3, 5].

There is also a need to develop a burst construction algorithm that finds a good trade-off between resource blocks wastage and fairness in mobile station average power consumption.

The evolution of scheduling algorithms for wireless networks started from those of wired networks. This helped in understanding the need to develop specific scheduling algorithms for wireless networks because it was realized later that approaches deployed in wired networks are not suitable for wireless networks. This is because in wired networks, there are no location dependent channel errors, making data scheduled for transmission received error free, whereas in wireless networks, data scheduled for a specific mobile

station could not be received error free because of location dependent channel errors [1, 15]. Using wired networks scheduling algorithms for wireless environments will lead to wastage of bandwidth [1]. The basic approach deployed to solve this problem is deferring the transmission of mobile station when it experiences bad channel, and compensating the mobile station for all the lost bandwidth when it returns to experiencing good channel [1, 12, 13, 14, 15, 19].

Unlike 4G systems, a specific modulation scheme called Gaussian Minimum Shift Keying (GMSK) is used in 2G systems, such as Global System for Mobile (GSM) system. Similarly, 2.5G and 3G systems use a specific modulation scheme called Quadrature Phase Shift Keying (QPSK) [25]. The degree of location-dependent channel errors is defined by Signal to Interference and Noise Ratio (SINR). The less the value of SINR, the more error prone the channel. For example, a mobile station experiencing 5 dB of SINR has more error-prone wireless channel than a mobile station experiencing 10 dB of SINR. There will be a value or range of SINR through which error free channel or error free transmission is achieved. Therefore, using GMSK or a specific modulation and coding scheme as in 2G and 3G systems, error-free transmission is achieved for a specific value or range of SINR. Whereas for all other values of SINR, channel is always considered bad because there is no optional modulation and coding scheme which can achieve error free transmissions for all other values of SINR.

Beyond 3G (3G+) and 4G systems, channel cannot be good or bad because these systems use link adaptation. Therefore, for different values or range of SINR, the system uses a variable modulation and coding scheme, which achieves error free transmission in all range of SINR. This is the reason that regardless of the location of mobile station with respect to base station, the channel condition appears to be good no matter what value or range of SINR of wireless channel, mobile station experiences. Approaches similar to deferring of transmissions are not feasible for 4G systems because channel always appears to be good in case of 4G systems.

Some algorithms were developed to adapt bandwidth scheduling according to 3G+ wireless systems [2]. These works diverted from traditional approaches similar to deferring

of transmissions. They focussed on scheduling necessary bandwidth for each Mobile Station proportionately, according to modulation and coding scheme used.

1.1. Motivation and Contributions of Thesis

The motivation behind this thesis is to discuss and understand all the necessary enhancements made in the physical layer of Mobile WiMAX systems. This thesis investigates all the major scheduling algorithms for wired as well as wireless environments, which have led to the vast and continuous development of scheduling algorithms for different generations of wireless systems. Furthermore, this thesis also investigates all the major existing burst construction algorithms for Mobile WiMAX systems. The two contributions of this thesis are: First, developing a dynamic bandwidth scheduling algorithm for downlink in Mobile WiMAX systems. The developed algorithm will consider necessary enhancements made in the physical layer, service level agreements of users, QoS parameters, the location of mobile user with respect to base station, and fairness to schedule proper amount of bandwidth which satisfies the QoS requirements of a mobile user. Second, developing a burst construction algorithm that would be fair in terms of average power consumption, to all the users present in a coverage area of the cell. The proposed burst construction algorithm also tries to reduce wastage of resource blocks.

1.2. Organization of Thesis

In order to satisfy extensive QoS requirements for different mobile users in a wireless environment, a dynamic bandwidth scheduling algorithm should always at least consider all the necessary enhancements made in the physical layer of a communication system. In such a kind of approach, specifics of an algorithm depend on the physical layer of a system. Therefore, a good understanding of all the necessary enhancements made in Mobile WiMAX is required to develop a good dynamic bandwidth scheduling algorithm. Furthermore, it will also help in developing a burst construction algorithm.

In order to gain a better understanding of how, the specifics of a bandwidth scheduling algorithm depends upon the enhancements made in the physical layer of a wireless system, it is necessary to investigate major works in wired as well as wireless scheduling algorithms.

Hence, Chapter 2 provides a background for Mobile WiMAX radio interface. Chapter 3 investigates all major wired as well as wireless bandwidth scheduling algorithms. It also discusses different existing burst construction algorithms for Mobile WiMAX. In Chapter 4, our proposed bandwidth scheduling and burst construction algorithms are introduced. Simulation modelling and results for joint proposed algorithms are discussed in chapter 5. And finally, conclusion and future work is discussed in chapter 6.

Chapter 2:

Background

2.1. What is Mobile WiMAX?

WiMAX is one of the global wireless communication systems that provides broadband wireless access. It is based on IEEE 802.16 standard. WiMAX technology has undergone six revisions starting from 802.16a to 802.16e [29]. The IEEE 802.16d standard is called Fixed WiMAX. Mobile WiMAX is considered as a 'mobility' upgrade to the WiMAX technology and is based on IEEE 802.16e standard. It is considered to be one of the 4G global telecommunication systems.

2.2. The concept of 'Mobility'

Wireless access does not necessarily means 'mobility'. There is a big difference between fixed access, nomadicity, portability and *mobility*. In fixed access, a stationary user receives services on a wireless channel [29]. Nomadicity means that the user still would be able to get access to the services on a wireless channel, while moving within a small area of apartment or a campus [29]. Portability defines the notion of a user receiving wireless services within a coverage area, while moving over longer distances with speeds of between 15 and 50 Kilometres per hour [29]. Mobility is almost the same as portability except the user travels at a speed of between 50 and 150 Kilometres per hour [29]. It means that the location of user is changing very fast with respect to wireless base station.

2.3. Chapter Introduction

Broadband wireless access is a scenario where different types of services receive QoS similar to what they receive in alternative Digital Subscriber Line or cable modem [26]. Meeting a challenge of achieving complete broadband wireless access with an increasing number of users requires that QoS be maintained even in the presence of mobility. For addressing this challenge, IEEE 802.16e standard for Mobile WiMAX was finalized and released in 2009, and offers many advanced features which make this technology an ideal solution to broadband wireless access. It has more choice of modulation schemes and intelligence in selecting those modulation schemes. This leads to error free and bandwidth-efficient transmission because proportional amount of data can be transmitted according to channel conditions [31, 29]. All the previous systems were Frequency Division Duplex (FDD) whereas Mobile WiMAX can also be used in Time Division Duplex (TDD) configuration. Therefore, in Mobile WiMAX networks, same chunk of spectrum can also be used for downlink as well uplink [25, 26, 30, 35].

Furthermore, Mobile WiMAX systems use link adaptation and a unique kind of multiple access scheme called OFDMA. It will be concluded in later sections that using link adaptation along with OFDMA leads to varying capacity of resource blocks according to channel conditions [42]. The summarized advanced features of Mobile WiMAX systems, which make them one of the best solutions for implementing broadband wireless access, are:

- Link adaptation
- TDD – OFDMA (Multiple Access scheme)

In order to understand how the link adaptation and OFDMA make Mobile WiMAX robust to location dependent channel errors, it is necessary to introduce both the enhancements in sufficient detail. Therefore, these two advanced features of Mobile WiMAX systems will be discussed in detail in order to understand various advantages that these two features provide over previous systems.

2.4. IEEE 802.16e Systems Physical Layer

2.4.1. Digital Modulation

Digital (bandpass) modulation is a signal process that maps the information contained in the bit stream of shaped pulses, to the carrier waveform [31]. The information is mapped by changing the frequency, phase or amplitude of the carrier waveform [31]. This process is a bridge between digital and analog worlds, and translates digital information at baseband frequency into analog information at radio frequency.

There can be broadly three kinds of digital modulation schemes based on modulating the frequency, phase, or amplitude of the carrier waveform [31, 29]. These are called Frequency Shift Keying (FSK), Phase Shift Keying (PSK) or Amplitude Shift Keying (ASK) respectively [31]. The system divides the bit stream of shaped pulses into groups of fixed number of bits, where each group is called *data symbol*. The symbol mapper then maps the information contained in a data symbol by changing the frequency, phase or amplitude of carrier waveform, to the one in a finite set of frequencies, phases or amplitudes of the carrier waveform. If $\boldsymbol{K}$ is the number of bits per data symbol, the finite set contains $\boldsymbol{2^K}$ frequencies, phases or amplitudes to map from [30, 31]. All the schemes for which $\boldsymbol{K = 1}$, are called binary schemes and all the schemes for which $\boldsymbol{K > 1}$, are called $\boldsymbol{2^K - ary}$ schemes [30, 31]. Mobile WiMAX systems broadly use two kinds of modulation schemes that are either similar to or some variants of PSK. These two schemes are:

- Quadrature Phase Shift Keying (QPSK)
- Quadrature Amplitude Modulation (QAM)

These two modulation schemes will be now discussed in detail because it will be observed later that Mobile WiMAX uses QPSK as well as $16 - ary$ and $64 - ary$ variants of QAM with various coding rates to produce seven different modulation and coding schemes. The seven modulation and coding schemes are considered by link adaptation, while selecting the optimum scheme so that error free transmission for any SINR range can be achieved.

2.4.1.1. Quadrature Phase Shift Keying (QPSK)

Quadrature Phase Shift Keying is a Phase Shift Keying (PSK) scheme where the system divides the bit stream of shaped pulses into data symbols consisting of two bits. As shown in Figure 2.1, the first bit of each data symbol i.e. *b0, b2, b4 and b6* is assigned to a bit stream called I bit stream. The second bit of each data symbol i.e. *b1, b3, b5 and b7* is assigned to a different bit stream called Q bit stream. Each bit belonging to I as well as Q bit streams is called *stream data symbol*.

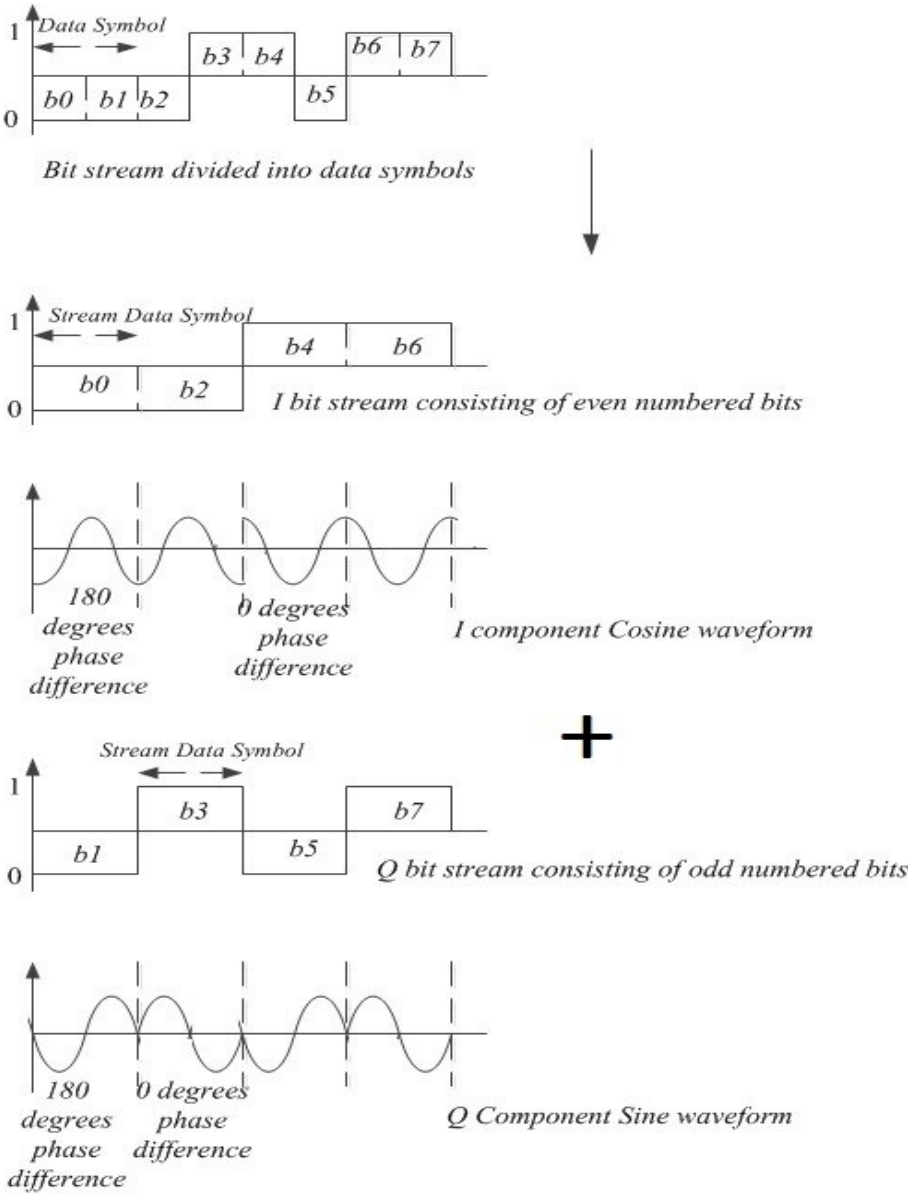


Figure 2.1: Formation of I and Q Components in QPSK

I and Q bit streams have the rate of half the original bit stream. Furthermore, as shown in figure 2.1, each I stream data symbol modulates cosine carrier waveform and the resultant carrier waveform is called I component. Each Q stream data symbol modulates sine carrier waveform and the resultant carrier waveform is called Q component.

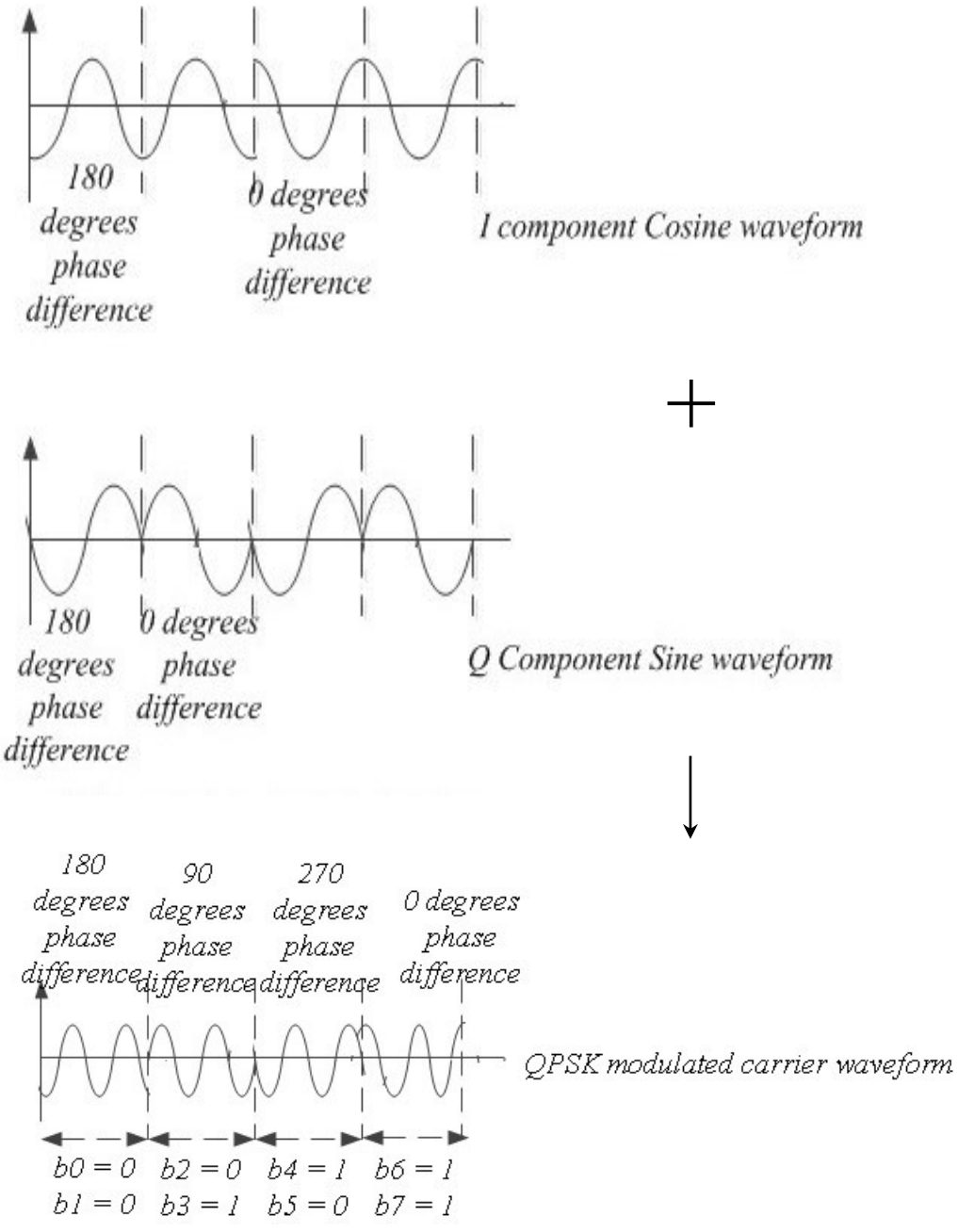


Figure 2.2: QPSK Modulation

As shown in Figure 2.2, when I and Q components are summed together they yield QPSK modulated carrier waveform having phase which varies according to the transition states of two bit data symbol. The transition states of the two bit data symbol are shown in Figure 2.3.

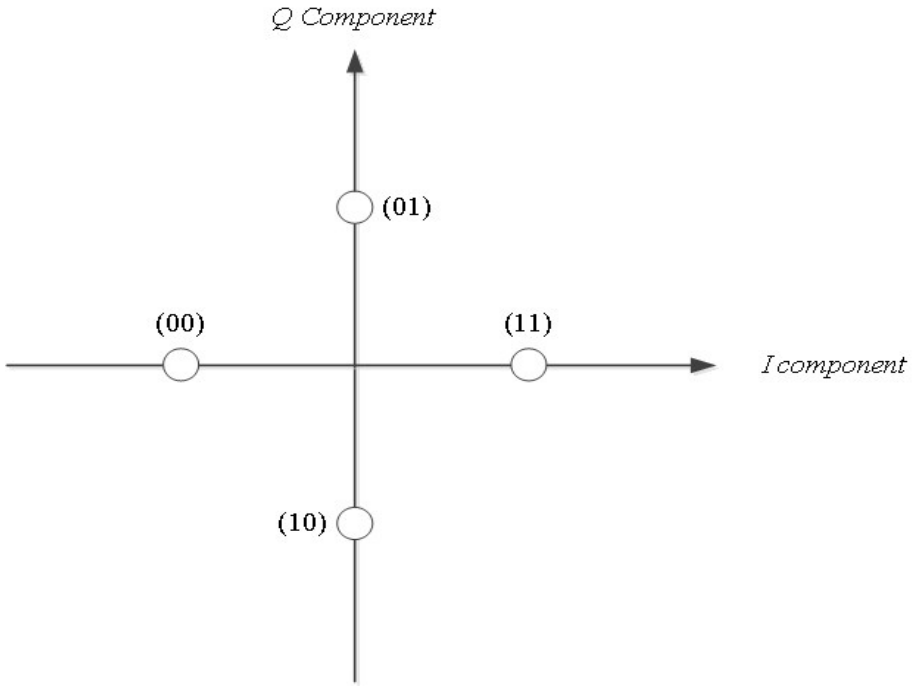


Figure 2.3: Transition states of a data symbol in QPSK Constellation

Since there are four different transition states for a two bit data symbol, therefore the symbol mapper maps the information contained in the data symbol by changing the phase of carrier waveform from the one in the set of four different phases of carrier waveform ($0^{\circ}, 90^{\circ}, 180^{\circ}, 270^{\circ}$).

2.4.1.2. Quadrature Amplitude Modulation (QAM)

In the previous section, it was observed that QPSK can transmit two bits per data symbol. More bits per data symbol can be transmitted by increasing the number of transition states in QPSK constellation. In this section, the idea of QPSK will be taken one step further and it will be shown how more information can be transmitted in each data symbol.

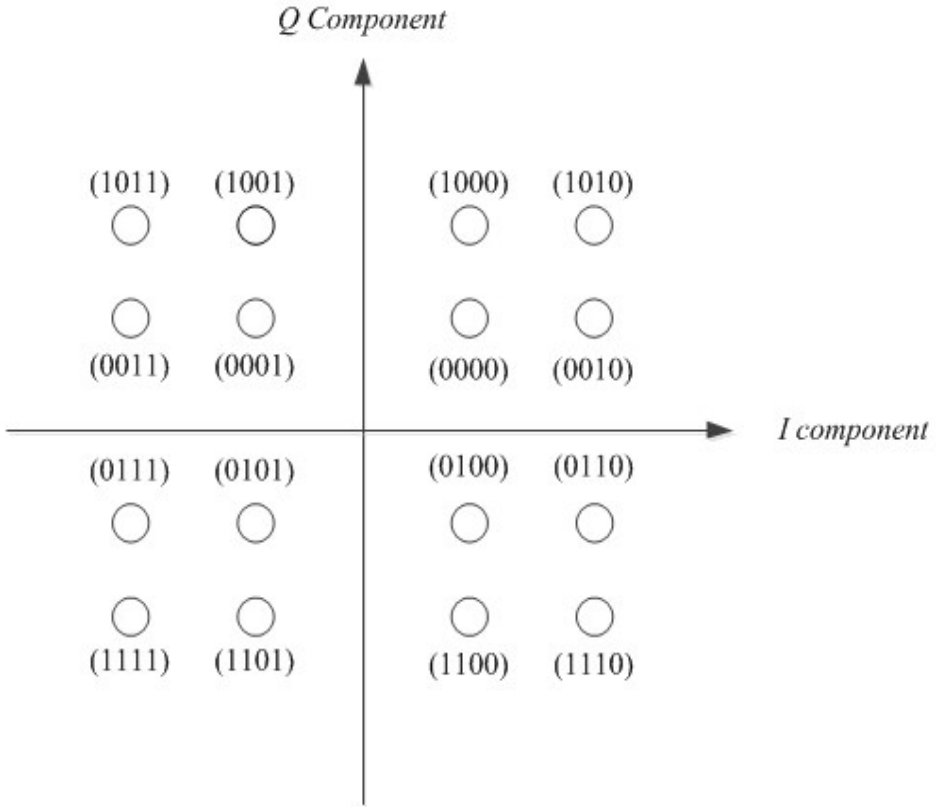


Figure 2.4: Transition states of a data symbol in 16-ary QAM constellation

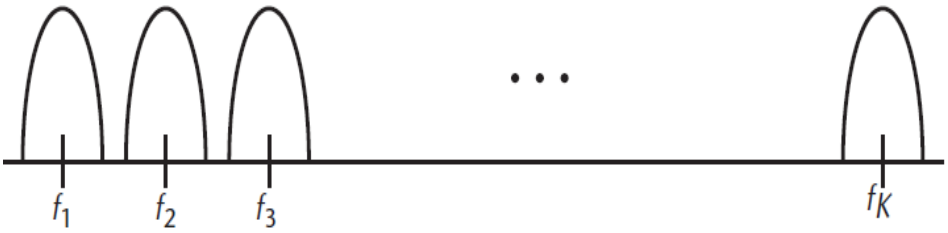
Quadrature Amplitude Modulation is also kind of PSK scheme which is used to transmit more information in each data symbol. Link adaptation in Mobile WiMAX uses $16-ary$ and $64-ary$ QAM, therefore each data symbol contains $K=4$ and $K=6$ bits per data symbol in $16-ary$ and $64-ary$ QAM, respectively. While mapping the information contained in the data symbol, QAM also follows the same process as explained in previous section. As each data symbol contains four bits in $16-ary$ QAM, therefore I and Q stream data symbols consist of two bits each. Furthermore, I stream data symbols phase modulate cosine waveform whereas Q stream data symbols phase modulate sine waveform. On the other hand, each data symbol consists of six bits in $64-ary$ QAM, therefore, I and Q stream data symbols consist of three bits each. Figure 2.4 shows the transition states of a four bit data symbol in rectangular constellation of $16-ary$ QAM. A larger constellation of $64-ary$ QAM can be constructed in the same way.

Errors can get introduced in the modulated analog waveforms that are phase modulated by data symbols. Therefore, there should be further processing on the signal

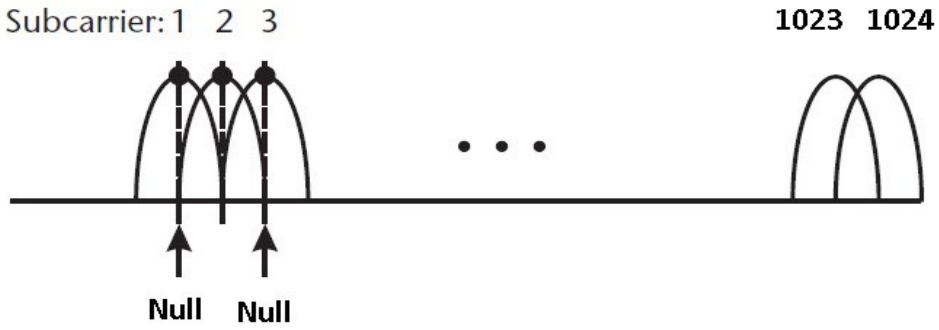
before it is transmitted over wireless channel. This further processing not only makes the signal more robust, but also adds a small code in the modulated data symbol which helps in detecting any errors introduced in the received carrier waveform and also helps in correcting those errors. Such types of codes are called Error Correcting Codes. The signal processing for introducing error correcting codes is called Channel Coding [31]. The error correcting codes are used with different coding rates of 1/2, 2/3 and 3/4 in Mobile WiMAX. The coding rate is an important parameter that decides the data rate of OFDMA resource block allocated to a mobile user. When coding rates of 1/2, 2/3 and 3/4 are used with QPSK, $16-ary$ as well as $64-ary$ QAM, then total of seven combinations are formed [29, 35]. These seven combinations are: 1/2-QPSK, 3/4-QPSK, 1/2-16-ary QAM, 3/4-16-ary QAM, 1/2-64-ary QAM, 2/3-64-ary QAM and 3/4-64-ary QAM. Hence, IEEE 802.16e Mobile WiMAX systems have the choice of seven modulation and coding schemes to select one of them using link adaptation. Global System for Mobile (GSM) systems or Enhanced Data rates for GSM Evolution (EDGE) or their counterparts have almost no choice of modulation and coding schemes for different SINR experienced by the user, when compared with Mobile WiMAX.

2.4.2. Orthogonal Frequency Division Multiple Access (OFDMA)

Mobile WiMAX uses an advanced multiple access technique called Orthogonal Frequency Division Multiple Access (OFDMA) to provide access to radio spectrum [35]. This technique is based on another technique called Orthogonal Frequency Division Multiplexing (OFDM) [35]. OFDM is a kind of Frequency Division Multiplexing (FDM) technique, where subcarrier frequencies in a spectrum are arranged in such a way that they are orthogonal to each other [27]. The property of being orthogonal in the context of frequency spectrum signifies that the peak value of one subcarrier frequency always meets at the null of the adjacent left and right subcarrier frequencies to it, as shown in figure 2.5. In this thesis, 10 Megahertz (MHz) of spectrum for Mobile WiMAX is considered. Since the physical layer of Mobile WiMAX defines 1024 subcarrier frequencies in 10 MHz bandwidth, 1024 subcarrier frequencies are arranged orthogonally [29, 35].



FDM Frequency Spectrum



OFDM Frequency Spectrum

Figure 2.5: Subcarrier frequencies in FDM and OFDM spectrum

The property of being orthogonal in OFDM has a big advantage over traditional FDM. Figure 2.5 shows the spectrum in case of FDM as well as OFDM. It can be observed that OFDM is much more spectral efficient than FDM because a larger number of subcarrier frequencies can be packed in the same amount of spectrum in comparison to FDM.

The physical layer of Mobile WiMAX defines different kinds of subcarriers. Figure 2.6 shows the different kinds of subcarriers. From 1024 subcarriers, 1 subcarrier that lies at the center of 10 MHz bandwidth is called DC subcarrier. There are 92 left guard carriers and 91 right guard carriers. The remaining $1024 - (92 + 91 + 1) = 840$ subcarriers are used for the purpose of sensing channel conditions experienced by users and transmitting the data of users in the downlink direction. Among the 840 subcarriers, there are 120 pilot subcarriers that are used for sensing the channel conditions experienced by users. The remaining $840 - 120 = 720$ subcarriers are called data subcarriers and they are used to transmit the data of users [29].

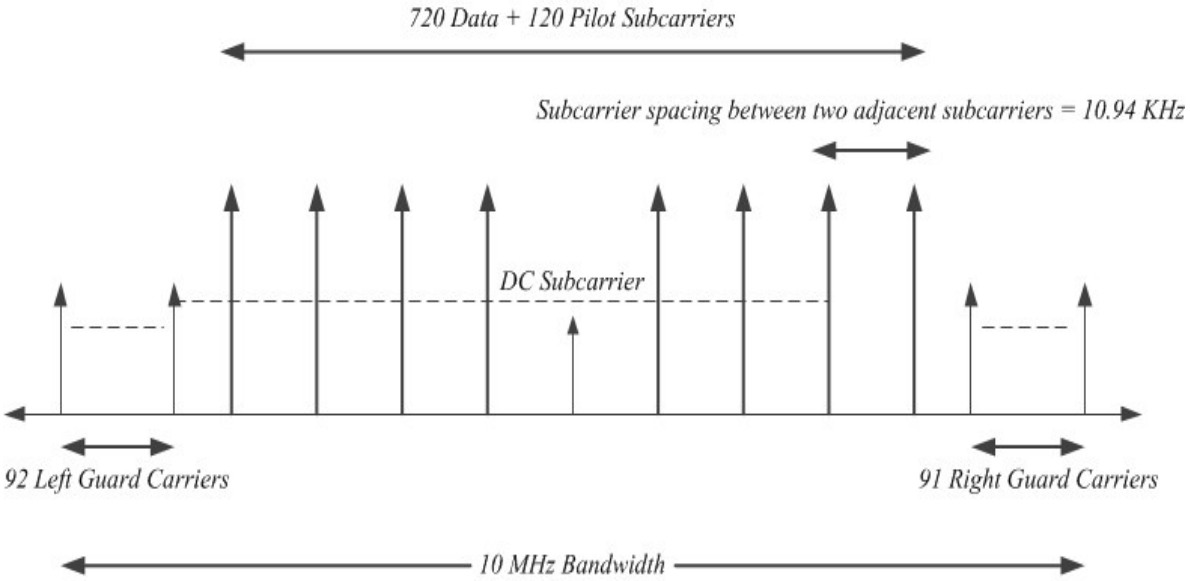


Figure 2.6: Different kinds of subcarriers

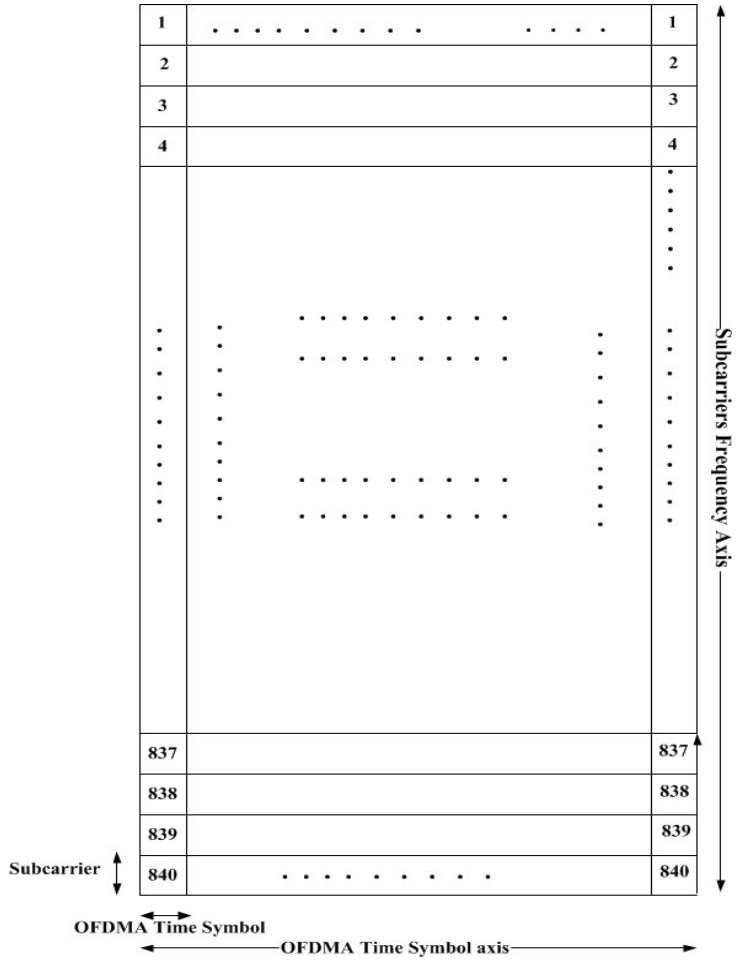


Figure 2.7: Time symbols and Subcarriers in Mobile WiMAX frame

As shown in Figure 2.7, each column along the horizontal time axis is called OFDMA time symbol and each row along the vertical frequency axis is called a subcarrier [35]. There are 120 pilot and 720 data subcarriers that sum to 840 subcarriers, which are shown in Figure 2.7. Each column along the horizontal time axis contains the same 840 subcarriers. Since the 840 subcarriers repeat themselves in every horizontal time symbol column therefore, there will be 840 distinct subcarriers in whole Mobile WiMAX frame. Since this thesis aims at developing bandwidth scheduling algorithm for downlink, the focus will be on downlink. The duration of Mobile WiMAX frame is measured on horizontal time axis, typically in the units of number of OFDMA time symbols. Different duration of OFDMA frame is proposed in the standard such as 2, 2.5, 4, 5, 8, 10, 12.5 or 20 milliseconds (ms) [29, 30] but duration of 5 ms is accepted in IEEE 802.16e standard. Only duration of 5ms is accepted because it results in acceptable values of delay experienced by

delay-sensitive applications. Since Mobile WiMAX frame duration is 5 ms in the horizontal time axis, the total number of time symbols can be determined by determining the symbol duration of a time symbol.

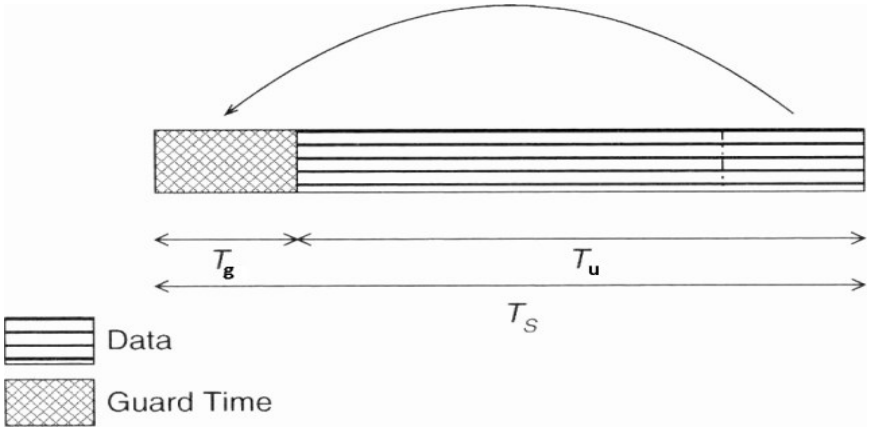


Figure 2.8: Basic Structure of OFDMA time symbol [29]

As shown in Figure 2.8, the basic structure of time symbol consists of Data and Guard Time. If the duration of Data time is T_u and duration of Guard time is T_g then the Total time symbol duration is given by T_s, where $T_s = T_u + T_g$. Total time symbol duration can be found out by making use of OFDMA time symbol primitive parameters defined by IEEE 802.16e standard [33]. The primitive parameters are Channel bandwidth, total number of subcarriers and sampling factor as shown in Table 2.1.

Parameter	**Description**	**Value**
BW	Channel Bandwidth	10 MHz
N_{total}	Total Subcarriers	1024
n	Sampling factor	28/25
G	Ratio T_g / T_u	1/8

Table 2.1: OFDMA Symbol Primitive Parameters

The sampling factor depends on the channel bandwidth and is set by IEEE 802.16e standard to 28/25 for the channel bandwidth of 10 MHz [33]. The sampling factor (n) is used in conjunction with the channel bandwidth (BW) and the total number of subcarriers (N_{total}) to determine the subcarrier spacing (Δf) and then time symbol duration of data, T_u.

Sampling frequency, F_s is calculated as:

$$F_s = n \times BW = \frac{28}{25} \times 10 = 11.2\ MHz \tag{2.1}$$

Within 10 MHz bandwidth, subcarrier spacing is calculated as:

$$\Delta f = \frac{F_s}{N_{used}} = \frac{11.2 \times 10^6}{1024} = 10.94\ KHz \tag{2.2}$$

Subcarrier spacing is the difference between the center frequencies of two subcarriers as shown in Figure 2.6.

Using this value of subcarrier spacing, duration of data time, T_u is determined as:

$$T_u = \frac{1}{\Delta f} = \frac{1}{10.94} = 91.4\ microseconds\ (\mu s) \tag{2.3}$$

The total symbol duration, T_s can now be calculated by using another primitive parameter, G which is the ratio between the duration of guard time, T_g and data time, T_u. The value of G which is a primitive parameter, has been set to 1/8 by IEEE 802.16e standard [29, 33]. The total symbol duration, T_s can then be computed as:

$$T_s = T_u + T_g = T_u + G \times T_u = 91.4 + \frac{1}{8} \times 91.4 = 102.8\ \mu s$$

Since, the frame duration is typically measured in units of number of OFDMA time symbols hence total number of time symbols can be calculated as:

$$S_{total} = \left\lceil \frac{Frame\ Duration}{T_s} \right\rceil = \left\lceil \frac{5\ msec}{102.8\ \mu s} \right\rceil = 49\ \ time\ symbols \tag{2.4}$$

It is now known that the whole Mobile WiMAX frame consists of 49 time symbols in the horizontal time axis. As shown in Figure 2.9, Mobile WiMAX frame consists of two parts: Downlink (DL) sub-frame and Uplink (UL) sub-frame. The DL sub-frame is used for transmitting data of users in the downlink direction and the UL sub-frame is used by users to transmit the data in uplink direction. There is a gap of two time symbols between DL and UL sub-frame, defined by IEEE 802.16e standard [33], therefore a fixed fraction of $(49 - 2) = 47$ time symbols are occupied by DL sub-frame as well as UL sub-frame. The gap of 2 time symbols is shown in Figure 2.9. Furthermore, IEEE 802.16e standard defines the ratio of 2:1 between the fraction of 47 time symbols occupied by DL and UL sub-frame, respectively. Therefore, DL and UL sub-frame consists of 29 and 18 time symbols, respectively as shown in Figure 2.9.

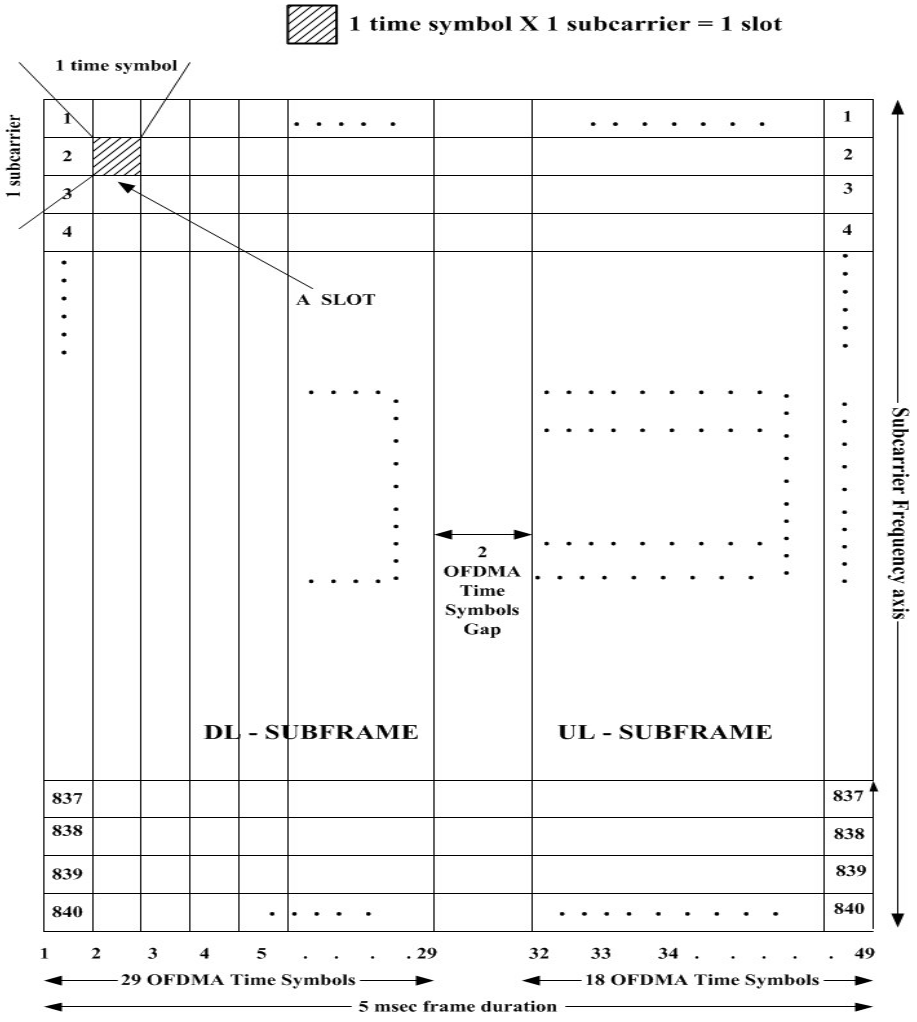


Figure 2.9: Mobile WiMAX frame showing DL and UL sub-frame

The most elementary unit, on which the data of a user can be transmitted in the downlink direction, is called a resource block. The whole area of DL sub-frame is occupied by fixed number of resource blocks. As, DL sub-frame consists of 29 time symbols in horizontal time axis and each of 29 time symbol columns consist of same 840 subcarriers in the vertical frequency axis, therefore the area of DL sub-frame will be $29 \times (1\ time\ symbol \times 840\ subcarriers)$, where $(1\ time\ symbol \times 840\ subcarriers)$ is the area of one time symbol column. The area of one time symbol column is composed of the most elementary unit $(1\ time\ symbol \times 1\ subcarrier)$ which is called a *slot* shown as shaded area in Figure 2.9. Recall that the 840 subcarriers repeat themselves in every horizontal time symbol column, hence the DL sub-frame still consists of 840 distinct sub-carriers, not $29 \times (1\ time\ symbol \times 840)$ distinct subcarriers. Unlike DL sub-frame, the area of a resource block can change according to the behaviour of users for whom the resource blocks are allocated [27, 29, 33, 35]. The users can behave in a mobile way i.e. their location is changing very fast with respect to the wireless base station or the users are fixed i.e. their location is fixed with respect to the base station. Since in this thesis, the users are assumed to be mobile, hence the focus will be on computing the area of a resource block for users when they are mobile.

According to the IEEE 802.16e standard [33], when the users are mobile then the area of one resource block occupies two time symbols on horizontal time axis with each time symbol occupying only the fraction of 840 slots in a time symbol column i.e. 28 slots [33, 35]. Each slot carries one subcarrier and on the total, 28 slots carry 28 subcarriers where the 28 subcarriers are collectively called a subchannel on a vertical frequency axis as shown in Figure 2.10. Out of 28 slots, 24 slots are allocated to data subcarriers and the remaining $(28 - 24) = 4$ slots are allocated to pilot subcarriers. Since a resource block occupies two time symbol columns, therefore 28 subcarriers will repeat themselves twice and get doubled in number, leading to 56 subcarriers. Out of 56 subcarriers, there will be $2 \times 24 = 48$ data subcarriers and $2 \times (28 - 24) = 8$ pilot subcarriers. In a nutshell, the area of a resource block is $2\ time\ symbols \times 1\ subchannel$ which consists of 56 slots where each slot consists of a subcarrier. Furthermore, out of 56 subcarriers, a resource block carries 48 data subcarriers and 8 pilot subcarriers [27, 29, 33, 35]. The area occupied by a resource block is shown in Figure 2.10.

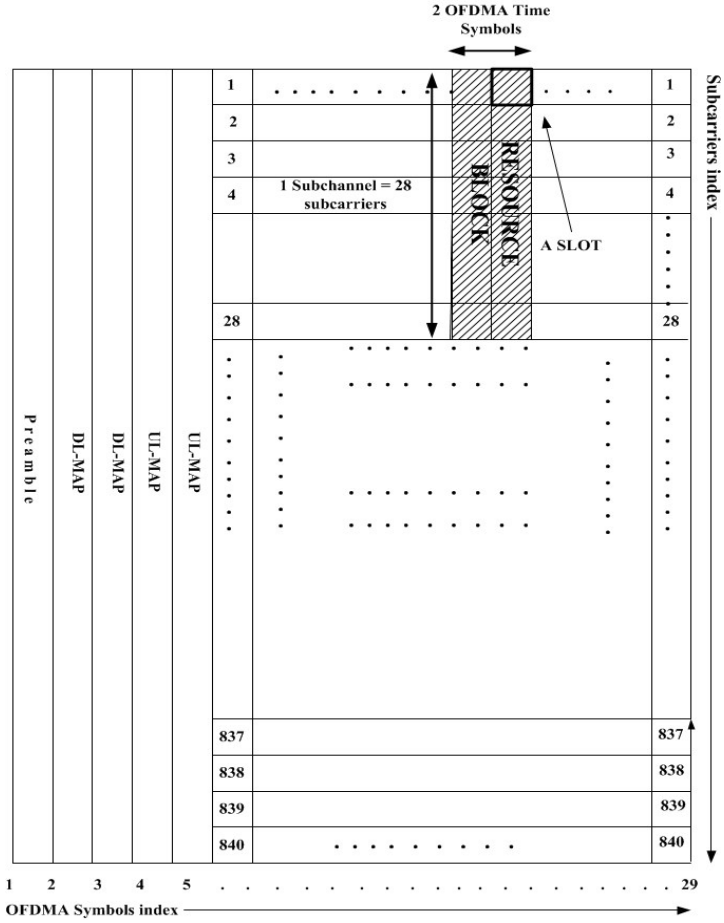


Figure 2.10: Structure of Resource Block in DL sub-frame

Since each subchannel in a time symbol column contains 28 subcarriers on the vertical frequency axis and one time symbol column contains 840 subcarriers, hence number of subchannels in a time symbol column, $N_{subchannels}$ can be computed as:

$$N_{subchannels} = \frac{Total\ (Data + Pilot)\ subcarriers}{Number\ of\ subcarriers\ in\ a\ subchannel} = \frac{840}{28} = 30$$

As shown in Figure 2.10, DL sub-frame also consists of several other components: Preamble, DL-MAP and UL-MAP. These three components carry the control information and they do not carry data of users. Since both DL-MAP and UL-MAP carry control information for users on the resource blocks that is why they also occupy two time symbol columns each, on horizontal time axis as shown in Figure 2.10. On the other hand, Preamble carries control information only on a single time symbol column as shown in

Figure 2.10. The remaining $29 - (2 + 2 + 1) = 24$ time symbol columns are available for transmitting the data of users in downlink. Therefore, the number of resource blocks available for transmitting the data of users in the downlink, N_{RB} is given by:

$$N_{RB} = \frac{Area\ occupied\ by\ 24\ time\ symbol\ columns}{Area\ occupied\ by\ one\ resource\ block} \tag{2.5}$$

$$N_{RB} = \frac{24 \times (1\ time\ symbol\ \times 840\ subcarriers)}{2\ time\ symbols\ \times 1\ subchannel}$$

Since, every time symbol column consists of same 840 subcarriers which are equal to 30 subchannels, therefore:

$$N_{RB} = \frac{24\ time\ symbols\ \times 30\ subchannels}{2\ time\ symbols \times 1\ subchannel} = 360$$

Hence, $N_{RB} = 360$ and there will be 360 resource blocks available for transmitting the data of users in downlink. *Furthermore, from now onwards, whenever we refer to OFDMA downlink sub-frame, a resource block will be considered as the most elementary building block of downlink sub-frame.*

Since our interest is in calculating the capacity of a resource block or the total amount of data which a resource block can carry, our focus will be only on data subcarriers in a resource block. A resource block consists of 48 data subcarriers. Data of several mobile stations is transmitted in OFDMA frame. Each mobile station's data is packed into one or more resource blocks, called burst [29, 35]. The number of resource blocks occupied by a burst of mobile station in a DL sub-frame duration depends upon the amount of data carried by a resource block and the amount of data scheduled for the mobile station to send. The amount of data to be sent for a mobile station in a frame duration is determined by the amount of bandwidth scheduled for the mobile station by bandwidth scheduling algorithm. On the other hand, the amount of data carried by a resource block depends on which modulation and coding scheme is used to transmit data of mobile station [29, 35]. The specific modulation and coding scheme is determined by another process used by Mobile WiMAX, called Link Adaptation.

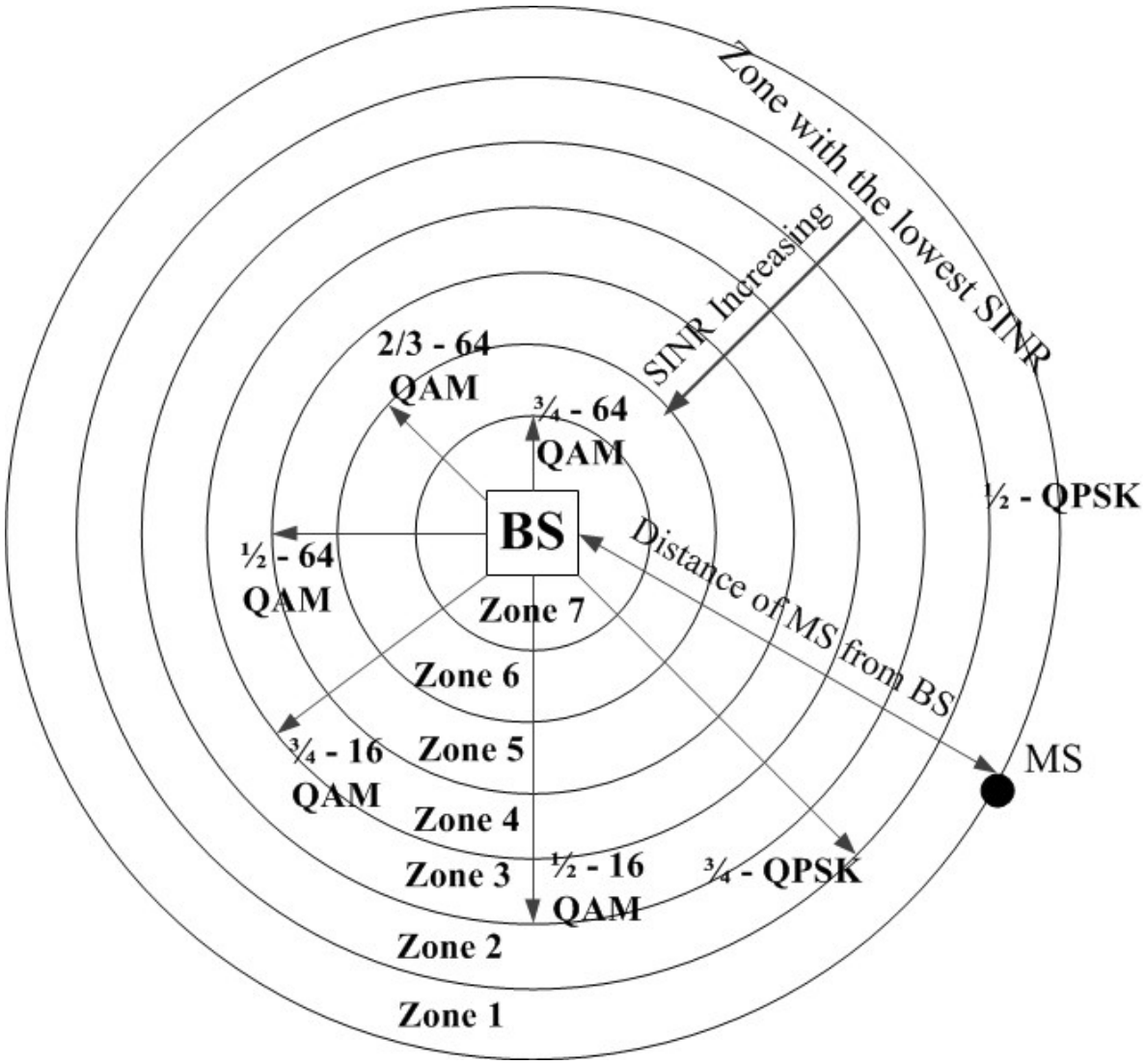


Figure 2.11: Classification of coverage area of a cell into different zones

Link adaptation is now used to determine the data rate of a resource block. It is known that a resource block contains 48 data subcarriers. A scenario as shown in Figure 2.11 can be considered in Mobile WiMAX, where the coverage area of a cell is divided into 7 zones. Each zone represents a range of distance of mobile station from the base station, which represents the channel condition in terms of range of SINR of signal received by the user. It is shown in Figure 2.11 that as the distance of mobile station from base station increases, the SINR experienced by the mobile station decreases. It further means that channel condition deteriorates, thus resulting in Mobile WiMAX selecting a modulation

and coding scheme with relatively less number of bits per data symbols [26, 35, 29]. With an example and the background built on digital modulation, variation in data rate of a resource block packed for a user in OFDMA DL sub-frame can be verified. Let us take an example when channel is in worst condition, having Mobile Station (MS) experienced SINR of 2-5 dB (this range of SINR is considered to be threshold of worst channel conditions in Mobile WiMAX environment). Hence modulation and coding scheme used here will be 1/2-QPSK. As can be observed that it is a $2^K - ary$ scheme where $K = 2$ bits per data symbol, 2 bits of data can be transmitted from source to destination on a single subcarrier. Furthermore, channel coding rate of 1/2 is used on the data symbol of 2 bits. Therefore, total bits of data carried by output codeword, is given by:

$$L_{data} = R_{coding} \times K = \frac{1}{2} \times 2 = 1\ bit$$

Where L_{data} is the number of data bits carried by output codeword after coding, R_{coding} is the channel coding rate and K is the number of bits per data symbol before coding. Since each resource block contains 48 data subcarriers, from the background built on digital modulation it is known that after coding, 1 data bit is used to phase modulate a single subcarrier. Hence, one subcarrier is carrying 1 bit of data, if $b_{z,RB}$ is the variable number of data bits carried by a resource block for an MS in zone number z in a frame duration, then the total data bits carried by 48 data subcarriers of a resource block are:

$$b_{1,RB} = 48\ data\ subcarriers \times L_{data} = 48 \times 1 = 48\ bits$$

where z = 1 in this case, as the MS lies in zone number 1, as shown in Figure 2.11. Since, the definition of a resource block is 2 $time\ symbols \times 1\ subchannel$, hence the duration of a resource block on horizontal time symbol axis will be twice the duration of elementary time symbol. Hence, if $r_{z,RB}$ is the data rate of a resource block for an MS in zone number z and T_s is the time duration of time symbol on horizontal time symbol axis, then the data rate of a resource block for an MS in zone number z, $r_{z,RB}$ can be computed as:

$$r_{z,RB} = \frac{Data\ bits\ carried\ by\ a\ resource\ block}{Duration\ of\ a\ resource\ block\ on\ time\ symbol\ axis} = \frac{b_{z,RB}}{2 \times T_s} = \frac{48\ bits}{205.6\ \mu s}$$

$$r_{z,RB} = 233.46\ kbps$$

In a similar way, data rate of a resource block for varying distances or varying zone numbers of mobile stations from base station can be determined, and these results are depicted in Table 2.2, which shows the varying data rates of a resource block for a mobile station according to location of a mobile station in a specific zone number. Depending upon the amount of data scheduled for a specific mobile station and data rate of a resource block, the number of resource blocks occupied by the mobile station is calculated. Resource blocks for the mobile station are packed to form a burst [29, 35]. Many mobile stations' data is sent together in one OFDMA frame, packed separately into bursts. Since every mobile station can be at different location with respect to the base station, a burst for a specific mobile station has different data rate than other mobile stations, and also has different number of resource blocks packed into its respective burst.

Zone Number (z)	**SNR (dB)**	**Modulation and Coding Scheme**	**Input data bits per subcarrier**	**Output data bits per subcarrier**	**Bits per resource block ($b_{z,RB}$)**	**Data Rate of a resource block ($r_{z,RB}$ kbps)**
1	2 - 5	½ QPSK	2	1	48	233.46
2	5 - 8	¾ QPSK	2	1.5	72	350.20
3	8 - 10	½ 16-QAM	4	2	96	466.92
4	10 - 14	¾ 16-QAM	4	3	144	700.40
5	14 - 16	½ 64-QAM	6	3	144	700.40
6	16 - 18	2/3 64-QAM	6	4	192	933.85
7	18 - 20	¾ 64-QAM	6	4.5	216	1050.58

Table 2.2: Data rate of a resource block for different modulation and coding schemes

The location of mobile stations with respect to the base station can be used to determine the time varying total capacity of wireless channel in downlink in a cell coverage area. We will now determine the time varying total capacity of wireless channel in downlink according to location of mobile stations, using a part of computations provided in [42]. According to Table 2.2, the coverage area of base station is divided into seven zones. The lower zone number represents the use of lower modulation and coding scheme, which indicates longer distances between a mobile station and base station, as shown in Figure 2.11. Since in Mobile WiMAX, mobile users are assumed to be continuously moving, the number of mobile stations present in each zone may change in every scheduling round.

If N is the total number of mobile stations present in a cell coverage area at all times and MS_z is the number of mobile stations out of total N mobile stations, present in zone number z, then $\sum_{z=1}^{7} MS_z = N$. If U_z is the fraction of N users, which are present in a zone number z in a scheduling round, then U_z is given by:

$$U_z = \frac{MS_z}{\sum_{z=1}^{7} MS_z} \tag{2.6}$$

Let $N_{z,RB}$ denote the total number of resource blocks allocated for all the users present in a zone number z, out of total N_{RB} resource blocks. The total number of resource blocks, N_{RB} is computed from equation (2.5). Therefore, $N_{z,RB}$ is given by [42]:

$$N_{z,RB} = U_z \times N_{RB} \tag{2.7}$$

such that $\sum_{z=1}^{7} N_{z,RB} = N_{RB}$. All the $N_{z,RB}$ resource blocks will use the same modulation and coding scheme where each resource block carries $b_{z,RB}$ bits of data, as can be observed from Table 2.2. Therefore, if C represents the total bytes of data carried by total N_{RB} resource blocks present in a DL sub-frame and b_{RB} denotes the number of data bits carried by a resource block, then C is given by:

$$C = \frac{b_{RB} \times N_{RB}}{8}$$

It is known that $\sum_{z=1}^{7} N_{z,RB} = N_{RB}$, therefore [42]:

$$C = \frac{b_{RB} \times \sum_{z=1}^{7} N_{z,RB}}{8}$$

Since, each of the $N_{z,RB}$ resource blocks allocated for users in zone number z, will carry $b_{z,RB}$ bits of data, therefore for a specific zone number $b_{RB} = b_{z,RB}$. This results in:

$$C = \frac{\sum_{z=1}^{7} (b_{z,RB} \times N_{z,RB})}{8} \tag{2.8}$$

where C is also called the capacity of Mobile WiMAX DL sub-frame in bytes in a single scheduling round [42]. Note that this capacity can vary in every scheduling round if the number of mobile stations present in a zone number z, MS_z changes in every scheduling

round. If MS_z changes in a scheduling round, then according to equation (2.6), fraction of N users, present in a zone number z, U_z will change. Furthermore, according to equation (2.7), this will lead to change in total number of resource blocks allocated for users present in zone number z, $N_{z,RB}$. Since, $N_{z,RB}$ changes in a scheduling round, therefore according to equation (2.8) capacity of Mobile WiMAX DL sub-frame will also change in a scheduling round.

Mobile WiMAX is able to achieve maximum capacity, when all the mobile stations present in a cell coverage area are in zone 7 or nearest to the base station. In that case, according to Table 2.2, all the 360 Resource Blocks (RBs) will carry, $b_{z,RB} = 216\ bits$. Therefore, using equation (2.8), maximum capacity of Mobile WiMAX for a cell coverage area in downlink is given as:

$$C = \frac{b_{7,RB} \times 360}{8} = \frac{77760\ bits}{8} = 9720\ bytes$$

And the maximum bandwidth in downlink of a cell coverage area is given by:

$$Maximum\ bandwidth = \frac{Maximum\ capacity}{Duration\ of\ total\ time\ symbols\ occupied\ by\ 360\ RBs}$$

$$Maximum\ bandwidth = \frac{Maximum\ capacity}{Duration\ of\ 24\ time\ symbols} = \frac{77760\ bits}{24 \times T_S}$$

$$Maximum\ bandwidth = \frac{77760\ bits}{24 \times 102.8\ \mu s} \approx 31\ Mbps$$

There are total of 360 resource blocks in resource matrix of DL sub-frame and every user's data is packed into bursts, however there is no clear boundary between the bursts of different users, resulting in a very interesting question. How do mobile stations become aware of the location of their specific burst in resource matrix of DL sub-frame? The answer to this question lies in the structure of DL-MAP. Hence, only DL-MAP will be discussed in this thesis. A DL-MAP message for a user is called DL-MAP IE and stores information about location of each user's burst in a DL sub-frame, in the form of location coordinates. Table 2.3 shows the contents of each DL-MAP IE message. As shown in Table 2.3, every DL-MAP IE message stores the location coordinates of a burst of user in

resource matrix, in the form of OFDMA time symbol offset, number of OFDMA time symbols, the subchannel offset and the number of subchannels [6, 29].

DL-MAP IE Parameters	**Use for the burst**
OFDMA symbol offset	Offset of OFDMA symbol in which burst starts, measured in OFDMA symbols from the beginning of DL sub-frame in which DL-MAP is transmitted
Subchannel offset	The lowest index of OFDMA subchannel used for carrying the burst, starting from subchannel# 1
Number of subchannels	The number of subchannels with subsequent indexes used to carry the burst
Number of OFDMA symbols	The number of OFDMA symbols with subsequent indexes used to carry the burst
Connection Identifier (CID)	Address of the receiver

Table 2.3: Major parameters of DL-MAP IE [6,29]

It is known that every kind of information in DL sub-frame is composed of groups of resource blocks except preamble which occupies one time symbol column on the horizontal time axis. This makes to remaining $29 - 1 = 28$ time symbols. Since the area occupied by one resource block on DL sub-frame containing 28 time symbol columns is $2\ time\ symbols \times 1\ subchannel$, therefore the 28 time symbols are grouped into groups of two time symbols on the horizontal time axis. This results in $\frac{28}{2} = 14$ such groups forming on the horizontal time axis. Similarly, for each 28 time symbol columns, the 840 subcarriers are grouped into groups of 28 subcarriers because each subchannel contains 28 subcarriers. The resultant resource matrix of DL sub-frame, after groupings on horizontal as well as vertical axis is shown in Figure 2.12, where the most elementary unit of radio resource will be a resource block. Figure 2.12 also shows the content of DL-MAP message of each user and how this contributes to an increase in the overall content of DL-MAP. This means that higher the number of users, the higher is the number of DL-MAP messages.

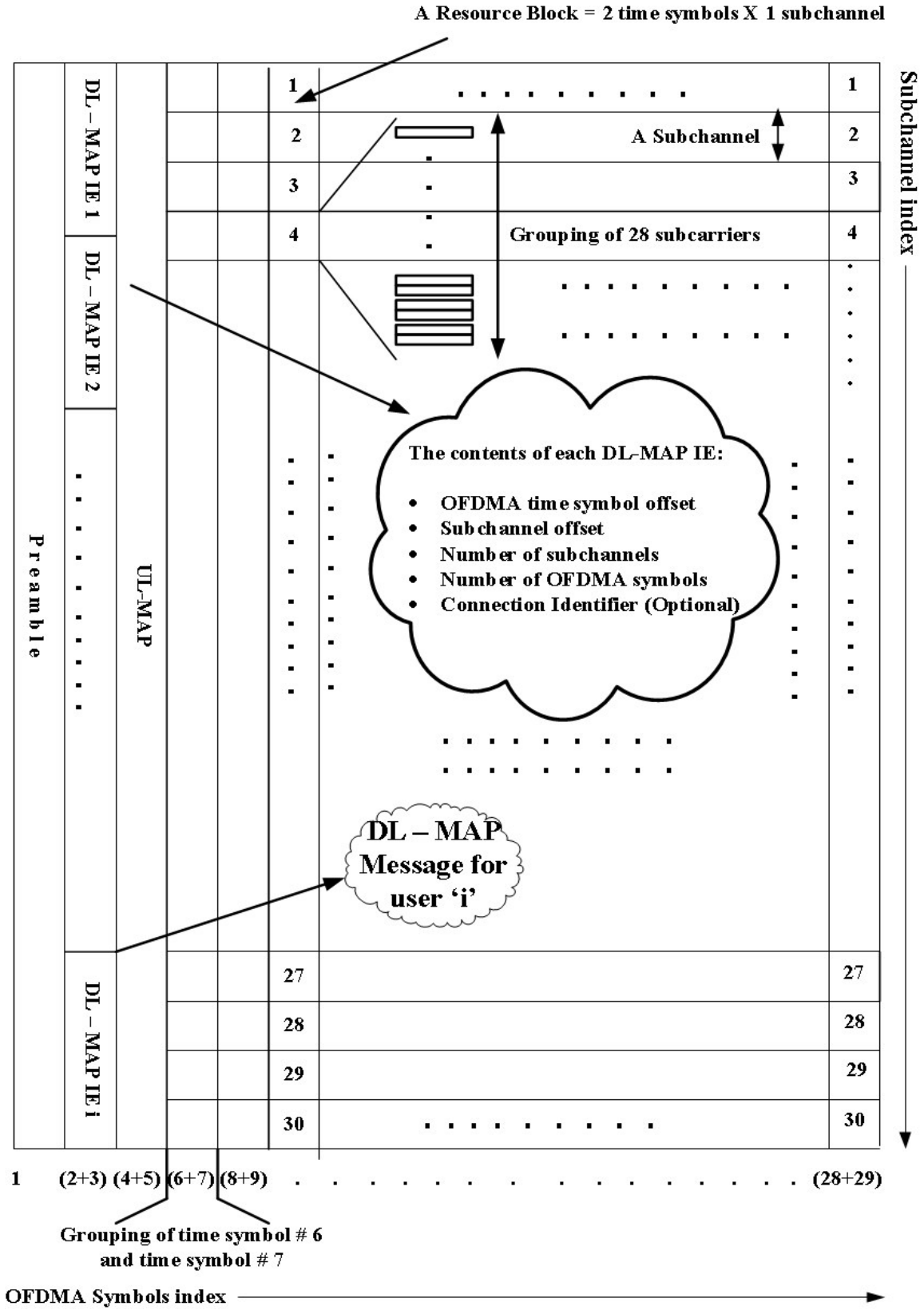


Figure 2.12: Composition of resource blocks

Figure 2.13 shows an example of two different users' data packed into bursts. *User #1 burst* is the *user# 1* data packed into a burst and *User # 4 burst* is the *user# 4* data packed into a burst. Each burst consists of a certain number of resource blocks and uses a specific modulation and coding scheme out of the seven choices [29, 35]. There are two important parameters involved in computing the wake up times of different users receiving their bursts. The first parameter is *burst delay*. Burst delay is the duration in time measured in terms of number of time symbols from the start of DL-MAP, for which the user has to wait until its burst arrives. The second parameter is *burst duration* which is defined as the number of time symbols occupied by the burst of a user on the horizontal time axis. Both parameters are shown in Figure 2.13. The OFDMA frame starts from DL-MAP and every mobile station wakes up at the start of DL-MAP and wait for its burst to arrive [33, 36]. Every burst contains an address of the mobile station, to which that burst is addressed. That address is called Connection Identifier (CID). Once the mobile station completely receives the burst addressed to it, then it again goes to sleep [33, 36]. Information about the location of the *User burst# 1* and *User burst# 4* in DL sub-frame is stored in their respective DL-MAP IEs. In such a scenario, the wake up time of mobile station will be the sum of burst delay and burst duration as shown in Figure 2.13. The majority of the devices present in cell coverage area are battery powered and undergo power consumption for the period of wakeup time [36]. Hence, less wakeup time means less power consumption. In order to decrease the power consumption of mobile stations, there is a privilege to include address of mobile station (CID) in their respective DL-MAP IEs. Therefore, if the CID of each mobile station is included in their respective DL- MAP IEs, then mobile station will only wake up for the period of burst duration, as shown in Figure 2.13 [36]. Note that including CID of each user in their respective DL-MAP IEs further increases the overall content of DL-MAP.

The more users' bursts are present in resource matrix, the higher the number of DL-MAP IEs will be present in DL-MAP. Therefore, DL-MAP can grow more than its usual size, which is $2\ time\ symbols \times 30\ subchannels$ as shown in Figure 2.13 [3, 5, 29, 35]. DL-MAP grows in the direction of increasing time symbol index and subchannel index as shown in Figure 2.13. The number of users having bursts allocated in resource matrix changes in every scheduling round. This results in the content of DL-MAP to decrease or

increase from its usual size. Therefore, it is always recommended to pack users' data into bursts from the end of DL sub-frame as opposed to the beginning of DL sub-frame, i.e., from *time symbol # (28+29)* to *time symbol# (6+7),* as shown in Figure 2.13. It will give more area for DL-MAP to grow if the number of users increases in resource matrix than the usual number.

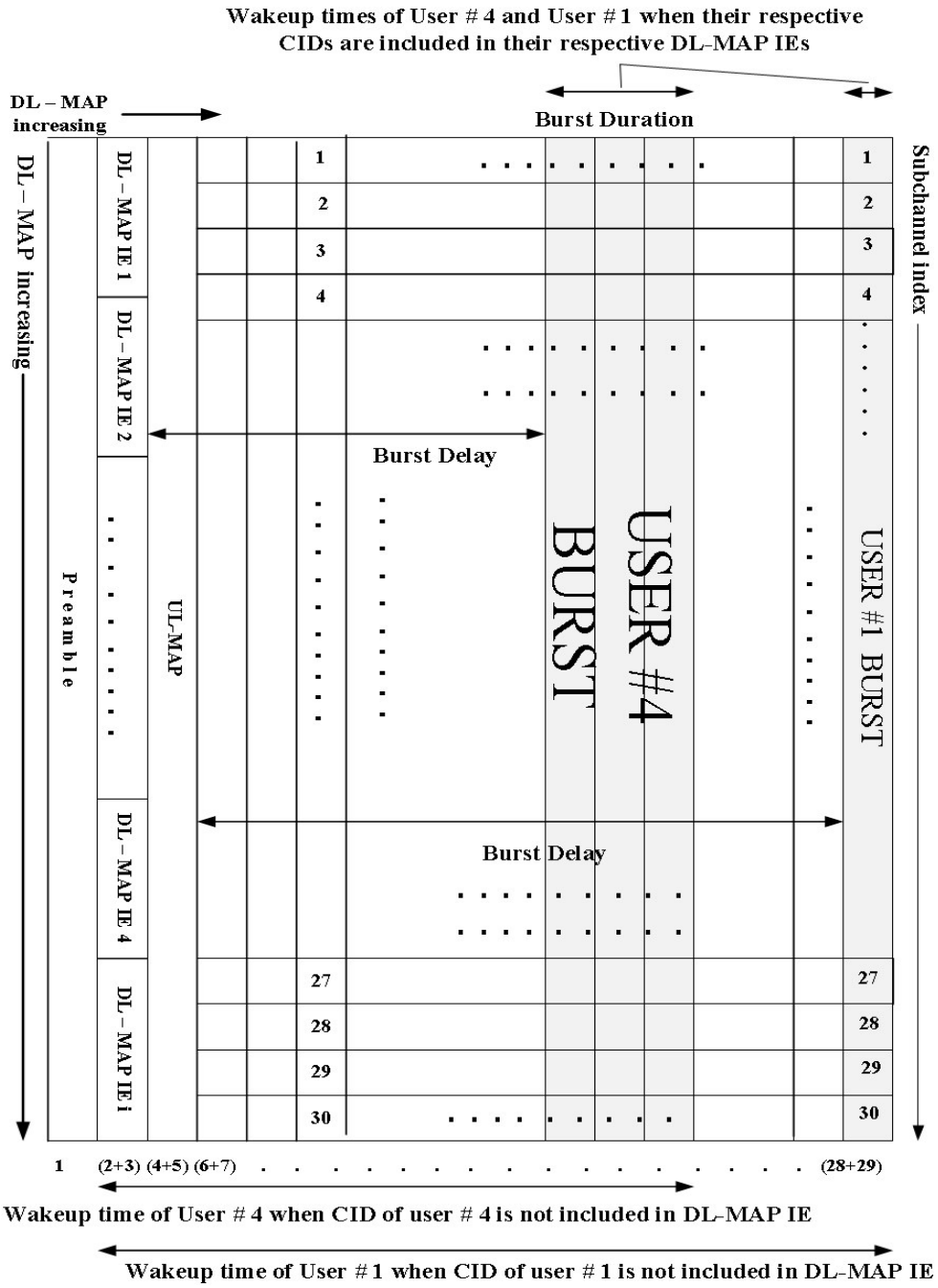


Figure 2.13: Depiction of Wake up times of users in two different cases

It is concluded that power consumption of any battery powered device depends specifically on how bursts are packed. The number of users keeps on changing in every scheduling round, which makes size of DL-MAP unpredictable with respect to its usual size. Therefore, packing from the end of frame is recommended, providing more space for DL-MAP to grow.

Including Connection Identifier (CID) of mobile stations in their respective DL-MAP IEs further contributes in the increase of overall content of DL-MAP, which increases the amount of control information in a DL sub-frame, and decreases the amount of resource blocks utilized to map users' data, hence system throughput also decreases. Due to this reason, the focus of this thesis will be on gaining fairness in average wakeup time for different users when CID is not included in their respective DL-MAP IEs.

2.5. Medium Access Control (MAC) Layer of IEEE 802.16e Mobile WiMAX

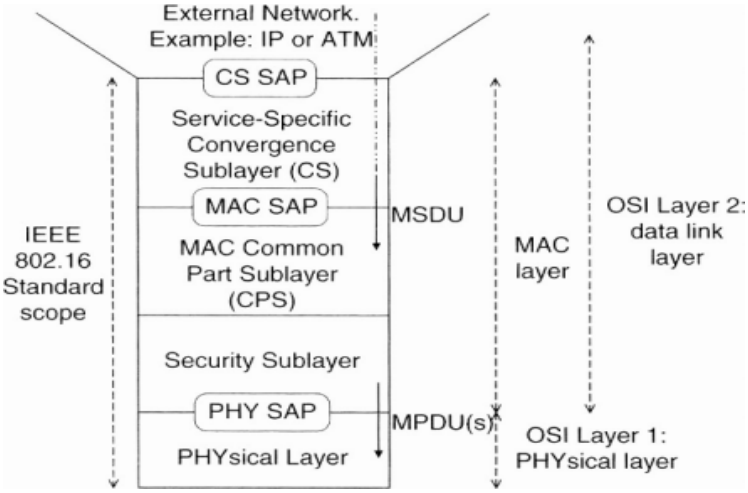


Figure 2.14: MAC Layer defined by IEEE 802.16e standard [29]

Figure 2.14 shows the layered protocol architecture of IEEE 802.16e standard. The MAC layer is divided into three sub-layers:

- Convergence Sub-layer (CS)
- Common Part Sub-layer (CPS)
- Security Sub-layer

As the core of MAC layer lies in CPS [28, 29, 35], hence only CPS will be discussed in this thesis. CPS performs many procedures which are of prime interest. It performs tasks like frame construction, multiple access, scheduling, radio resource management, QoS management, etc [28, 29]. The ones that are needed to be introduced in this thesis are: scheduling and QoS management. In fact, these two are considered to be the key functions of CPS [28, 29, 35].

Quality of Service (QoS) management is the efficient use of limited system bandwidth in satisfying the requirements of various users using different kind of applications. These applications can vary from Web Browsing, email, Mobile TV, Live Streaming, Voice over Internet Protocol (VoIP), etc. The QoS requirements of these applications are specified in terms of packet delay, packet delay variation, and packet loss rate [29, 35].

Common Part Sub-layer (CPS) defines five different kinds of scheduling services for different kind of applications. These applications are typically described in terms of their data rates and burstiness. Data rates are defined in terms of three parameters: Minimum Reserved Traffic Rate (MRTR), Sustained Traffic Rate (STR) and Maximum Sustained Traffic Rate (MSTR) [28, 29, 35].

- **Maximum Sustained Traffic Rate (MSTR)**

This parameter defines the peak rate of a Class of Service (CoS). Every CoS should be policed to conform to this parameter, on average, over time [33]. This parameter also acts as a maximum threshold for limiting the CoS whenever it tries to transmit at higher rate [28].

- **Minimum Reserved Traffic Rate (MRTR)**

This parameter specifies the minimum rate reserved for a CoS. It is the minimum rate at which a CoS should transmit data when averaged over time [33]. It is also the guaranteed amount of bandwidth which a flow is allocated when averaged over time [28].

- **Sustained Traffic Rate (STR)**

This parameter specifies the rate between MRTR and MSTR which a CoS should maintain when averaged over time. It is also called average rate of CoS [39].

The five different kinds of scheduling services for different kind of applications, defined by IEEE 802.16e standard, are:

- **Unsolicited Grant Service (UGS)**

This service is also called Constant Bit Rate (CBR) service. This service emulates the same effect as circuit switching. Even if no data is available for transmission, applications using UGS always receive their fixed share of bandwidth and always transmit data in fixed size packets [28, 29, 35]. As it is a CBR service, irrespective of amount of data waiting for transmission, a fixed amount of bandwidth will be allocated to these applications [28, 29, 35]. For such kind of services, MRTR will be equal to MSTR. This service supports applications like VoIP with no silence suppression [35].

- **Real Time Polling Service (rtPS)**

This service is also called Real Time Variable Bit Rate (rtVBR) service. It supports real time delay sensitive applications [29, 35]. Bandwidth allocated to applications based on rtPS can vary depending upon the amount of data available for transmission. Allocated bandwidth to rtPS connection is bounded by MRTR and MSTR. Applications based on rtPS transmit data in the form of variable size packets in variable sized bursts [35]. These applications have some packet delay requirements to be met by the network [29, 35]. rtPS is designed to support delay-sensitive and variable bit rate applications such as Mobile TV, Live Streaming, and Video on Demand (VoD) [28, 35].

- **Extended Real Time Polling Service (ertPS)**

This is a service class that is built on the efficiency of both UGS and rtPS [35]. It is similar to UGS in a way that even if no data is available for transmission, applications using ertPS always receive bandwidth [28, 29, 35]. Whereas, it is similar to rtPS in a way

that applications using ertPS receive variable bandwidth grants as opposed to fixed bandwidth grants in the case of UGS [28, 29, 35]. This class of service is suitable for variable rate real time applications which have delay and data rate requirements. VoIP with silence suppression is an example of application using such kind of service [29].

- **Non Real Time Polling Service (nrtPS)**

This type of service is also called Non Real Time Variable Bit Rate (nrtVBR). Applications using nrtPS are delay-insensitive and hence do not require any packet delay guarantees [29]. But such service requires that MRTR requirements need to be met always [28, 29]. Hence, applications based on such kind of service should always be granted minimum reserved bandwidth. Applications transmit data in the form of variable size packets in variable bursts, just like rtPS [28, 29]. Such service support applications such as File Transfer Protocol (FTP).

- **Best Effort (BE) Service**

Best Effort requires no guarantees in terms of bandwidth or delay [29]. Bandwidth is usually distributed among UGS, rtPS, ertPS and nrtPS first, and if any bandwidth is left, then it is allocated to applications based on Best Effort service. Network puts its best effort to transmit the data waiting for transmission for the applications using Best Effort service [29]. This is the reason, why such service is called Best Effort. Such service supports applications like email, etc [28].

Chapter 3:

Literature Review

3.1. Introduction

A bandwidth scheduling algorithm in any communication network makes a decision to allocate a proper share of link capacity to all users. The link can be either wired or wireless. A proper share of capacity to every user ensures their negotiated QoS. The time interval during which an algorithm makes this decision is very short and constant. It is called scheduling or allocation interval. A good scheduling algorithm also ensures fair allocation of bandwidth, where fairness criteria can vary depending upon the objective to be achieved, which can include throughput guarantees, bounded delays, or both.

To better understand the evolution of scheduling algorithms for wireless networks, some major scheduling algorithms in wired networks are discussed briefly first. Their discussion will help define the notion of fairness in context of bandwidth scheduling. The algorithms to be discussed are: Generalized Processor Sharing (GPS) [7], Packet by packet GPS (PGPS) [8], Worst case Fair Weighted Fair Queuing (WF^2Q) [9], and Self Clocked Fair Queuing (SCFQ) [10]. The transition is then made from wired networks to major 2G networks scheduling algorithms. It will be observed during the discussion of 2G networks scheduling algorithms that algorithms in this category followed the approach of adapting wired networks scheduling algorithms. The major scheduling algorithms developed for 2G networks to be discussed are: Channel State Dependent Packet Scheduling (CSDPS) [12], Idealized Wireless Fair Queuing (IWFQ) [14], and Channel condition Independent Fair Queuing (CIF–Q) [15].

Further advancements in 2G wireless networks led to the development of 3G wireless networks. The discussion of CIF–Q will conclude that the approach of adapting wired networks scheduling algorithms is not feasible for 3G networks [15] because unlike wired networks, wireless networks experience location-dependent channel errors combined with the fact that 3G networks also support data along with voice [15]. It will be further concluded that there is a need for novel approaches to 3G networks and the idea of fairness for wireless networks needs to be re-examined [15]. CIF-Q has indeed introduced two different types of fairness in wireless networks: Short-term fairness and Long-term fairness [15]. The two kinds of fairness will be defined now.

- **Short-term fairness**

A wireless scheduling discipline is said to have a property of short-term fairness if the maximum disparity between the services received by two flows (in bits) in a scheduling round of a discipline, is bounded [10, 15, 37].

- **Long-term fairness**

A wireless scheduling discipline is said to have a property of long-term fairness if every flow serviced in a discipline, has received its fair share or negotiated average bandwidth, when averaged over time [13, 14, 15].

The notion behind both types of fairness can be explained with a simple scenario, where two users are using different services, namely web browsing and VoIP. It is known that web browsing is a Best Effort (BE) service and hence requires no guarantees whereas VoIP is a Real Time Polling Service (rtPS) and hence requires delay as well as bandwidth guarantees. Suppose that both users receive services in the same cell coverage area over the wireless channel, however in a scheduling round, web browsing user experiences high SINR, whereas VoIP user experiences low SINR. If a scheduling algorithm developed to make efficient use of wireless channel, is used in such a situation then it defers the packet transmission of VoIP user because VoIP user is experiencing low SINR. Since, VoIP user is experiencing low SINR, therefore transmitting the data of VoIP user will result in inefficient use of wireless channel. Hence, VoIP user receives no service at all whereas the

web browsing user receives all the service surrendered by VoIP user. If such a situation prevails for a time duration longer than a scheduling round then it leads to degradation in the QoS received by VoIP user because maximum disparity between the service received by two users was not bounded in a scheduling round. This resulted in unbounded maximum service disparity which increased in every scheduling round in a time duration. Although, when the VoIP user returns to experiencing high SINR, then the scheduling discipline compensates the VoIP user with the bandwidth that it has lost during all the past opportunities to transmit. Therefore on average, over time, scheduling discipline is long-term fair to the users in terms of their Sustained Traffic Rate (STR).

On the other hand, if a scheduling algorithm developed to keep the maximum service disparity between two users, bounded in a scheduling round, is used then it will not defer the transmission of VoIP user in the condition of low experienced SINR. This results in maximum service disparity between two users, being tightly bounded and therefore, VoIP user receives good QoS. Hence, a scheduling discipline is short-term fair to the users in terms of delay and bandwidth guarantees.

Scheduling algorithms developed for 3G networks lie in the first category i.e. they are developed to make efficient use of wireless channel, as poor degree of short-term fairness is achieved in 3G networks. This is because even if the packet transmission of a user experiencing low SINR, is not deferred, for most values of SINR, it will be an erroneous transmission, as 3G systems use limited modulation and coding schemes. Since, 3G systems use limited modulation and coding schemes, therefore they cannot make the channel appear as good for any value of SINR, like 4G systems. Hence, not deferring the packet transmission of a user experiencing low SINR, will lead to inefficient use of wireless channel, therefore it is better to defer the packet transmissions.

Since wired networks do not experience location-dependent channel errors, a scheduling discipline being short-term fair implies being long-term fair and vice-versa. This is the reason why there are not two different types of fairness in scheduling algorithms for wired networks, it is simply called fairness. The degree of fairness provided by wired networks scheduling algorithms is measured by absolute fairness index. Absolute fairness index measures the degree of fairness provided by wired network scheduling algorithm, by

computing disparity between the service received by a flow in a discipline and the service received by a same flow in a discipline which is used as a benchmark [7, 8, 11, 37]. Absolute fairness index is used in some other form in wireless networks to measure the degree of short-term fairness and it is called relative fairness index [11]. Relative fairness index measures the degree of short-term fairness provided by wireless network scheduling discipline, by computing disparity between the service received by two flows in the same discipline [10, 11, 37]. Long term fairness is usually measured in terms of STR requirements of different users or cell throughput. The fairness of scheduling algorithms developed for 3G and 4G wireless networks is always measured in terms of both short-term as well as long-term fairness. After the discussion on 2G networks scheduling algorithms, two major 3G networks scheduling algorithms namely Server Based Fairness Approach (SBFA) [1] and Token Bank Fair Queuing (TBFQ) [19] will be discussed. The main purpose behind discussing the two algorithms is twofold:

Firstly, the discussion of SBFA will depict how the performance of a wireless scheduling discipline improves if a discipline takes the enhancements in the physical layer of a wireless system into consideration [1].

Secondly, the discussion of TBFQ will depict how relative fairness index measures the degree of disparity between the service received by two flows and how the degree of disparity of service, measures the short-term fairness [19]. Another algorithm for 3G+ wireless networks, called Channel State independent Wireless Fair Queuing (CS-WFQ) [4], will be also discussed. The discussion of CS-WFQ will act as an example of a scheduling algorithm developed for wireless networks using variable modulation and coding schemes. Further evolution of 3G wireless networks led to the development of 4G wireless networks. The discussion of 3G wireless networks scheduling algorithms will conclude that wireless scheduling disciplines adopting similar approaches as SBFA or TBFQ will not be suitable for 4G networks. Since, such approaches are not suitable for 4G wireless networks, they need improvisation. The discussion of CS-WFQ provides basic idea behind improvisations of scheduling algorithms for 4G networks.

Then two major 4G networks scheduling algorithms, namely Multi Rate Fair Queuing (MRFQ) [2] and Adaptive Token Bank Fair Queuing (ATBFQ) [41], will be discussed. The main purpose behind discussing the two algorithms is to know how to develop an algorithm that considers all important enhancements in the physical layer of 4G networks. The discussion on 4G networks scheduling algorithms will conclude that MRFQ and ATBFQ are developed under the assumption that the wireless channel capacity is not variable in every scheduling round. It means that location of users in a cell coverage area, is considered to be fixed or changing slowly with respect to wireless base station but in reality the users are highly mobile and hence, moving within a cell coverage area with high speeds. Therefore, MRFQ and ATBFQ cannot be declared as complete scheduling solution for 4G wireless systems like Mobile WiMAX, that are developed to provide broadband wireless access in true sense.

As shown in Figure 3.1, burst construction mechanism in Mobile WiMAX systems accepts the scheduled data for different users from the scheduler. The burst construction mechanism packs the scheduled data in the form of bursts and maps those bursts to the OFDMA downlink sub-frame. Different burst construction algorithms map bursts considering one of the several constraints. The major constraints are, maximizing cell throughput, QoS of different users, average reduction in wastage of resource blocks, and minimization of average power consumption (average wake up times) of mobile stations. Following the section on scheduling algorithms, some existing major burst construction algorithms [3, 6, 20, 21] will be discussed that focus on mapping bursts under a constraint or set of constraints. Finally, the whole discussion on scheduling as well as existing burst construction algorithms will be summarized and some interesting conclusions will be drawn about scheduling as well as burst construction algorithms.

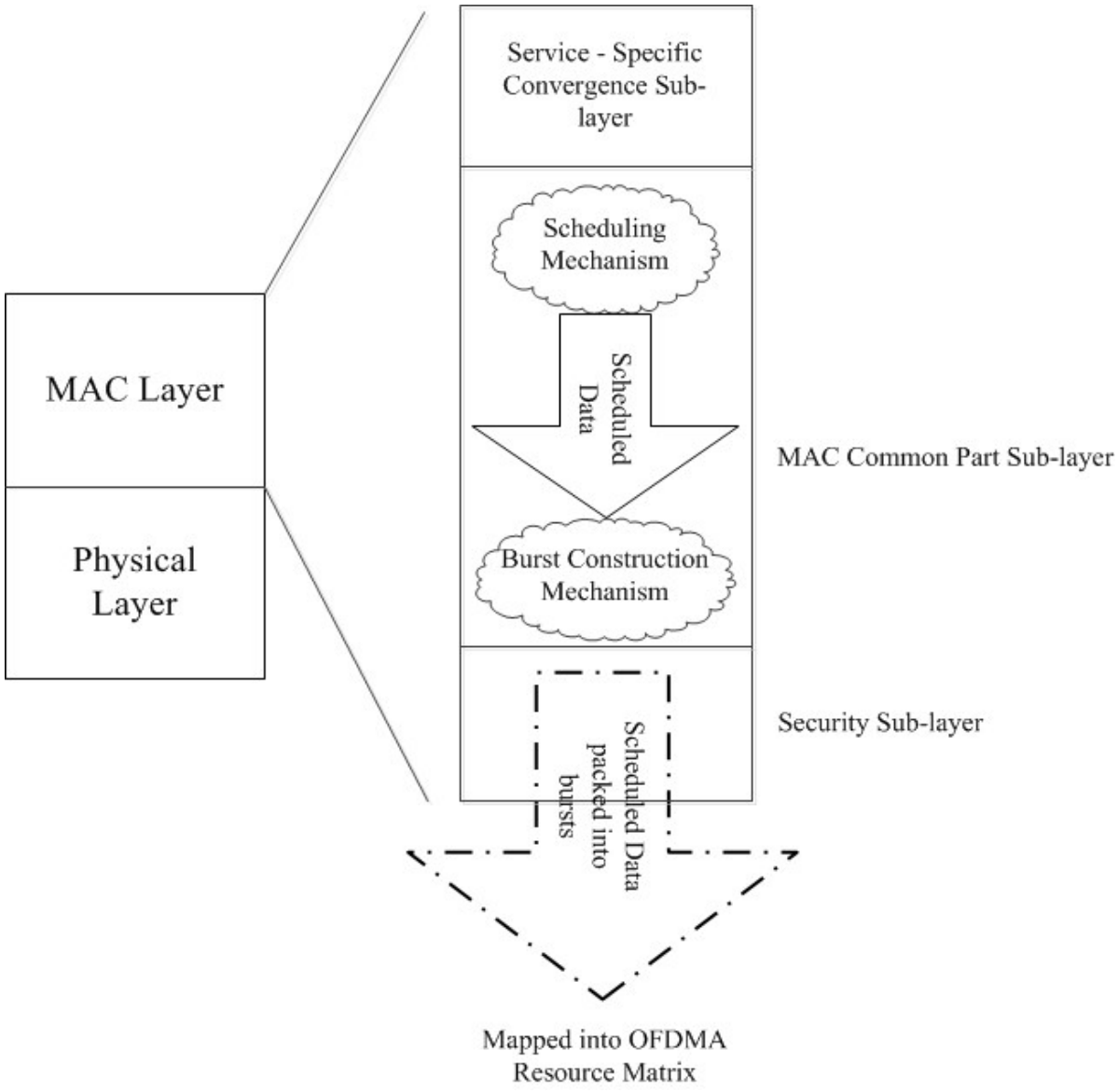


Figure 3.1: Relationship between Scheduler and Burst Construction Mechanism

3.2. Wired Network Scheduling Algorithms

3.2.1. Generalized Processor Sharing (GPS)

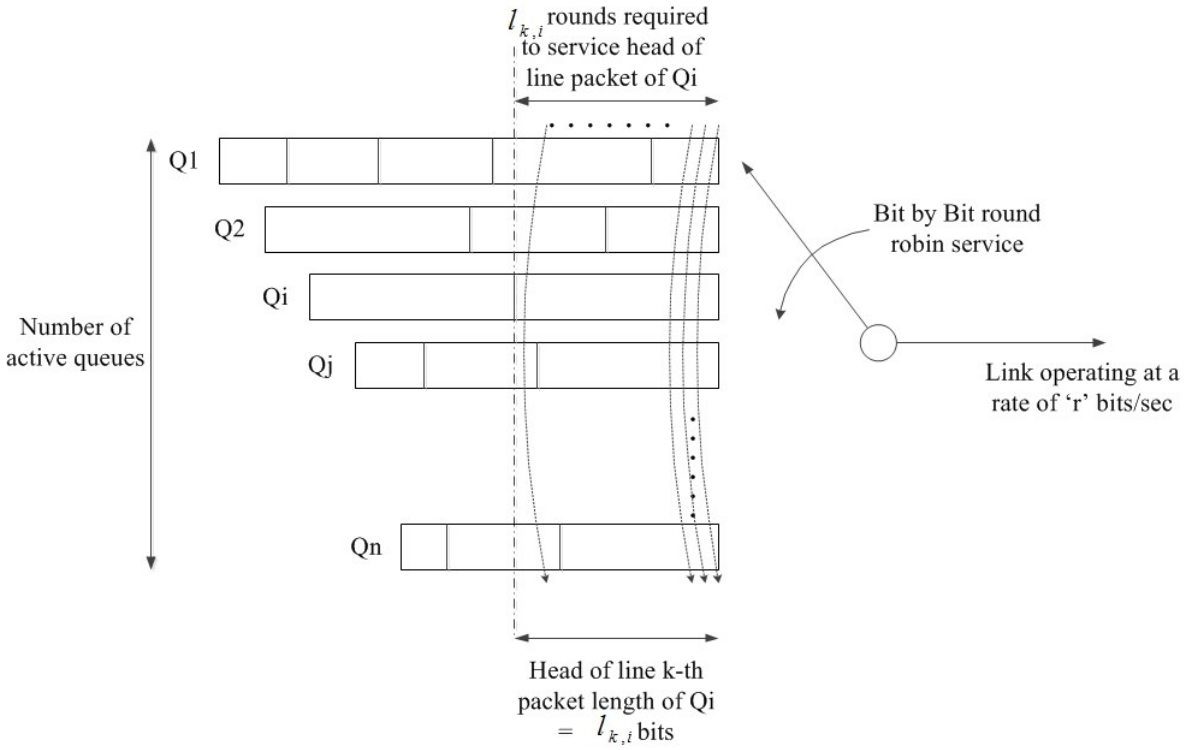


Figure 3.2: Practical bit by bit round robin system to approximate GPS system

This scheduling discipline, also called Fluid Fair Queuing [7], set the benchmark for the fairness of any other scheduling algorithm. A GPS server consists of N flows present in a GPS system, sending their data through a link at a rate of r bits per second. Every flow has a queue and the data queued in queue of each flow can be seen as fluid that is flowing continuously. If there are $n\ (\leq N)$ active flows or flows having packets to transmit in a round as shown in Figure 3.2, then the ideal GPS system is fair in the following sense: if the link is operating at a rate of r bits per second then the link rate will be equally shared by active flows in a round, resulting in r/n bits per second for an active flow [7, 39]. As shown in Figure 3.2, a round consists of a cycle in which all $n\ (\leq N)$ active flows are offered service. In practice, dividing the link rate exactly equally is not possible, therefore GPS service is approximated using bit by bit round robin service [39]. This scheduling discipline services each flow bit by bit in a round, as shown in Figure 3.2 [39]. A bit by bit system would begin by transmitting one bit from flow 1, then flow 2 and so on, in a

scheduling round. Therefore, n bits will be transmitted in a round, if there are n ($\leq N$) active flows present. As shown in Figure 3.2, it takes $l_{k,i}$ rounds to transmit head of line packet of flow i, where $l_{k,i}$ is the length of $k\ th$ packet in bits. The $k\ th$ packet is a head of line packet of flow i. Thus, a bit by bit round robin system can also be said as an approximated GPS system [39]. *Hence, from now onwards whenever we refer to the GPS system, we actually refer to the bit by bit round robin system.*

3.2.2. Packet by packet Generalized Processor Sharing (PGPS)

Since Generalized Processor Sharing (GPS) discipline services each active flow bit by bit in a round, which is practically not feasible hence; PGPS [7, 8] is introduced. PGPS is an extension to the GPS scheme, in response to its biggest demerit of considering bit by bit fluid traffic model [7]. Unlike GPS, traffic in PGPS is considered to be flowing in the form of packets as a whole [7, 8]. The goal of this scheme is to make the best possible approximation to the GPS scheme because GPS scheme is fair [7, 8]. In order to achieve the best possible approximation, PGPS tries to approximate the finishing number of a packet to the finishing number of same packet in GPS system [7, 8, 39]. For that, PGPS involves running a GPS system in parallel, called reference GPS system so that the finishing number of packet in GPS as well as PGPS system can be computed. Finishing number of a packet in PGPS is the number of scheduling rounds to be traversed in reference GPS system, from time $t = 0$ to the time when the packet arrives in PGPS system [39]. A packet is considered to be arrived in PGPS system when the last bit of packet has arrived [7]. Hence in PGPS, each time a packet arrives at a queue, the finishing number of a packet in PGPS is computed from GPS system. Each time a packet is serviced completely in PGPS system, the next packet to be serviced is the packet with smallest finishing number in PGPS system.

When fairness of PGPS is measured using absolute fairness index taking GPS as a benchmark discipline, it is found that PGPS is not able to completely catch up with the GPS scheme and it lags behind GPS scheme in terms of fairness [7, 8, 11]. This lag is expressed in terms of number of scheduling rounds, by which PGPS can lag behind reference GPS system. The value of this lag is bounded by the maximum length of packet in the system.

This means that in worst case, there is a difference of l_{max} scheduling rounds between the service received by a flow in PGPS and reference GPS system. This difference is absolute fairness index of PGPS and is upper bounded by l_{max}, where l_{max} is the maximum length of packet in the system. The upper bound of absolute fairness index is called absolute fairness bound.

3.2.3. Worst-case Fair Weighted Fair Queuing (WF^2Q)

It is concluded from the last section that PGPS lags behind reference GPS system and this lag is bounded by maximum length of packet (l_{max}) in the system. It was proven theoretically later in the study of WF^2Q that PGPS cannot lag behind reference GPS system. In fact, PGPS can lead reference GPS system and by even more than l_{max} scheduling rounds, in worst case [9]. Therefore, in order to keep the absolute fairness bound to l_{max}, a new scheduling discipline called WF^2Q [9] proposed an modification to PGPS. WF^2Q proposed that a packet having smallest starting as well as finishing number is selected for servicing next. The starting number is the number of scheduling rounds that need to be traversed in reference GPS system from the time $t = 0$ to the time when packet enters the queue in the PGPS system. A packet is considered to be entered in the queue in a system when the first bit of the packet arrives. This modification to PGPS ensured that absolute fairness bound remains l_{max}.

3.2.4. Self-Clocked Fair Queuing (SCFQ)

Packet by packet GPS (PGPS) and WF^2Q compute the finishing as well as starting number of a packet from the reference GPS system. This involves running a reference GPS system in parallel that results in huge computational complexity of fair scheduling disciplines like PGPS and WF^2Q. To remove the huge computational complexity involved in fair scheduling disciplines, a new scheduling discipline called SCFQ [10] was proposed. It proposed not to compute the starting as well as finishing number of a packet in fair scheduling disciplines, from the reference GPS system, but within the scheduling discipline [10]. This resulted in no requirement for a reference GPS system in parallel; hence a drastic

decrease in computational complexity is achieved. Furthermore, SCFQ redefines the idea of measuring the fairness of a scheduling discipline. As there is no reference GPS system anymore, SCFQ introduces a new measure of fairness of a scheduling discipline, called relative fairness index [10, 11, 37]. The relative fairness index is the difference between the service (in bits) received by two flows, in a scheduling round of a scheduling discipline [10, 11]. The maximum value of relative fairness index, as computed in the study of SCFQ for a fair scheduling discipline is $(l_{i,max} + l_{j,\max})$ where $l_{i,max}$ and $l_{j,\max}$ are the maximum length of packets of flow i and flow j, respectively [10]. This maximum value of relative fairness index is called relative fairness bound and means that during a scheduling round of a fair scheduling discipline, in worst case, the maximum disparity between the service received by a pair of flows i and j cannot exceed $(l_{i,max} + l_{j,\max})$. It is also proven that satisfying the bound of relative fairness index implies satisfying the bound of absolute fairness index and vice versa [11].

3.3. Wireless Network Scheduling Algorithms

3.3.1. Scheduling Algorithms for 2G systems

3.3.1.1. Channel State Dependent Packet Scheduling (CSDPS)

The working principle behind CSDPS is simple. In every scheduling round, the algorithm marks queues of those flows that are experiencing channel errors and defers packet scheduling for them. Flow is marked for a time interval of time out. If channel conditions improve for a flow before time out interval, it is unmarked before time out interval; otherwise it is unmarked after time out interval. The time out interval is equal to the average burst error time, which is the average duration over which channel conditions remain bad. Deferring the scheduling of flows with bad channel conditions, results in efficient use of wireless bandwidth because bandwidth gets distributed only among the flows with good channel conditions. Therefore, if a flow remains in good channel, it will keep receiving service and may exceed the service than the fair share. On the other hand, if a flow experiences burst errors frequently, then it will keep receiving less service than the fair share. Note that in CSDPS, if a flow which was experiencing errors, starts experiencing good channel conditions or recovered from errors, then bandwidth is shared by both the

flows: the flow which was already in good channel from long time and the flow which has just recovered from bad channel conditions. This is not fair allocation of bandwidth because the flow which has just recovered from bad channel conditions should be given excess bandwidth to compensate for all the past lost opportunities to transmit.

To solve this issue, CSDPS is combined with Class Based Queuing (CBQ). The algorithm is called CSDPS +CBQ [13]. CBQ exercises fair allocation of bandwidth among different flows. When CBQ is combined with CSDPS, then CBQ makes the flow in good channel conditions surrender fraction of its bandwidth, once any other flow recovers from bad channel conditions. The fraction of bandwidth surrendered by flow in good channel conditions, will act as excess bandwidth for flow which has just recovered from bad channel conditions. CSDPS working in conjunction with CBQ on the other hand, makes sure that transmissions of flows experiencing bad channel conditions should be deferred.

3.3.1.2. Idealized Wireless Fair Queuing (IWFQ)

Idealized Wireless Fair Queuing (IWFQ) defers the scheduling of all the flows that experience bad channel conditions, therefore all the flows with bad channel conditions lag behind in terms of service (in bits). On the other hand, all the flows with good channel conditions lead beyond, in terms of service. In order to compute the leading and lagging amount in bits, IWFQ proposed a scheduling discipline called Wireless Fluid Fair Queuing (WFFQ) which will be running in parallel with IWFQ [14]. In contrast to IWFQ where flows can experience either good or bad channel conditions, WFFQ is an ideal scheduling discipline where all the flows are assumed to be experiencing good channel conditions at all times. Since, in WFFQ all the flows experience good channel conditions at all times, therefore service received by a flow in WFFQ is the service which a flow will receive in ideal conditions or in the absence of bad channel at all times. It can also be said that service received by a flow in WFFQ is the service received by a flow in IWFQ when IWFQ is operating in wired networks because wired networks do not experience location-dependent channel errors. Since, wired networks do not experience location-dependent channel errors therefore channel conditions are assumed to be good at all times in wired networks.

Hence, IWFQ uses WFFQ as a reference system and computes the disparity between the service received by a flow in real and ideal channel conditions, by taking the difference between the service received by a flow in IWFQ and WFFQ. This disparity is also called the leading or lagging amount in bits. Once the flows with bad channel conditions return to good channel conditions then they are compensated for the amount by which they were lagging but the amount of compensation is bounded. Furthermore, when the compensation takes place, leading flows are penalized and hence they are not serviced at all for a while but the amount of penalty on leading flows is also bounded. Hence, the disparity of leading and lagging flows is bounded in IWFQ.

3.3.1.3. Channel condition Independent Fair Queuing (CIF – Q)

Since Idealized Wireless Fair Queuing (IWFQ) [14] penalizes leading flows completely when lagging flows are compensated for their lost service, this leads to heavy degradation in the QoS received by all the leading flows. Since 3G networks also support data along with voice, therefore sudden degradation in the QoS of flows with data services is not feasible for 3G networks [15]. Therefore, CIF–Q [15] proposed an improvisation in IWFQ and similar scheduling disciplines. It proposes that there should be *graceful degradation* in the QoS received by leading flows when lagging flows are compensated for their lost service [15]. To achieve this, CIF-Q proposes that leading flows should be penalized by a fixed fraction, but not completely. Therefore, in every scheduling round, a leading flow will receive $(1 - \alpha)$ fraction of service where α is a fixed fraction of bandwidth to penalize and, $0 \leq \alpha \leq 1$.

Scheduling schemes similar to IWFQ also involved running an ideal scheduling discipline like Wireless Fluid Fair Queuing (WFFQ) in parallel, to calculate the disparity in service received by a flow. Running a scheduling discipline in parallel, results in huge computational complexity hence CIF–Q proposes the use of relative fairness index and its bound, to bound the service disparity. Relative fairness index and its bound was first proposed by Self Clocked Fair Queuing (SCFQ) [10, 11]. As known from discussion on SCFQ, computing service disparity using relative fairness index, does not require a

reference scheduling discipline to be run in parallel, therefore computational complexity will get drastically reduced.

Furthermore, since 3G wireless networks also support data along with voice, it becomes mandatory to bound the service disparity between two flows with different kinds of services. CIF-Q defines a wireless scheduling discipline as short term fair if the discipline is able to keep the maximum service disparity between a pair of flows bounded by relative fairness bound of $(l_{i,max} + l_{j,\max})$ [10, 15]. The pair should either consist of both leading flows or both lagging flows, where $l_{i,max}$ and $l_{j,\max}$ are the maximum length of packets of flow i and flow j [10, 15].

Furthermore, CIF–Q proposes that Sustained Traffic Rate (STR) requirements for all the flows should be met. According to CIF–Q, if a wireless scheduling discipline is able to satisfy STR requirements for all flows, then a discipline is long-term fair. In a conclusion, CIF–Q defines a good wireless scheduling discipline to have all the three properties: Graceful degradation of leading flows, Short-term fairness within pair of either leading or lagging flows, and Long-term fairness for all the flows.

3.3.2. Scheduling Algorithms for 3G systems

3.3.2.1. Server Based Fairness Approach (SBFA)

Server Based Fairness Approach (SBFA) [1] is a scheduling policy for 2.5G and 3G wireless networks that use a specific modulation scheme with varying length codes. SBFA exploits the enhancements made in the physical layer of 3G networks and it can be integrated with any wired networks scheduling algorithm to adapt a wired network algorithm to wireless networks. 3G networks use a single modulation scheme with varying length codes. If the channel conditions are bad, then the physical layer of such networks uses long length codes. Otherwise if channel conditions are good, then it uses short length codes. SBFA defines maximum number of deferrals of packet transmission that are acceptable for a flow. The number of deferrals is expressed in terms of number of scheduling rounds. Hence, if a flow is delay-sensitive then the maximum number of deferrals for this flow will be small in comparison to that for delay-insensitive flow. If a

flow is experiencing bad channel conditions then SBFA will defer the transmission of that flow [1]. It considers the optimistic scenario that if channel conditions improve in the upcoming scheduling round, then it schedules the data of flow for transmission, and transmission will be done using short length codes [1]. However, if channel conditions do not improve, then SBFA will keep deferring the packet transmission until maximum number of deferrals have been reached, and schedules the data for transmission [1]. Furthermore, transmission will be done using long length codes because channel conditions are still bad [1].

3.3.2.2. Token Bank Fair Queuing (TBFQ)

Token Bank Fair Queuing (TBFQ) is a very novel approach that is able to achieve the three objectives: graceful degradation of leading flows, loosely bounded short-term fairness, and long term fairness for all the flows [19]. The system architecture of TBFQ consists of leaky buckets and a token bank. Every flow in TBFQ has a queue and a leaky bucket associated with it. A leaky bucket is defined with two parameters: bucket depth and bucket rate [39, 40]. During every scheduling round, the total number of C tokens is generated in the system, where C is the capacity of wireless link, in bytes [16, 37]. Hence, each token generated in a system corresponds to one byte of data [19]. The bucket rate is the rate at which leaky bucket gets refilled with tokens and is expressed in terms of number of tokens per second. Furthermore, the leaky bucket of each flow gets refilled at a rate corresponding to its minimum guaranteed rate (r_i) [19], decided by Service Level Agreements [17, 39, 40]. If the leaky bucket gets refilled at a rate of r_i tokens per second and d_i is the bucket depth, then bucket depth is given by:

$$d_i = r_i \times duration\ of\ a\ scheduling\ round \tag{3.1}$$

Therefore, if the leaky bucket of a flow gets refilled at a rate of r_i tokens per second, it is equivalent to say that during each scheduling round, the leaky bucket gets refilled with d_i tokens. If after the token buckets are completely filled with d_i tokens and there are still some tokens remaining in the system, then the remaining tokens are deposited in the token bank. The remaining tokens deposited in the token bank are called excess tokens [19]. The

token bank is central entity that manages the total tokens generated in the system, whereas excess tokens are used for supplementing the flows with excess bandwidth, if they need it. The number of tokens in a leaky bucket of a flow corresponds to the total number of bytes a flow can send to the output buffer, in a scheduling round [16, 19, 40].

There are several parameters associated with the leaky bucket of each flow and they are defined as follows:

- *Token Balance (E_i)*: Token balance is the number of excess tokens borrowed from or submitted to token bank, by a flow [16, 18, 37]. The token balance is given by:

$$E_i = d_i - b_i \tag{3.2}$$

  Where, b_i is the number of bytes a flow wants to send or demand of flow. Therefore, if the demand of flow is more than d_i bytes, then a flow borrows $(d_i - b_i)$ tokens from token bank and same amount is deducted from E_i. On the other hand, if the demand of flow is less than d_i bytes, then a flow deposits $(d_i - b_i)$ tokens to token bank and same amount is added to the E_i.

- *Priority Index (P_i)*: Priority index is a metric that determines the priority of a flow for borrowing excess tokens from the token bank. The higher the priority index, the higher the priority of a flow [18, 37]. The priority index is given by:

$$P_i = \frac{E_i}{r_i} \tag{3.3}$$

- *Burst Credit (BC_i)*: The maximum number of tokens that a flow can borrow from the token bank, in a scheduling round, is bounded by Burst Credit [16, 18, 37].
- *Debt Limit (DL_i)*: The maximum number of tokens that a flow can keep borrowing consecutively in more than one scheduling rounds, is bounded by Debt Limit [16, 18, 37].
- *Creditable Threshold (CT_i)*: The amount of tokens, a flow must deposit to the token bank, before it can again borrow tokens from the token bank [16, 18, 37].

In a scheduling round of TBFQ, all the flows are arranged in decreasing order of their priority index (P_i) and the system distributes excess tokens to the flows in the same

order [16, 18, 37]. A flow cannot borrow more than BC_i tokens in a scheduling round but if the flow will keep borrowing tokens consecutively in every scheduling round, then other flows will receive less service than their fair share. Hence, the disparity between service received by two flows will increase with the scheduling rounds because one flow will consecutively keep receiving excess service whereas other flow will consecutively keep receiving less service than the fair share [18, 37]. Therefore, the value of relative fairness bound will be large which means that the maximum service disparity between the two flows will be very large and that is not fair at all. For a scheduling discipline to be fair, the value of relative fairness bound should be smaller [19, 37]. Therefore, a flow can borrow maximum DL_i tokens consecutively, in more than one scheduling rounds. Once the flow has reached borrowing DL_i tokens from the token bank, then it is not allowed to borrow excess tokens and hence keep surrendering excess tokens to the bank so that other starving flows can borrow these tokens [18, 37]. Once the amount of tokens deposited by a flow to the bank, reach CT_i, then the same flow can borrow excess tokens again. The maximum service disparity or the Relative Fairness Bound (RFB) between the two flows is given by:

$$RFB = \left|Minimum\ service\ received_i - Maximum\ service\ received_j\right|$$

where $Minimum\ service\ received_i$ is the minimum service received by flow i and $Maximum\ service\ received_j$ is the maximum service received by flow j. According to TBFQ, the minimum service received by a flow i is zero and the maximum service received by another flow j is the case when it has already consumed DL_j tokens consecutively in say, N rounds. Hence, relative fairness bound in N^{th} round is given by [37]:

$$RFB = \left|0 - DL_j\right| = DL_j\ bytes$$

3.3.2.3. Channel State independent Wireless Fair Queuing (CS-WFQ)

Channel state independent Wireless Fair Queuing (CS-WFQ) [4] uses the notion of link adaptation. Link adaptation is a mechanism by which the rate of scheduled data is controlled according to channel conditions so that error free transmission can be achieved

in any value of Signal to Interference and Noise Ratio (SINR). Therefore, if SINR of a channel experienced by a flow is low, a low data rate modulation and coding scheme is selected. Channel State independent Wireless Fair Queuing (CS-WFQ) also proposes a reference system called Wireless Generalized Processor Sharing that computes a fair share of bandwidth of a flow in every scheduling round [4]. This fair share of bandwidth is the scheduled bandwidth. Furthermore, CS-WFQ determines the fair share of time slots for a flow according to channel capacity sensed by a flow in every scheduling round. Therefore, according to varying channel conditions, the capacity of wireless channel is varying in every scheduling round, which results in variable fair share of time slots for a flow. If channel capacity sensed by a flow is low, transmitting a scheduled amount of data will require more slots and vice versa [4]. In this case, if the flow occupies more time slots, then other flows will starve. Thus, CS-WFQ sets an upper bound on the number of time slots required by a flow to transmit its data, which will not allow the flow with extremely low sensed capacity to occupy time slots more than a certain threshold [4].

3.3.3. Scheduling Algorithms for 4G Systems

3.3.3.1. Multi Rate wireless Fair Queuing (MRFQ)

Multi Rate wireless Fair Queuing (MRFQ) [2] is a scheduling algorithm developed for wireless networks that have various modulation and coding schemes defined in their physical layer, as part of link adaptation mechanism. It exploits the feature of having multiple modulation and coding schemes in the physical layer of a wireless network, which is the case with 4G systems. During every scheduling round, the algorithm distributes bandwidth to all the active flows according to their fair share. After distributing bandwidth, if there is still bandwidth in the system, the algorithm distributes the remaining bandwidth among the flows that are in need of excess bandwidth, during the second iteration [2]. Before distributing excess bandwidth to a flow, MRFQ sets different threshold levels on the excess bandwidth required by a flow. The number of threshold levels is equal to the number of modulation and coding schemes used by a wireless network [2]. For example, if the wireless network uses four modulation and coding schemes, then the algorithm puts four thresholds on the excess bandwidth acquired by a flow. The four modulation and coding

schemes are then arranged in increasing order of data rates, as: $M_1 < M_2 < M_3 < M_4$, where M_4 can be any high data rate modulation and coding scheme such as ¾ 64-QAM and M_1 can be any low data rate scheme such as ½ QPSK. Accordingly, there are four thresholds on excess bandwidth required by a flow namely: $E_1 < E_2 < E_3 < E_4$, where E_4 is the maximum excess bandwidth that a flow can borrow, and E_1 is minimum excess bandwidth, a flow can borrow. In a specific scheduling round, if for example a flow borrows some excess bandwidth in the range of $[E_2, E_3)$, then the system will use only one of the modulation and coding schemes M_1 or M_2 for a flow. If the excess bandwidth lies in the range of $[E_3, E_4)$, then system will use only modulation and coding schemes M_1 or M_2 or M_3 for the flow [2]. Unlike Channel State dependent Wireless Fair Queuing (CS-WFQ), MRFQ does not takes into account the effect of varying total wireless channel capacity, which needs to be accounted for and assumes that total wireless channel capacity is constant. Hence, MRFQ is developed under wrong assumption of constant wireless channel capacity, which is not the case with the wireless systems using link adaptation mechanism.

3.3.3.2. Adaptive Token Bank Fair Queuing (ATBFQ)

Adaptive Token Bank Fair Queuing (ATBFQ) [24, 41] is an extension to the TBFQ [19] algorithm. TBFQ is developed for 3G networks and also assumes channel conditions as either good or bad [16, 18, 19]. The transmissions of the flows experiencing bad channel conditions will be deferred in TBFQ, and they surrender all their bandwidth in a scheduling round. When they surrender their bandwidth, then flows with bad channel will submit all the tokens back to the token bank resulting in surplus excess tokens available in token bank. In the second iteration, excess tokens are distributed to the active flows in the order of decreasing priority index, but a flow cannot borrow more than burst credit (BC_i) amount of tokens. Therefore, it will be the case in TBFQ that even each flow borrows BC_i amount of tokens, there are still tokens left in token bank, resulting in inefficient use of wireless channel bandwidth.

Therefore, ATBFQ proposes to make burst credit (BC_i) of each flow dynamic or adaptive [24]. If there is a small number active flows in a specific scheduling round, there will likely be surplus excess tokens available in the token bank. Therefore, ATBFQ

increases the BC_i of each active flow so that all the excess tokens available in token bank can be used. Furthermore, if the number of active flows increase in the system, then ATBFQ decreases the BC_i of a flow so that every flow can borrow some excess tokens from the token bank. Similar to TBFQ, ATBFQ also assumes that the total tokens generated in a system, in a scheduling round, are constant [24], which is equivalent to say that total wireless channel capacity is not varying, in a scheduling round. After scheduling the bandwidth of a flow, ATBFQ maps the data of that flow onto OFDMA resource matrix and then schedules the bandwidth for the next active flow [24, 41]. Since, channel conditions will never be considered bad in 4G wireless networks because these networks use link adaptation with combination of seven modulation and coding schemes, all flows will experience good channel condition to a certain degree. Therefore, the number of active flows will not vary because of deferral of transmissions by several flows but because several flows would have no data to transmit. Since, for heavy load on the network, flows having no data to transmit is a rare possibility, variation in BC_i does not only depends upon number of active flows. It also depends on the number of remaining resource blocks in the OFDMA resource matrix as well as modulation and coding scheme to be used by a flow [24, 41]. 4G systems make the channel conditions always appear as good because they use link adaptation. Since, ATBFQ exploits this advanced feature of physical layer of 4G systems, therefore there is no need for deferring of transmission of users with bad channel conditions. As, transmission of users with bad channel conditions is not deferred, hence maximum service disparity between the two active flows is always tightly bounded in a scheduling round. Therefore, ATBFQ provides good short-term fairness [23]. Since, ATBFQ also assumes that constant number of tokens get generated in a system, in a scheduling round, therefore it is also developed under the wrong assumption of constant wireless channel capacity in a scheduling round.

3.4. Burst Construction Algorithms

A burst construction algorithm packs the scheduled data of a user into a burst and maps the burst onto a DL sub-frame. While mapping the burst of a user, a burst construction algorithm can consider one or more than one constraints such as QoS of user,

minimization in the wastage of resource blocks, power consumption, etc. In this section, major existing burst construction algorithms will be discussed and it will be observed that how mapping in a specific way helps satisfying different constraints. The concept of unoccupied and wasted resource blocks will be also discussed.

3.4.1. Fixed Burst Approach

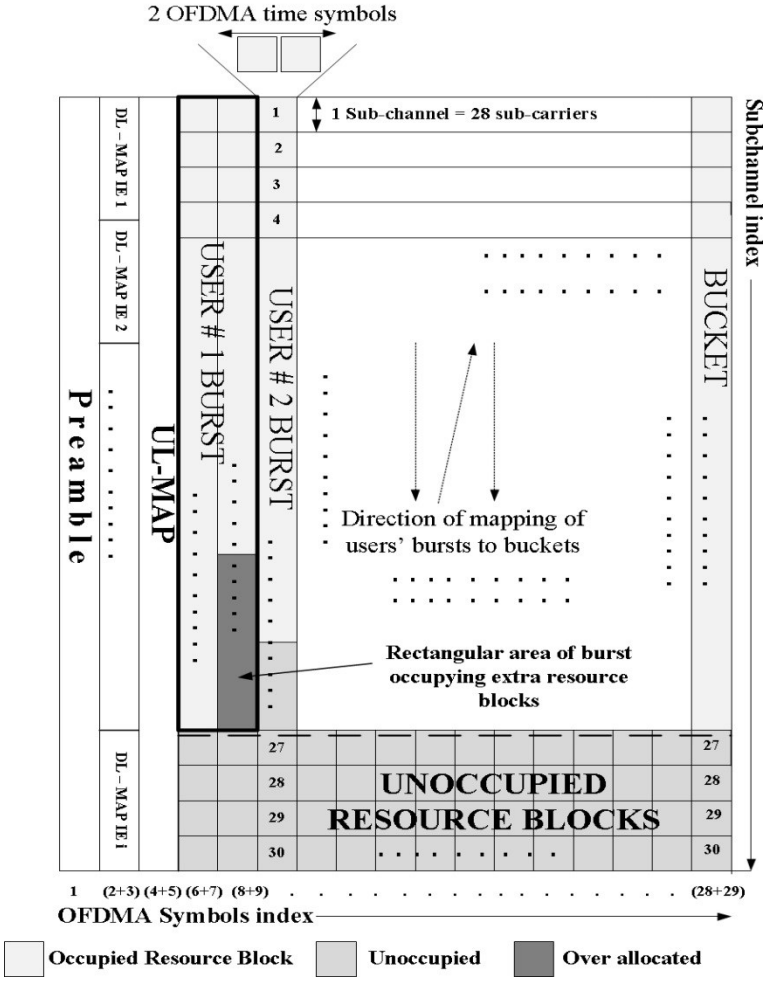


Figure 3.3: Mapping of users' bursts to resource matrix in fixed burst approach

As shown in Figure 3.3, the fixed burst approach [20] maps the bursts of users onto one or more than one buckets. It is now known that a resource block occupies two OFDMA time symbols on the horizontal frequency axis. A bucket is a rectangular area defined as $2\ time\ symbols \times n\ subchannels$, where $0 < n \leq 30$ and n is also the number of resource blocks. This rectangular area is also called the size of bucket. Hence, the maximum rectangular area occupied by a bucket is $2\ time\ symbols \times 30\ subchannels$

and can consist of maximum 30 resource blocks. The size of a burst of user in the downlink sub-frame is measured in terms of number of resource blocks occupied by the burst of a user in downlink sub-frame. Therefore, the size of a bucket is the maximum burst size among the burst sizes of all users that can be completely fit in the maximum rectangular area occupied by a bucket i.e. $2\ time\ symbols \times 30\ subchannels$. Similarly, fixed burst approach [20] computes the size of a bucket in a frame duration. As shown in Figure 3.3, fixed burst approach keeps allocating buckets to the burst size of user until area occupied by allocating buckets is less than burst size.

According to the constraint of absolute rectangular shape of IEEE 802.16e standard [33], the burst of a user should occupy a rectangular area, whose lengths and breadths are equal. Hence, when the burst of a user is mapped on the DL sub-frame, the absolute rectangular area of a burst can end up occupying some extra resource blocks as shown in Figure 3.3. The absolute rectangular area occupied by *user # 1 burst* in Figure 3.3 here occupies extra resource blocks. Burst of any other user cannot be mapped on those extra resource blocks occupied by absolute rectangular area of *user # 1 burst*. The burst of any other user cannot be mapped because if the burst of any other user is mapped on the extra resource blocks, then the area occupied by *user # 1 burst* will not anymore comply with the definition of absolute rectangular shape. The area occupied by *user # 1 burst* will not comply with the definition of absolute rectangular shape because both the length of rectangular area (number of subchannels occupied on vertical frequency axis) will not be equal anymore. Hence, no users' burst can be mapped on those extra resource blocks. Therefore, some resource blocks get wasted like that and such kind of wastage is called wastage due to over-allocation of resource blocks. On the other hand, the shaded area in the bottom, as shown in Figure 3.3, contribute to unoccupied resource blocks.

Mapping the bursts of users in a way, as shown in Figure 3.3 also leads to large fraction of unoccupied as well over allocated resource blocks because the size of bucket is computed once at the start of every frame duration and is fixed. Since, the size of a bucket is not computed for burst size of every user, in a frame duration hence; the computational complexity of calculating the exact size of bucket for burst size of each user in every scheduling round, is greatly reduced.

3.4.2. Mapping with Appropriate Truncation and Sort (MATS)

Unlike fixed burst approach, the MATS algorithm [21] tries to achieve almost 100 percent packing efficiency. During the first iteration of a scheduling round, MATS inserts the scheduled data of different users into a First In First Out (FIFO) queue, called Request Queue [21]. MATS then tries to map the scheduled data of different users in the request queue into exact rectangular allocation. If MATS is not able to map the complete scheduled data of every user in the request queue in a single iteration, it fragments the scheduled data of the user and sends the user's unmapped residual scheduled data into another First In First Out queue called, Fragmentation Queue [21]. When the first iteration of mapping of scheduled data is finished, MATS first checks the request queue for scheduled data to be mapped [21]. If there is still scheduled data of users queued in request queue, then first MATS will map the scheduled data in request queue until the request queue is empty or resource matrix is fully occupied. If the request queue is empty and there are still left-over resource blocks in the resource matrix, MATS checks the fragmentation queue to map the residual scheduled data of users. Mapping of each user's scheduled data in such a way leads to distribution of user's data all over the resource matrix, which results in formation of more than one burst of a user, which increases the amount of DL-MAP message stored for a user. Therefore, packing of user's data in such a way increases the amount of control information.

3.4.3. Burst placement for optimized receiver duty cycling

Burst placement for optimized receiver duty cycling [36] tries to minimize the wakeup time of the mobile stations. It is known that when Connection Identifier (CID) of a mobile station is included in the DL-MAP message of mobile station, then the wakeup time of mobile station is simply the burst duration. Connection Identifier is the address of mobile station to which a burst is addressed to. When CID of mobile station is not included in DL-MAP message of mobile station, the wake up time of mobile station is the sum of burst duration and burst delay. Burst placement for optimized receiver duty cycling [36] deploys recursive binary tree full search approach to find the best possible fit for a burst of mobile station. The recursive binary tree full search approach divides the downlink sub-frame into

two equal parts and compares the size of the burst to be mapped with the size of one of the parts. If the size of half part of downlink sub-frame is greater than the size of the burst of a user, then left half of the downlink sub-frame is further divided into two equal parts, considering that initially, left part of downlink sub-frame is selected. Further, again the size of one of the half of left half of downlink sub-frame is compared with size of burst of user. This recursive process continues until the best fit for the size of a burst is found or it reaches to the point where the size of the burst to be mapped is greater than the smallest area recovered [36]. If the size of the burst is greater than smallest area recovered on left half of downlink sub-frame, then the same recursive process is repeated with right half of DL sub-frame until the best fit for the size of a burst is found. The best possible fit results in the minimum average burst duration as well as average burst delay.

The biggest advantage of Burst placement for optimized receiver duty cycling [36] is that, it is able to achieve significant minimization in the average wakeup time of receivers, thereby decreasing the power consumption. There are however many disadvantages of this approach. First, since it uses recursive binary tree full search approach, the computational complexity is huge. Second, it focusses only on power consumption of mobile stations and not on other constraints such as wastage of resource blocks. Third, the algorithm is limited to just eight users in a frame. The algorithm is limited to just eight users because as number of users exceed more than eight users, computational complexity related to finding the best possible fit for all users, quickly grows.

3.4.4. enhanced One Column Stripping with non-increasing Area (eOCSA)

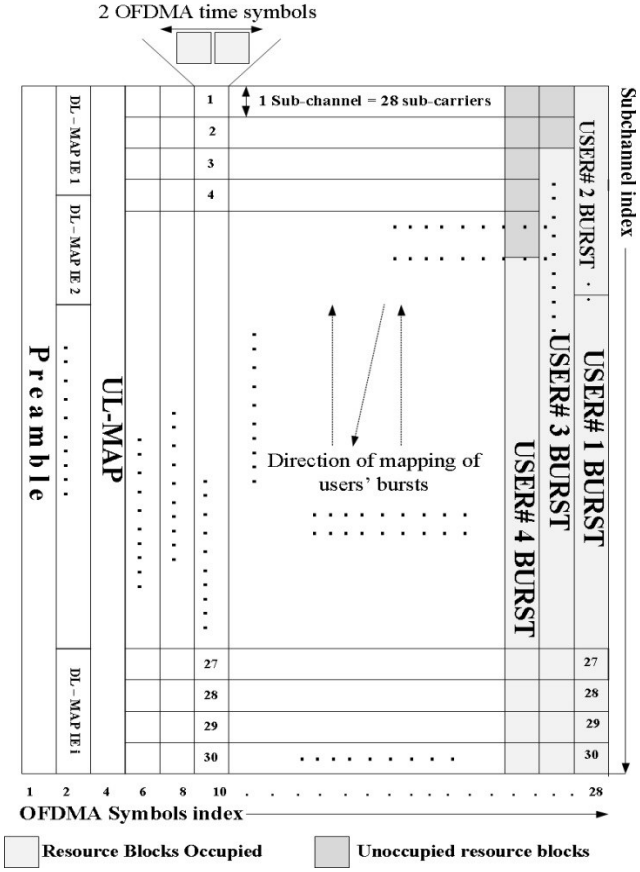


Figure 3.4: Mapping of users' burst to resource matrix in eOCSA

As shown in Figure 3.4, eOCSA [3] maps the bursts of the users starting from the end of downlink sub-frame towards the start of the downlink sub-frame and from bottom to top [5]. The start of downlink sub-frame will be the point where DL-MAP messages of users start. eOCSA first arranges the users in decreasing order of size of their respective bursts. After mapping the burst of the first user in the vertical direction, if there are any left-over resource blocks on the top of the resource matrix, eOCSA tries to find any other users whose bursts can fit the left-over space on the top [5]. In this way, it is able to achieve very high packing efficiency because very small resource blocks get wasted. Furthermore, as shown in Figure 3.4, eOCSA maps the burst of a user vertically and therefore tries to keep the burst duration as short as possible [3, 5]. Although eOCSA arranges the users in decreasing order of the size of burst, whenever it encounters any user with delay sensitive data to be mapped, eOCSA will not proceed to the next user, until it has mapped the data of

user with delay sensitive data [3]. Therefore, eOCSA also considers the QoS of different users while mapping bursts of different users.

3.5. Conclusion

A good scheduling algorithm for 4G networks should ensure short-term as well long-term fairness among different flows. Ensuring short-term fairness will satisfy QoS requirements for different users. For an algorithm to meet short-term fairness, it should bound the maximum service disparity between any pair of flows by a maximum value called relative fairness bound. The lesser the value of this bound, the fairer the algorithm will be. On the other hand, if an algorithm is long- term fair, then it should be able to provide reasonable cell throughput or be able to satisfy STR requirements of every user i.e. the algorithm should also be opportunistic in behaviour [38]. Furthermore, the scheduling algorithm should take into consideration the enhancements made in the physical layer of wireless systems while making a scheduling decision. 4G systems like Mobile WiMAX are developed to provide true broadband wireless access in highly mobile wireless environments, which results in varying wireless channel capacity in almost every scheduling round. Hence, scheduling algorithm for 4G networks should be developed under the assumption of highly mobile wireless environment. The algorithms for 4G networks mentioned in this thesis: Multi Rate Fair Queuing (MRFQ) and Adaptive Token Bank Fair Queuing (ATBFQ) are not developed under the assumption of highly mobile environment.

Since, none of the burst construction algorithms mentioned in this thesis tried to address the issue of fairness in average wakeup times (average power consumption), a burst construction algorithm should address this issue. The most elementary unit of physical radio resource allocated to a user in Mobile WiMAX networks is a resource block. Since there are limited resource blocks in OFDMA resource matrix so their wastage should be avoided. As shown in Figure 3.3, resource blocks get wasted due to either over allocation or because none of the users' burst can be mapped to the resource matrix. Therefore, a burst construction algorithm should always minimize the both types of wastage.

Chapter 4:

Proposed Algorithms

4.1. Introduction

A dynamic bandwidth scheduling algorithm named Leaky Bucket Token Bank (LBTB) and burst construction algorithm named Burst Construction for Fairness in Power (BCFP) for Mobile WiMAX systems is proposed in this chapter. The system architecture of LBTB is based on the Token Bank Leaky Bucket architecture [17]. LBTB attempts to gain short-term fairness at the expense of high system throughput. It considers traffic backlog, Class of Service, and modulation and coding scheme used by a flow to service different flows during a given scheduling round. Furthermore, LBTB considers every flow's transmission history, while penalizing or distributing excess bandwidth to it.

The proposed burst construction algorithm (BCFP) attempts to minimize the burst duration and wastage of resource blocks whenever possible. Furthermore, it tries to achieve fairness in average wakeup time (power consumption) for the case, when connection identifier of mobile stations are not included in DL-MAP message.

4.2. System Architecture

It is known that the scheduler and burst construction mechanism work in MAC layer of Mobile WiMAX networks. Burst construction mechanism receives the data from the scheduler and packs them in the form of bursts. Furthermore, the burst construction mechanism maps bursts in rectangular arrangement onto the OFDMA radio resource matrix.

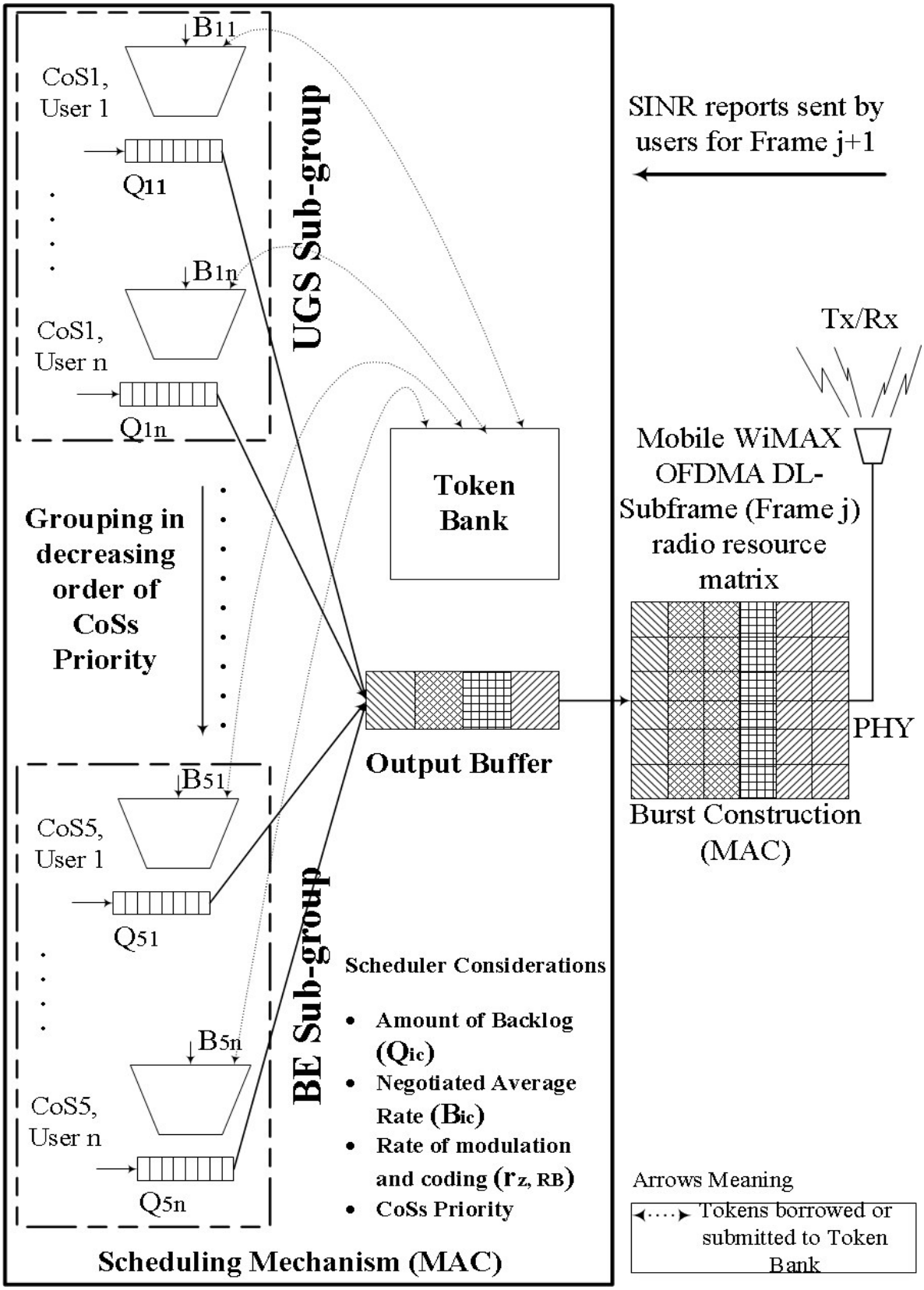


Figure 4.1: System Architecture at MAC Layer of Mobile WiMAX

As shown in Figure 4.1, the system architecture of LBTB consists of leaky buckets and a token bank. At the start of a scheduling round, LBTB gathers the information about the SINR experienced by each user present in a cell coverage area, through the SINR reports sent by them, as shown in Figure 4.1. Based on the SINR reports sent by all users, LBTB determines the fraction of users from total users, present in each of the seven zones, using Table 2.2. Furthermore, the physical layer uses link adaptation to assign the amount of scheduled data carried by a resource block for the users present in zone z. Let U_z represent the fraction of users from total users, present in zone number z and N_{RB} represents the total number of resource blocks in the downlink sub-frame, then amount of resource blocks from N_{RB} resource blocks that can be allocated to all the users present in a zone number z, $N_{z,RB}$ can be determined by using equation (2.7) as:

$$N_{z,RB} = U_z \times N_{RB}$$

Let $b_{z,RB}$ represents the amount of scheduled data carried by each of $N_{z,RB}$ resource blocks, then the data rate of each of $N_{z,RB}$ resource blocks is given by:

$$r_{z,RB} = \frac{Amount\ of\ scheduled\ data\ carried\ by\ a\ resource\ block}{Duration\ of\ a\ resource\ block\ on\ time\ axis} = \frac{b_{z,RB}}{2 \times T_s} \tag{4.1}$$

where, $r_{z,RB}$ represents the data rate of each of $N_{z,RB}$ resource blocks allocated to all the users present in a zone number z and T_s is the duration of a time symbol on horizontal time axis. Furthermore, let C represent the time varying capacity of wireless link in a scheduling round, then C in bytes is determined by using equation (2.8), as:

$$C = \frac{\sum_{z=1}^{7}(b_{z,RB} \times N_{z,RB})}{8} \tag{4.2}$$

Therefore, in a scheduling round, C tokens are generated in the system. Hence, one token has a value equivalent to one byte and the words ‘token’ and ‘byte’ can be used interchangeably. Every CoS (flow) of each user in LBTB has a queue and a leaky bucket associated with it. The number of bytes sitting in the queue of CoS c of user i at any time, is called queue length and is represented by Q_{ic}. The CoS c can be represented according to five different CoSs defined in the IEEE 802.16e standard [33], as:

$$\text{CoS } c = \{1\ for\ UGS, 2\ for\ ertPS, 3\ for\ rtPS, 4\ for\ nrtPS, 5\ for\ BE\}$$

A leaky bucket is defined with two parameters: bucket depth and bucket rate. The bucket depth is related to the bucket rate as follows:

$$bucket\ depth = bucket\ rate\ \times duration\ of\ a\ scheduling\ round \tag{4.3}$$

The bucket rate is expressed in terms of tokens per second. As shown in Figure 4.1, in LBTB, the leaky bucket of each flow gets refilled at the rate corresponding to sustained traffic rate, B_{ic} of CoS c of user i. The sustained traffic rate is also called average rate, therefore in LBTB, we call the bucket depth related to bucket rate of B_{ic} tokens per second by equation (4.3), as the *average depth*. Let D_{ic} represents the average depth of a leaky bucket of CoS c of user i, then average depth is given by equation (4.3) as:

$$D_{ic} = B_{ic} \times duration\ of\ a\ scheduling\ round$$

Similarly, we call the bucket depth related to minimum reserved traffic rate by equation (4.3), as the *minimum depth*, M_{ic}. Furthermore, we call the bucket depth related to maximum sustained traffic rate by equation (4.3), as the *maximum depth*, P_{ic}. Let $s_{ic,n}$ represents the amount of bytes scheduled for CoS c of user i in $n-th$ scheduling round and A_{ic} represents the amount of bytes scheduled for CoS c of user i in current scheduling round, when averaged over all past scheduling rounds, then A_{ic} is given by:

$$A_{ic} = \frac{\sum_{n=1}^{(N-1)} s_{ic,n}}{(N-1)} \tag{4.4}$$

Where, $N-th$ scheduling round is the current scheduling round.

The proposed scheduling algorithm (LBTB) classifies each flow at the start of a scheduling round as leading or lagging based on the following condition:

$$flow\ is \begin{cases} lagging, & if\ A_{ic} < D_{ic} \\ leading, & if\ A_{ic} > D_{ic} \end{cases} \tag{4.5}$$

There are set of parameters associated with lagging and leading flows. The following set of parameters are particularly associated with *lagging flows*:

- *Excess Tokens* (E_{ic}): The maximum amount of extra tokens that *can be* granted to a lagging flow during a scheduling round. Excess tokens for any lagging flow are upper bounded by following set of inequalities:

$$E_{ic} = \begin{cases} 0, & Q_{ic} < D_{ic} \\ Q_{ic} - D_{ic}, & D_{ic} \leq Q_{ic} \leq P_{ic} \\ P_{ic} - D_{ic}, & Q_{ic} > P_{ic} \end{cases} \tag{4.6}$$

- *Upgraded Excess Tokens* ($\alpha_{ic}E_{ic}$): The amount of excess tokens that *is* granted to a lagging flow during a scheduling round.
- *Scheduled bytes* (u_{ic}): The amount of bytes scheduled for a lagging flow in a scheduling round. The scheduled bytes are given by:

$$u_{ic} = D_{ic} + \alpha_{ic}E_{ic} \tag{4.7}$$

Figure 4.2 shows the parameters particularly associated with lagging flows.

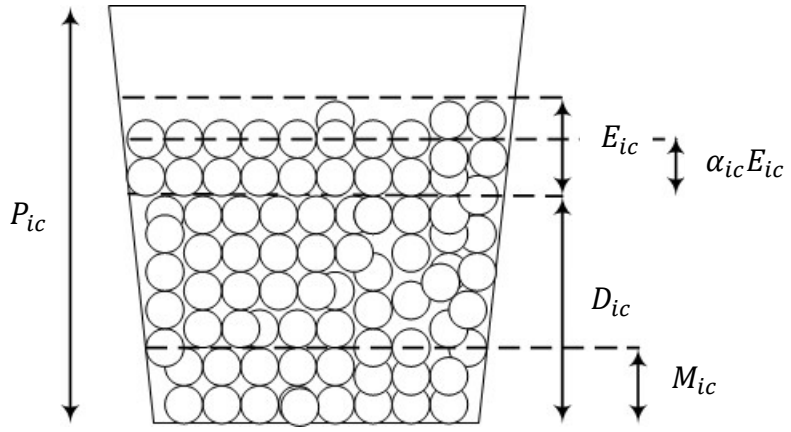


Figure 4.2: Parameters particularly associated with Lagging flows

The following set of parameters are particularly associated with *leading flows*:

- *Penalized Tokens* (X_{ic}): The maximum amount of tokens by which a leading flow *can be* penalized during a scheduling round. Penalized tokens for any leading flow are lower bounded by the following set of inequalities:

$$X_{ic} = \begin{cases} 0, & Q_{ic} \leq D_{ic} \\ D_{ic} - M_{ic}, & Q_{ic} > D_{ic} \end{cases} \quad (4.8)$$

- *Degraded Penalized Tokens* ($\beta_{ic}X_{ic}$): The amount of penalized tokens by which a leading flow *is* penalized during a scheduling round.
- *Scheduled bytes* (d_{ic}): The amount of bytes scheduled for a leading flow in a scheduling round. The scheduled bytes are given by:

$$d_{ic} = D_{ic} - \beta_{ic}X_{ic} \quad (4.9)$$

Figure 4.3 shows the parameters associated with leading flows.

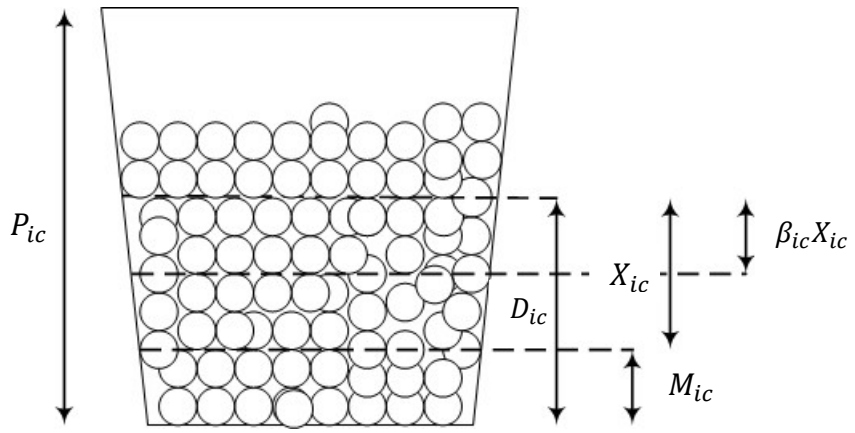


Figure 4.3: Parameters particularly associated with Leading flows

Therefore, the bytes scheduled by LBTB for *any* CoS c of user i in $n-th$ scheduling round, is given by:

$$s_{ic,n} = \begin{cases} u_{ic}, & for\ lagging\ flows \\ d_{ic}, & for\ leading\ flows \end{cases} \quad (4.10)$$

4.3. LBTB Algorithm

The proposed scheduling algorithm is called Leaky Bucket Token Bank (LBTB). It is a downlink algorithm which executes at the base station and schedules data in bytes for different users. The scheduled data is then sent to burst construction mechanism that converts the data into bursts and maps them on OFDMA Mobile WiMAX downlink sub-frame. The following steps are involved in scheduling data for different users:

Iteration 1 of a round

Step 1:

All the backlogged flows are selected in a scheduling cycle. They are grouped into two different groups: Leading and Lagging, using condition (4.5), that is:

$$flow\ is \begin{cases} lagging, & if\ A_{ic} < D_{ic} \\ leading, & if\ A_{ic} > D_{ic} \end{cases}$$

Step 2:

Within each group, all the flows are grouped according to their CoS priorities as shown in figure 4.1. Therefore, there is a group for each CoS priority within both leading and lagging groups, called *priority sub-groups*. All the individual CoS priority sub-groups within leading and lagging groups are arranged in decreasing order of their respective priorities, as shown in figure 4.1. Furthermore, within each CoS priority sub-group in both leading and lagging groups, flows are arranged in decreasing order of following ratio:

$$R_{ic} = Q_{ic} \times \frac{B_{ic}}{B_{c,min}} \times \frac{r_{lowest}}{r_{z,RB}} \tag{4.11}$$

Where, $B_{c,min}$ is the lowest average rate among the individual CoS c sub-group within leading and lagging group, $r_{z,RB}$ is computed from equation (4.1) and is the data rate of a resource block carrying the scheduled data of CoS c of user i in zone number z and r_{lowest} is the lowest data rate of a resource block among the data rates of a resource block carrying scheduled data of all flows in leading and lagging groups.

Step 3:

LBTB then computes the capacity of wireless link using equation (4.2) and C tokens are generated in a system. Furthermore, LBTB fills the leaky buckets of all the backlogged flows except CoS 5 flows, in leading and lagging groups, with D_{ic} amount of tokens because CoS 5 flows require no guarantees in terms of rate. The D_{ic} amount of tokens for every flow c of user i is distributed from C tokens such that:

$$\sum_{i=1}^{n} \sum_{c=1}^{4} D_{ic} < C \tag{4.12}$$

Where n is the number of users having backlogged flows. After the D_{ic} tokens have been distributed among each leading and lagging backlogged flow c of user i, the remaining tokens in the system, C_r is given by:

$$C_r = C - \sum_{i=1}^{n} \sum_{c=1}^{4} D_{ic} \tag{4.13}$$

Therefore, at the end of first iteration of a scheduling round, remaining C_r tokens in the system are deposited in the token bank.

Iteration 2 of a round

Step 1:

Leading group is selected first for servicing. In this group, leading flows are penalized in the same order they are arranged, as in step 2 of iteration 1. Let n_{lead} be the number of backlogged flows in a leading group, n_{lag} be the number of backlogged flows in lagging group and n_p be the number of flows penalized. Each leading flow is penalized by $\beta_{ic} X_{ic}$ amount of penalized tokens, where β_{ic} is the dynamic fraction of penalized bytes and β_{ic} is computed as follows:

$$\beta_{ic} = e^{-g}, \tag{4.14}$$

where $g = \frac{N_{lead}}{n_{lag}+n_{lead}}$, $N_{lead} = n_{lead} - n_p$ with $n_p = 0$ at the start of every scheduling round and with every penalized flow, $n_p = n_p + 1$. Furthermore, every time a flow is penalized, $\beta_{ic} X_{ic}$ amount of penalized tokens are deposited in the token bank. Therefore,

when all the leading flows get penalized, then amount of tokens in token bank are updated by:

$$C_r = C_r + \sum_{n_{lead}} \beta_{ic} X_{ic} \tag{4.15}$$

Note that the number of tokens in token bank will now be more than the initial number of tokens in token bank given by equation (4.13) at the end of iteration 1.

Step 2:

Lagging group is now selected for servicing. In this group, lagging flows are compensated in the same order they are arranged, as in step 2 of iteration 1. Let n_e be the number of flows compensated. Each flow is compensated by $\alpha_{ic} E_{ic}$ amount of excess bytes, where α_{ic} is the dynamic fraction of excess bytes and α_{ic} is computed as follows:

$$\alpha_{ic} = 1 - e^{-d}\ , \tag{4.16}$$

where $d = \frac{N_{lag}}{n_{lag} + n_{lead}}$, $N_{lag} = n_{lag} - n_e$ with $n_e = 0$ at the start of every scheduling round and with every compensated flow, $n_e = n_e + 1$. Furthermore, every time a flow is compensated, $\alpha_{ic} E_{ic}$ amount of excess tokens are borrowed from token bank. Therefore, when all the lagging flows get compensated, then amount of tokens in token bank are updated by:

$$C_r = C_r - \sum_{n_{lag}} \alpha_{ic} E_{ic} \tag{4.17}$$

Step 3:

If C_r is still not zero then rest of the tokens are distributed among CoS 5 flows. First, LBTB serves the CoS 5 flows in lagging group in decreasing order of ratio R_{i5} computed by using equation (4.11) and distributes D_{i5} amount of tokens from C_r tokens in the token bank. If C_r is still not zero, then LBTB distributes the tokens from remaining C_r tokens to the CoS 5 flows in leading group in the similar way as it does for CoS 5 flows in lagging group.

4.4. BCFP Algorithm

The proposed burst construction algorithm (BCFP) packs the amount of scheduled bytes of a user into a burst. The amount of scheduled bytes of user j in $n-th$ scheduling round is given by:

$$s_{jn} = \sum_{c=1}^{5} s_{jc,n} \tag{4.18}$$

where $s_{jc,n}$ is the amount of scheduled bytes of CoS c of user j in $n-th$ scheduling round as computed from equation (4.10) and s_{jn} is the amount of scheduled bytes of user j in $n-th$ scheduling round. A burst consists of one or more than one resource block and the number of resource blocks in a burst of user j in $n-th$ scheduling round, RB_{jn} is given by:

$$RB_{jn} = \frac{s_{jn}}{b_{z,RB}} \tag{4.19}$$

where $b_{z,RB}$ is the amount of scheduled bytes of user j currently located in zone number z, carried by a resource block, as introduced in equation (4.1). Therefore, user j's scheduled data in $n-th$ scheduling round is packed into a burst which consists of RB_{jn} resource blocks. A burst construction algorithm then maps the burst of user j on downlink sub-frame in $n-th$ scheduling round by mapping the scheduled data on RB_{jn} left-over resource blocks in downlink sub-frame.

As it is known that a resource block occupies two time symbols on horizontal time axis, therefore the area of $2\ time\ symbols \times 30\ subchannels$ consists of 30 resource blocks. In BCFP, we call this area as a *strip*, hence a strip consists of 30 resource blocks as shown in figure 4.4. Since one time symbol is occupied by preamble and two time symbols are occupied by DL-MAP and UL-MAP each, therefore the remaining $29-(1+2+2)=24$ time symbols are available for transmitting scheduled data for different users. As it is known that a strip occupies two time symbols, hence there will be $\frac{24}{2}=12$ strips available for transmitting scheduled data of different users, where each strip consists of 30 resource blocks. Since, BCFP starts mapping of bursts from the end of downlink sub-frame i.e. from *time symbol # (28+29)* to *time symbol # (6+7)*, therefore strips will be numbered in

increasing order. In BCFP, the strip occupying *time symbol # (28+29)* will be numbered as *strip # 1* and so on strip occupying *time symbol # (6+7)* will be numbered as *strip # 12*. We define L_i to be the number of empty resource blocks in *strip # i* in a scheduling round and call it the length of a *strip # i*. Let v_{jn} be the number of strips on horizontal time axis occupied by the burst of user j in $n-th$ scheduling round and is also called burst duration of user j in $n-th$ scheduling round as shown in Figure 4.4. Similarly, we represent the burst delay of user j in $n-th$ scheduling round as shown in Figure 4.4, by h_{jn}.

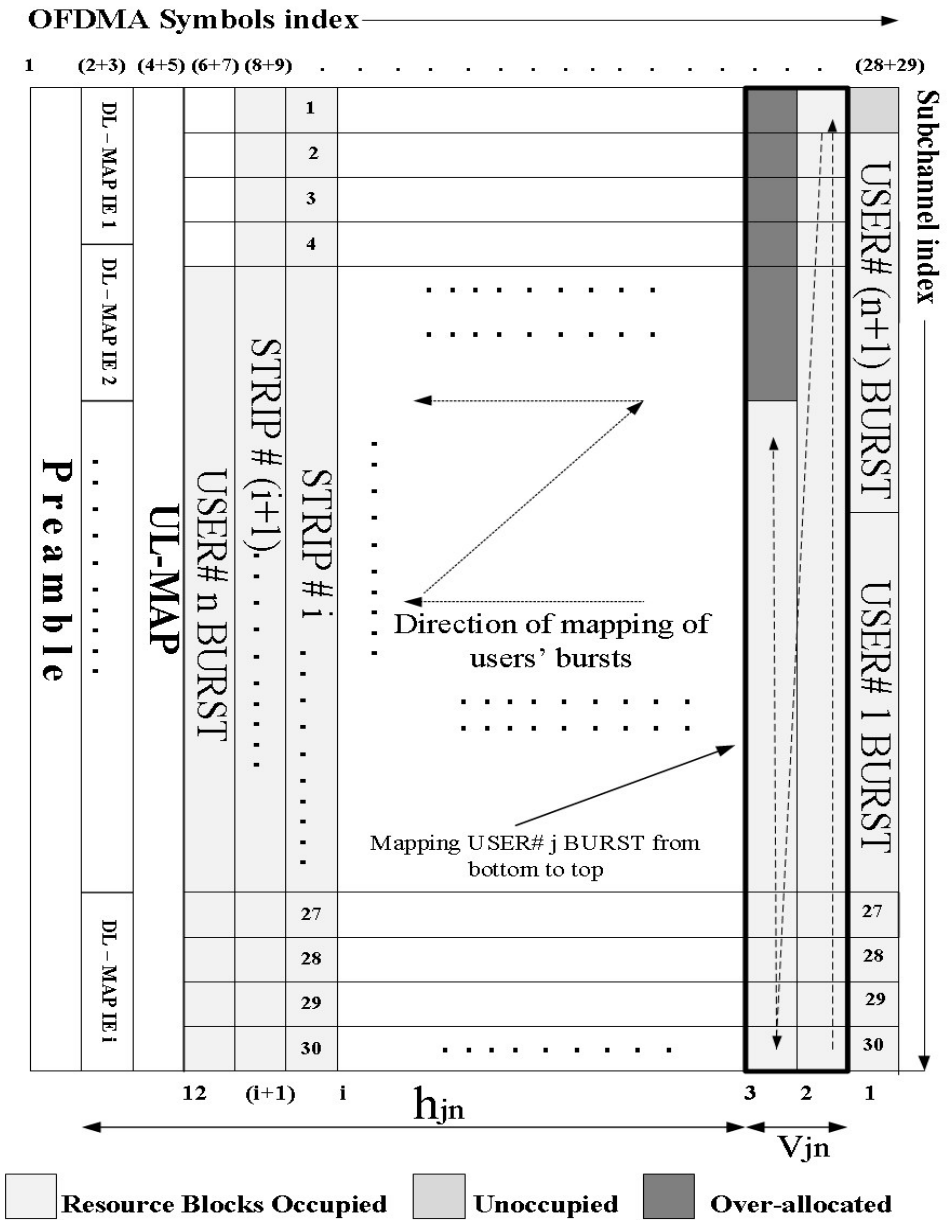


Figure 4.4: Burst mapping in BCFP Algorithm

As shown in Figure 4.4, LBTB maps the burst in a strip from bottom to top and right to left. Therefore, if number of resource blocks in the burst of user j in $n-th$ scheduling round, RB_{jn} is greater than the length of a *strip # i,* L_i, then it occupies another strip and so on. According to the IEEE 802.16e standard [33], for a burst to occupy absolute rectangular shape, length of all the strips occupied by a burst should be equal. Let the length of all the strips occupied by burst of user are represented by L_i. Hence, if a_{jn} is the area occupied by rectangular shape of burst of user j in $n-th$ scheduling round, then a_{jn} is given by:

$$a_{jn} = L_i \times v_{jn} \tag{4.20}$$

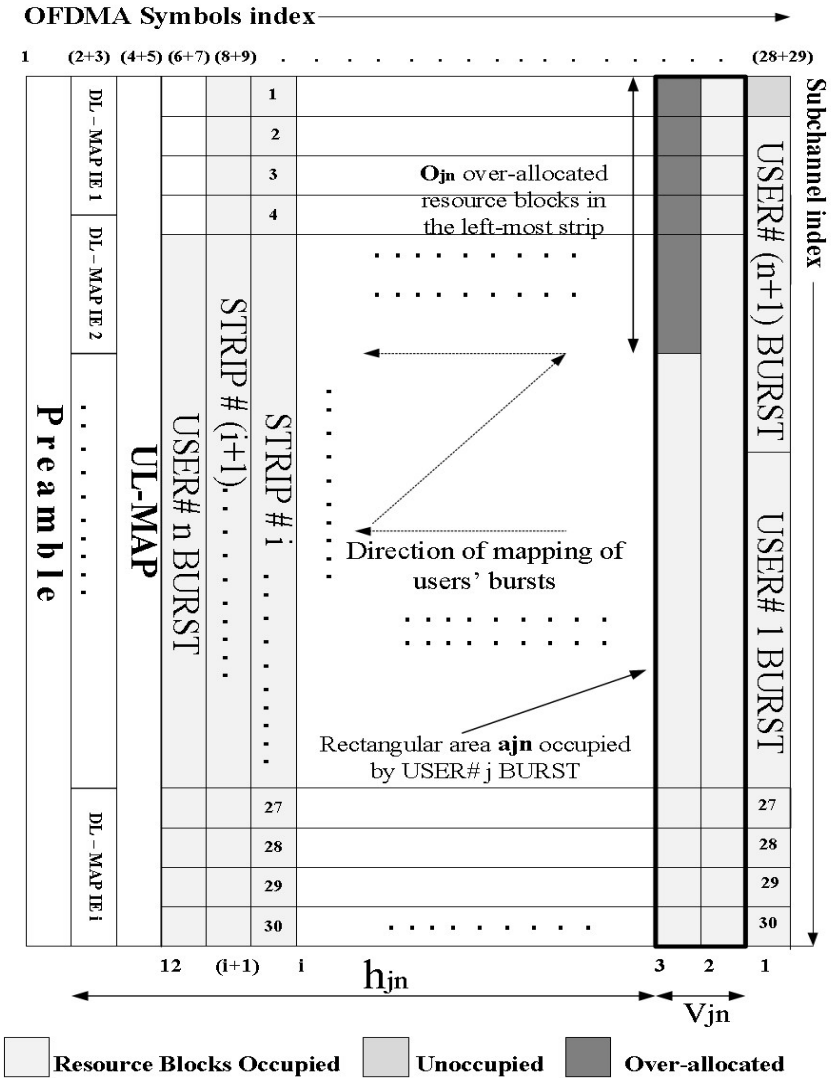


Figure 4.5: Over-allocation of resource blocks in a burst

Since L_i is measured in number of resource blocks, therefore a_{jn} also represents the number of resource blocks occupied by the absolute rectangular shape of a burst of user j in $n-th$ scheduling round. Over-allocation of resource blocks in the area occupied by the absolute rectangular shape of burst of user j in $n-th$ scheduling round, results when:

$$a_{jn} > RB_{jn} \tag{4.21}$$

Hence, over-allocated resource blocks in the area occupied by the absolute rectangular shape of a burst of user j in $n-th$ scheduling round, is given by:

$$o_{jn} = (a_{jn} - RB_{jn}) \tag{4.22}$$

where o_{jn} is the over-allocated resource blocks in the area occupied by absolute rectangular shape of burst of user j in $n-th$ scheduling round. As shown in Figure 4.5, the mapping of burst goes from right to left, hence the left-most strip out of total v_{jn} strips will consist of o_{jn} over-allocated resource blocks.

The following steps are traversed in a scheduling round of BCFP:

Step 1:

All the users are grouped in seven different groups, each corresponding to a specific modulation and coding scheme (or zone). For example, all users in zone 1 (using ½ QPSK as modulation and coding scheme) are placed in group 1; all the users in zone 2 (using ¾ QPSK) are placed in group 2 and so on.

Step 2:

Within each group, users are arranged in increasing order of average wakeup time. The average wakeup time of a user j averaged over past $(N-1)$ scheduling rounds, T_{jw} is defined as:

$$T_{jw} = \frac{\sum_{n=1}^{(N-1)}(v_{jn}+h_{jn})}{(N-1)} \tag{4.23}$$

where N^{th} scheduling round is the current scheduling round and T_{jw} is also defined in terms of number of strips on horizontal time axis.

Step 3:

The mapping of bursts of users starts from the users in the group with the lowest zone number to the group with the highest zone number i.e. bursts of all the users in group 1 are mapped in the same order as they are arranged, then burst of all the users in group 2 and so on.

For mapping burst of user j, the following steps are traversed:

Step 3.1

The downlink sub-frame is traversed from right to left and the first strip, say *strip # i,* with $L_i \neq 0$ is selected and number of strips occupied by user j in $n-th$ scheduling round, v_{jn} is incremented by 1, i.e. $v_{jn} = v_{jn} + 1$.

Step 3.2

Area occupied by v_{jn} strips, a_{jn} is computed using equation (4.20) as:

$$a_{jn} = L_i \times v_{jn}$$

If $a_{jn} \geq RB_{jn}$ *then goto Step 3.3 else goto step 3.4*

Step 3.3

Then there are enough empty resource blocks available for the burst of the user to be mapped. The burst of user is mapped from right to left and bottom to top in each of v_{jn} strips as shown in Figure 4.4 and length of each of v_{jn} strips occupied by the burst of user j, is updated to make them equal. Over-allocated resource blocks in the left-most strip, o_{jn} are computed if $v_{jn} > 1$, and then over-allocated resource blocks are computed using equation (4.22) as:

$$o_{jn} = (a_{jn} - RB_{jn})$$

For reducing o_{jn} in the left-most strip as shown in Figure 4.5, the remaining $(v_{jn} - 1)$ strips occupied by the rectangular area of burst of user j, are traversed from left to right.

With every traversed strip, a resource block is moved to the left-most strip and o_{jn} is decremented by 1, i.e. $o_{jn} = o_{jn} - 1$. Once, the right-most strip occupied by burst of user j is reached and $o_{jn} = 0$, then mapping is complete and, v_{jn} and h_{jn} for user j in $n - th$ scheduling round are recorded. If $o_{jn} \neq 0$, then again remaining $(v_{jn} - 1)$ strips except the left-most strip, are traversed from left to right and with every traversed strip a resource block is moved to the left-most strip, until $o_{jn} = 0$. Figure 4.6 shows the operation involved in moving the resource blocks to the left-most strip.

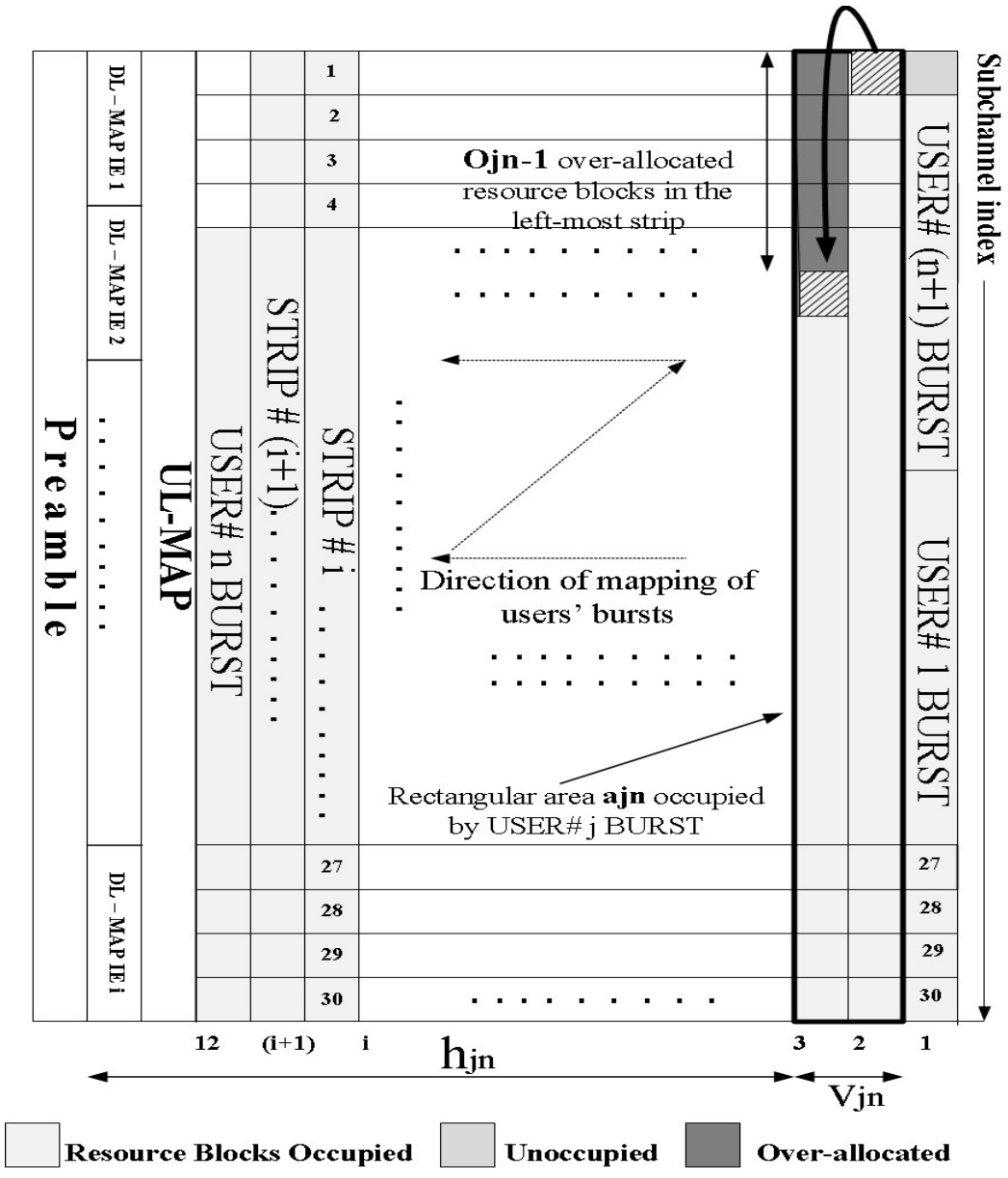


Figure 4.6: Moving resource blocks to the left-most strip in BCFP

The final result of this operation will be that the rectangular area occupied by the burst of user will decrease and leads to less or no over-allocated resource blocks. The final reduced rectangular area occupied by the burst of the user j in $n-th$ scheduling round is shown in Figure 4.7.

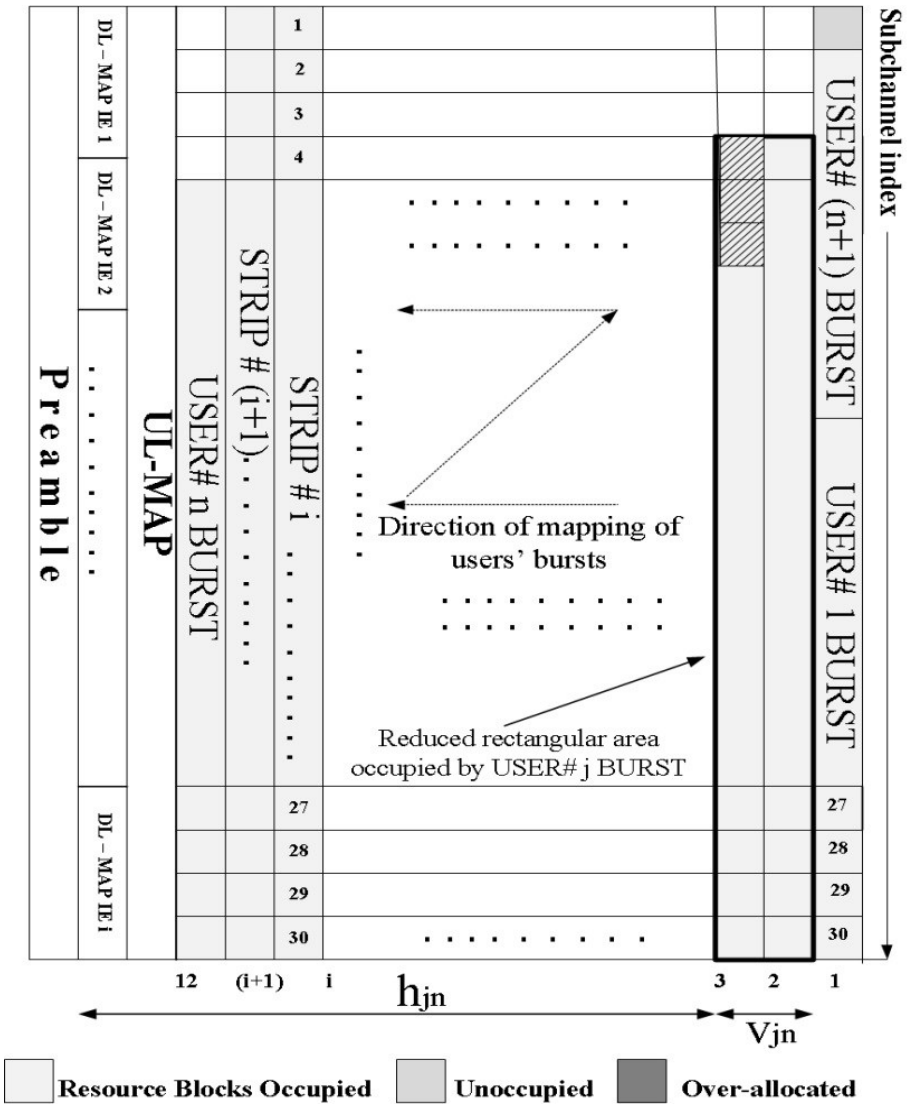


Figure 4.7: Rectangular area after moving resource blocks in BCFP

Step 3.4

- If the next consecutive strip, say *strip # (i+1)* has length, $L_{(i+1)} = 0$ or $i = 12$ i.e. we have already reached *strip # 12*, then a_{jn} is fixed and the burst of current user is

not mapped, next user j is selected for mapping until $a_{jn} \geq RB_{jn}$ and after mapping set $i = 1$, if we have already reached *strip # 12*.

ELSE

- If the next consecutive strip, say *strip # (i+1)* has length $L_{(i+1)} \neq 0$ and $i < 12$, then $i = i + 1$.

Step 4: Step 3.1 and Step 3.2 are traversed until not enough resource blocks are empty for burst of any of the user to be mapped.

Chapter 5:

Simulation Modelling & Results

5.1. Introduction

For evaluating the performance of any scheduling discipline, a proper system simulation modelling is required. A proper system simulation model involves selection of proper mathematical models for traffic generation related to different CoSs. This is the least and mandatory requirement for developing system simulation model for evaluating the performance of either wired or wireless scheduling algorithms. Furthermore, proper evaluation of wireless scheduling algorithm also demands proper modelling of wireless channels. Most of the works like Server Based Fairness Approach (SBFA) [1], Channel State dependent Wireless Fair Queuing (CS-WFQ) [4], Idealized Wireless Fair Queuing (IWFQ) [14], Channel condition Independent Fair Queuing (CIF-Q) [15], considered a two-state Markov channel model [43] to model the wireless channel as either good or bad because they were developed for 2G and 3G wireless systems. But modelling a wireless channel for 4G wireless systems requires different approach and hence proper selection of mathematical model for modelling wireless channel.

Selection of proper mathematical models for traffic generation patterns for different CoSs and wireless channel helps in simulating more realistic scenarios for evaluating the performance of wireless scheduling algorithm. Hence in the next section, we present the model and parameters used in the model for generating different types of traffic for different CoSs. An optimum wireless channel model, appropriate for creating realistic wireless channel scenario for 4G systems, is also introduced.

5.2. Traffic models for different CoSs

Six in-house simulation programs in C++ are developed to evaluate and compare the performance of joint proposed scheduling algorithm i.e. Leaky Bucket Token Bank (LBTB) and proposed burst construction algorithm i.e. Burst Construction for Fairness in Power (BCFP), in terms of queuing delay, packet loss, throughput and fairness. The performance is compared with joint Token Bank Fair Queuing (TBFQ) and BCFP as well as joint Adaptive Token Bank Fair Queuing (ATBFQ) and BCFP. Five classes of traffic are implemented in our simulation programs: CoS 1 to CoS 5. CoS 1 represents the Unsolicited Grant Service (UGS) or Constant Bit Rate (CBR). CoS 2 is an extended real time Polling Service (ertPS) representing all highly delay-sensitive applications with variable size packets. CoS 3 is real time Polling Service (rtPS) representing delay sensitive applications with variable size packets arriving at variable rates. CoS 4 is non-real time Polling Service (nrtPS), representing all delay- insensitive applications requiring certain minimum guaranteed rate to be satisfied. CoS 5 is a Best Effort (BE) involving all the applications requiring no service guarantees, either on delay or throughput.

There are broadly three constraints that define the traffic characteristics related to a specific CoS:

- **Burstiness of traffic generation**
- **Packet inter-arrival times within a burst**
- **Length of packets**

- **Burstiness of traffic generation**

Burstiness of traffic is usually defined by ON and OFF periods, where the ON period is the active period and the OFF period is silent period. During the active period, packets arrive in the form of train, whereas there is no activity in silent period. In the current simulation scenario, the Pareto distribution is used to generate traffic for different kinds of applications [45]. In this case, the random variable is ON and OFF periods. If X is a Pareto distributed random variable, then the Probability Density Function (PDF) of X is given by:

$$f_X(x) = \begin{cases} \frac{\beta \rho^{\beta}}{x^{(\beta+1)}}, & x \geq \rho \\ 0, & x < \rho \end{cases} \quad (5.1)$$

where ρ is called a location parameter and β is shape parameter. In this case, the random variable is ON period and mean ON period (T_{ON}) is set to 50 milliseconds. For the purpose of this simulation, β_{ON} (shape parameter for ON period) and β_{OFF} (shape parameter for OFF period) has been set to 1.4 and 1.2, respectively. The expected value of Pareto distributed random variable X, $E[X]$ is given by:

$$E[X] = \frac{\beta \rho}{(\beta - 1)} \quad (5.2)$$

Since, we know expected value of Pareto distributed ON period, T_{ON} and shape parameter for ON period, β_{ON}, therefore location parameter for ON period, ρ_{ON} can be determined as:

$$\rho_{ON} = \frac{T_{ON} \times (\beta_{ON} - 1)}{\beta_{ON}} \quad (5.3)$$

We can now generate random values of ON period because ρ_{ON} is known. In order to generate random values of OFF period, the expected value of OFF period, T_{OFF} and location parameter of OFF period, ρ_{OFF} need to be determined. As, we know T_{ON}, hence T_{OFF} can be determined as:

$$T_{OFF} = \left(\frac{1 - OL}{OL}\right) \times T_{ON} \quad (5.4)$$

Where OL is the source offered load. Furthermore, ρ_{OFF} can be determined similar to (5.3) as:

$$\rho_{OFF} = \frac{T_{OFF} \times (\beta_{OFF} - 1)}{\beta_{OFF}} \quad (5.5)$$

The Cumulative Distribution Function (CDF) of Pareto distribution is given by:

$$F_X(x) = \int_{-\infty}^{\infty} f_X(x)\, dx = \begin{cases} 1 - \left(\frac{\rho}{x}\right)^{\beta}, & x \geq \rho \\ 0, & x < \rho \end{cases} \quad (5.6)$$

Equation (5.6) is used to generate pareto distributed values of ON and OFF time, using inverse transform sampling. As a part of inverse transform sampling, the following

expression is solved for x in terms of continuous uniformly distributed random variable u in (0,1] :

$$F_X(x) = u \Leftrightarrow 1 - \left(\frac{\rho}{x}\right)^{\beta} = u \tag{5.7}$$

After solving equation (5.7) for x, the following relation is achieved:

$$x = \frac{\rho}{(1-u)^{1/\beta}} \tag{5.8}$$

Using equation (5.8), finally random values of ON and OFF periods can be generated with the help of uniformly distributed values of u. For a specific generated value of u by invoking subroutine of Continuous Uniform Random Distribution in a simulation environment, a specific value of ON or OFF periods can be generated.

- **Packet inter-arrival times within a burst**

Packets arrival pattern within a burst is in the form of a train of incoming packets. Although, the arrival of packets do not follow such pattern in case of UGS traffic because with UGS, packets arrive at a fixed periodic interval. Since, the packets arrive at fixed periodic interval hence, there is no ON or OFF time. This train pattern is represented in the form of inter-arrival times between the packets. In many cases, packet inter-arrival times are either constant, varying, or packets arrive in a continuous fashion that is back to back. In the current simulation scenarios, packets are considered to be arriving back to back during ON period.

- **Length of packets**

Each user has all the five CoSs. Based on G.711 standard, in the current simulation scenario, the data rate for CBR is chosen as 66 packets per second and packet length is chosen as 120 bytes, which will result in 64 kbps. The amount of CBR traffic is kept constant for all simulations. For all other CoSs, the variable length of packets is generated

and follows tri-modal distribution [44]. In such kind of distribution, three different lengths of packets: 64 bytes, 594 bytes and 1518 bytes are generated with frequency of 62%, 10% and 28% respectively [44].

5.3. Simulation Set-up

From equation (5.2), it is known that T_{OFF} depends upon source offered load, OL. In current simulation scenario, the maximum system bandwidth in downlink is set to 31 Mbps which is the peak data rate that can be achieved in downlink. So, for a specific source offered load, aggregate traffic for all CoSs of all users is generated at a rate of $(31 \times OL)\ Mbps$. Traffic generated at this rate gets divided into individual CoSs of each user. Since each user has 5 CoSs in the simulation set-up and CoS 1 is CBR, packets get generated at constant rate at constant inter-arrival time. Therefore, a fixed amount of bandwidth gets reserved for CoS 1 of each user, irrespective of any value of network offered load. In this simulation environment, network offered load is varied in increments of 10 %, starting from 10% to 100%, that is the network offered load is varied in [0.1, 1.0]. For CoS 1 of each user, 0.064 Mbps is reserved in any case of network offered load.

Furthermore, the input aggregate traffic generation rate of $(31 \times OL)\ Mbps$ gets divided into N users present in a cell coverage area. For this simulation, N has been set to 16. Hence, each user receives aggregate traffic generated at $\frac{(31 \times OL)}{N}\ Mbps$. From this point, traffic generated at individual user rate gets divided among CoS 2, CoS 3, CoS 4 and CoS 5 according to weights assigned to different CoSs such that $\sum_{i=2}^{5} W_i = 1$. Here i represents CoS i. In this simulation set-up, $W_5 = 0.35, W_4 = 0.3, W_3 = 0.2\ and\ W_2 = 0.15$. Therefore, traffic for CoS 5 is generated at a rate of 35% of $\frac{(31 \times OL)}{N}\ Mbps$. Similarly, the traffic for other CoSs of a user is generated at following rates:

$$Traffic\ Generation\ rate\ for\ CoS\ 1 = 0.064\ Mbps \tag{5.9}$$

$$Traffic\ Generation\ rate\ for\ CoS\ 2 = \frac{W_2(31 \times OL - 0.064)}{N}\ Mbps \tag{5.10}$$

$$Traffic\ Generation\ rate\ for\ CoS\ 3 = \frac{W_3(31 \times OL - 0.064)}{N}\ Mbps \tag{5.11}$$

$$Traffic\ Generation\ rate\ for\ CoS\ 4 = \frac{W_4(31\times OL-0.064)}{N}\ Mbps \tag{5.12}$$

$$Traffic\ Generation\ rate\ for\ CoS\ 5 = \frac{W_5(31\times OL-0.064)}{N}\ Mbps \tag{5.13}$$

Simulation set-up is done for evaluating the performance of scheduling disciplines in a Mobile WiMAX radio interface, where users are mobile in true sense. Since, users are mobile in true sense, the distance of users with respect to base stations changes very frequently almost in every scheduling round. The maximum coverage area of a Mobile WiMAX base station is 5 Kilometres [35]. As, coverage area of Mobile WiMAX base station is divided into seven zones, therefore to make the clear boundary between different zones, the maximum coverage area of a cell is assumed as 7 Kilometres for this simulation scenario. The reported average SINR by users is inversely proportional to the distance between base station and users. The relationship between average SINR and transmitting distance is as follows [35]:

$$S_{avg} = \begin{cases} 19\ dB,\ 0 \le D \le 1\ Kilometre \\ 17\ dB,\ 1 < D \le 2\ Kilometres \\ 14\ dB,\ 2 < D \le 3\ Kilometres \\ 11\ dB,\ 3 < D \le 4\ Kilometres \\ 8\ dB,\ 4 < D \le 5\ Kilometres \\ 5\ dB,\ 5 < D \le 6\ Kilometres \\ 1\ dB,\ 6 < D \le 7\ Kilometres \end{cases} \tag{5.14}$$

Where S_{avg} is average SINR and D is the distance between the base station and a given user. The reported instantaneous SINR which is a random variable here, follows a Rayleigh distribution in this case because instantaneous SINR is the result of multipath fading where the received signal is the superposition of several reflected multipath components [31]. Therefore, the PDF of Rayleigh distributed instantaneous SINR, S is given by:

$$f_S(s) = \begin{cases} \frac{s}{{S_{avg}}^2} \times \exp\left(\frac{-s^2}{2\times {S_{avg}}^2}\right), & s \ge 0 \\ 0, & s < 0 \end{cases} \tag{5.15}$$

The CDF of Rayleigh distribution is given by:

$$F_S(s) = \int_{-\infty}^{\infty} f_S(s)\ ds = \ 1 - \exp\left(\frac{-s^2}{2\times {S_{avg}}^2}\right),\ \ s \ge 0 \tag{5.16}$$

Where S is the instantaneous SINR reported by user in a scheduling round. The instantaneous SINR is generated randomly by following the method of inverse transform sampling and using equation (5.16) to generate random values of S. Therefore, S will be expressed in terms of continuous uniformly distributed random variable, u in [0,1). For every uniformly distributed value of u generated by invoking the subroutine of Continuous Uniform Random Distribution, the instantaneous value of SINR is obtained from following relation:

$$1 - \exp\left(\frac{-s^2}{2 \times {S_{avg}}^2}\right) = u \quad \Leftrightarrow \quad s = \sqrt[S_{avg}]{2 \ln \frac{1}{(1-u)}} \tag{5.17}$$

The distance between base station and user is randomly distributed in every scheduling round, in the simulation environment, between (0,7] Kilometres.

5.4. Simulation Analysis of the LBTB joined with BCFP

In order to do the analysis of joint proposed scheduling and burst construction algorithms i.e. LBTB+BCFP, the performance of LBTB+BCFP is compared with joint Token Bank Fair Queuing (TBFQ) and BCFP as well as joint Adaptive Token Bank Fair Queuing (ATBFQ) and BCFP. The four cases for which results are plotted, are:

- When LBTB is joined with BCFP i.e. LBTB+BCFP.
- When TBFQ is joined with BCFP i.e. TBFQ+BCFP.
- When ATBFQ is joined with BCFP i.e. ATBFQ+BCFP
- When LBTB is joined with Round Robin (RR) burst construction algorithm i.e. LBTB+RR.

The performance of the four cases is evaluated in the scenarios of low as well as high loading conditions. The low loading conditions being less or equal to 50% offered load, whereas the high loading conditions being more than 50% offered load.

5.4.1. Throughput

The average cell throughput is an important criterion for evaluating the performance of a scheduling algorithm. It is a very good indicator of how much efficient use of wireless channel an algorithm is able to achieve.

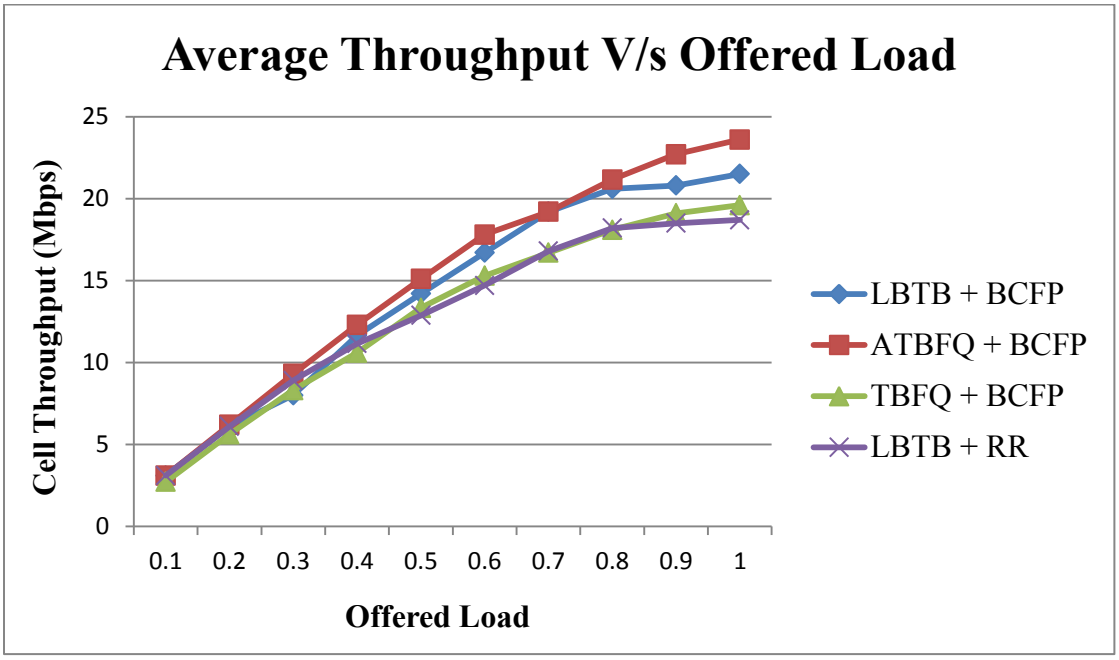


Figure 5.1: Average Cell Throughput for different network loadings

It can be observed from Figure 5.1 that the cell throughput increases as the loading on network increases for all the four cases. As the maximum cell throughput in downlink is 31 Mbps, it can be observed that only ATBFQ + BCFP and LBTB + BCFP are able to exceed cell throughput more than 20 Mbps. Although the cell throughput provided by ATBFQ + BCFP and LBTB + BCFP is not 31 Mbps but it can be observed that they provide the highest cell throughput among all the four cases. This means that they have the highest wireless channel utilization. Among ATBFQ+BCFP and LBTB+BCFP, ATBFQ+BCFP has the highest cell throughput because ATBFQ assumes that wireless channel capacity is constant or constant number of tokens is generated in a system in a scheduling round. This results in highest cell throughput. On the contrary, LBTB serves users under the assumption of varying channel capacity in every scheduling round or varying number of tokens is generated in a system in a scheduling round. Hence, sometimes the channel capacity can go extremely low, which results in less channel throughput.

5.4.2. Average Packet Delay

The average Packet Delay is defined as the difference between the time instant when the packet enters the queue and the time instant when the packet gets transmitted. Average packet delay experienced by a packet belonging to CoS 1, CoS 2 and CoS 3 is shown in this section. It is also an important parameter which evaluates the performance of a scheduling algorithm in terms of bounds on maximum and minimum delay experienced by a packet belonging to a specific CoS in a network.

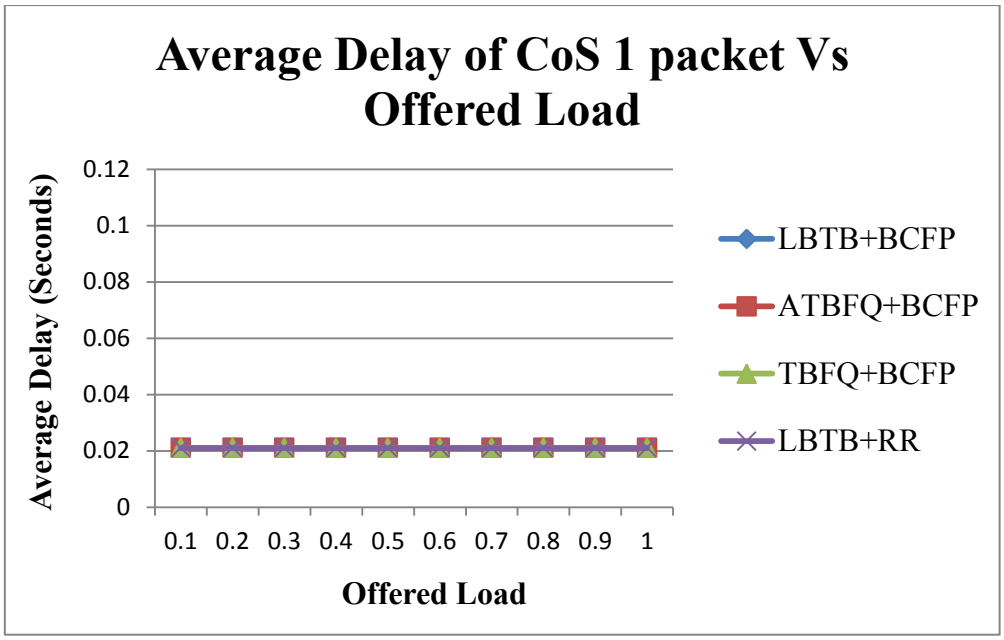


Figure 5.2: Average packet Delay of CoS 1 for different network loadings

Figures 5.2, 5.3, and 5.4 show the average packet delay for CoS 1, CoS2, and CoS 3, respectively. As observed from Figure 5.2, since CoS 1 is CBR service hence the average delay experienced by a packet for this CoS during different network loading conditions for different combinations is constant.

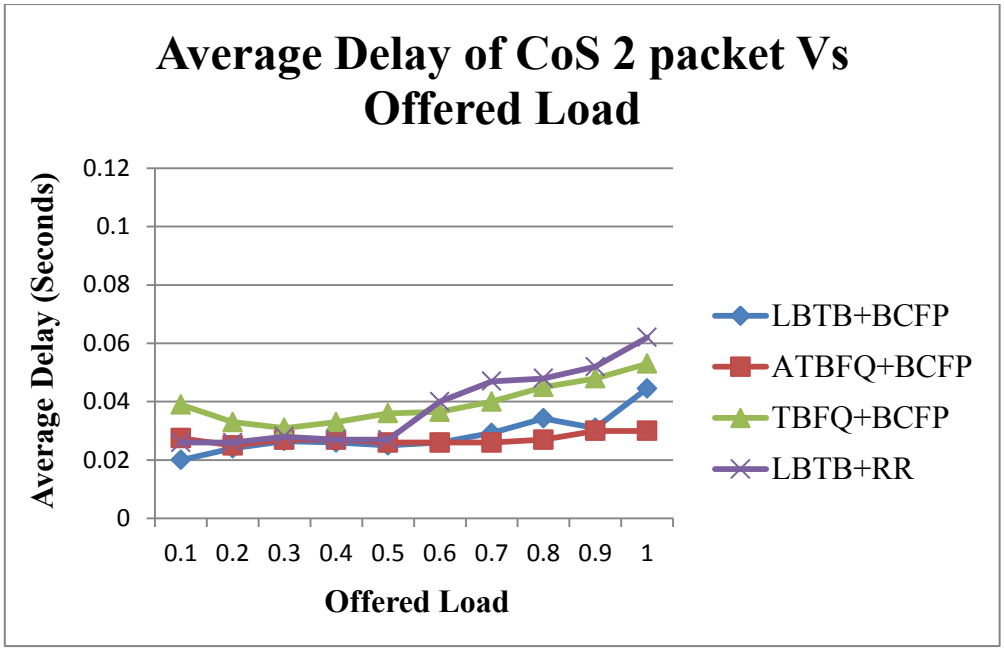


Figure 5.3: Average packet Delay of CoS 2 for different network loadings

As observed from Figure 5.3, when offered load is less than 30 %, LBTB+BCFP clearly outperforms ATBFQ+BCFP in terms of CoS 2 average packet delay. It is also observed that till the offered load of 70 %, the average delay experienced by a packet of CoS 2 for LBTB+BCFP mimics the average delay for ATBFQ+BCFP. It is known that ATBFQ serves the users when the wireless channel capacity is constant, on the other hand LBTB serves the users when they are highly mobile. Therefore, it can be said that till 70% offered load, the delay performance of users served by LBTB mimics the delay performance of users in ideal conditions i.e. when the channel capacity is constant. On the other side, LBTB+BCFP clearly outperforms TBFQ+BCFP for low as well as high loading conditions. This clarifies the fact that TBFQ is not suitable for 4G wireless networks like Mobile WiMAX because of bad delay performance of users. Furthermore, LBTB+BCFP also clearly outperforms LBTB+RR. This means that LBTB+BCFP clearly outperforms all the other cases for both low as well as high network loadings, except for offered load greater than 70% in ATBFQ+BCFP.

It can be observed from Figure 5.3 that beyond the offered load of 70 %, ATBFQ+BCFP clearly outperforms LBTB+BCFP and all other cases. This is because ATBFQ first maps the scheduled data of a user on resource matrix and then schedules the data of another user. In this way, ATBFQ considers the number of radio resources left in

radio resource matrix while scheduling the data of a user, therefore it is able to serve all the backlogged users in a scheduling round. On the other hand, LBTB first schedules the data of every user and then maps it on the radio resource matrix, hence it is not able to map the data of each backlogged user on radio resource matrix, in a scheduling round. This results in increased average delay of backlogged users. Same reasoning accounts for the behaviour exhibited by different combinations for the average delay experienced by packet of CoS 3 as shown in Figure 5.4. As shown in Figure 5.4, LBTB+BCFP mimics and even surpasses ATBFQ+BCFP in terms of CoS 3 average packet delay till the offered load of 80 %. On the other hand, ATBFQ+BCFP outperforms LBTB+BCFP for more than 80% offered load, whereas these two combinations outperform the other two combinations: TBFQ+BCFP as well as LBTB+RR for different network loadings.

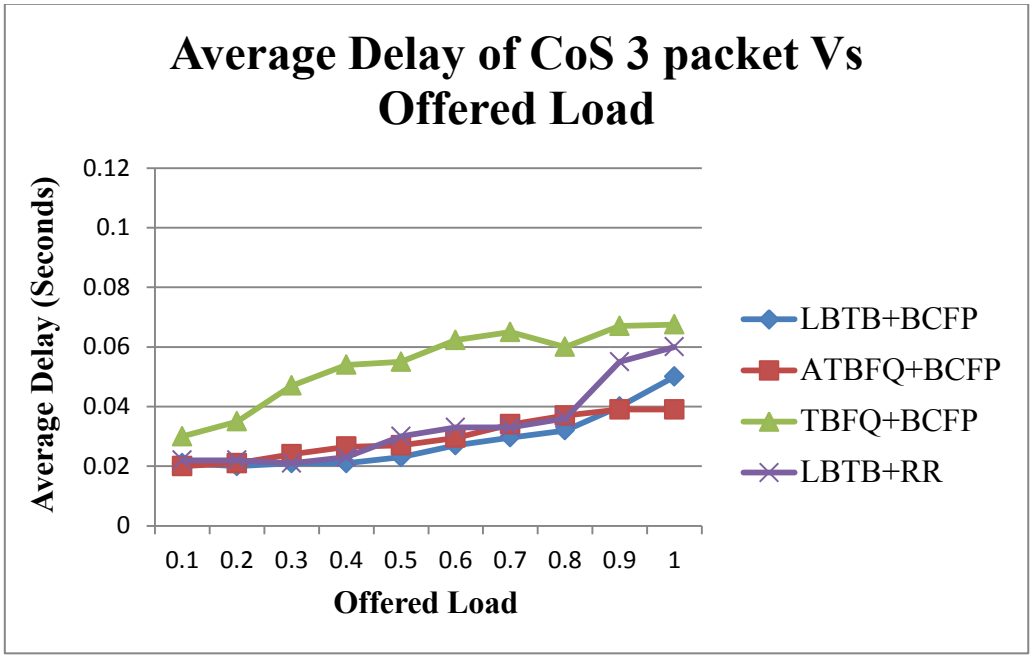


Figure 5.4: Average packet Delay of CoS 3 for different network loadings

5.4.3. Fraction of packets transmitted for varying distances

Figure 5.5 shows the fraction of total packets transmitted for users with varying distances from base station at 40% offered load. The fraction of packets transmitted for all the users when they are in a zone number z, f_z is given by:

$$f_z = \frac{total\ packets\ transmitted\ for\ all\ users\ in\ zone\ number\ z}{total\ packets\ transmitted\ for\ all\ users\ in\ all\ zones}$$

The combination of LBTB+BCFP and LBTB+RR in this case clearly outperforms TBFQ+BCFP for far distances of users from base station but lag slightly behind TBFQ+BCFP as well as ATBFQ+BCFP in case of users being near to base station. This is because LBTB distributes more bandwidth to the users with relatively far distances from base station. Even in the worst channel conditions when the user is at farthest distance from base station, LBTB+BCFP and LBTB+RR are both able to achieve almost 90% packets transmission.

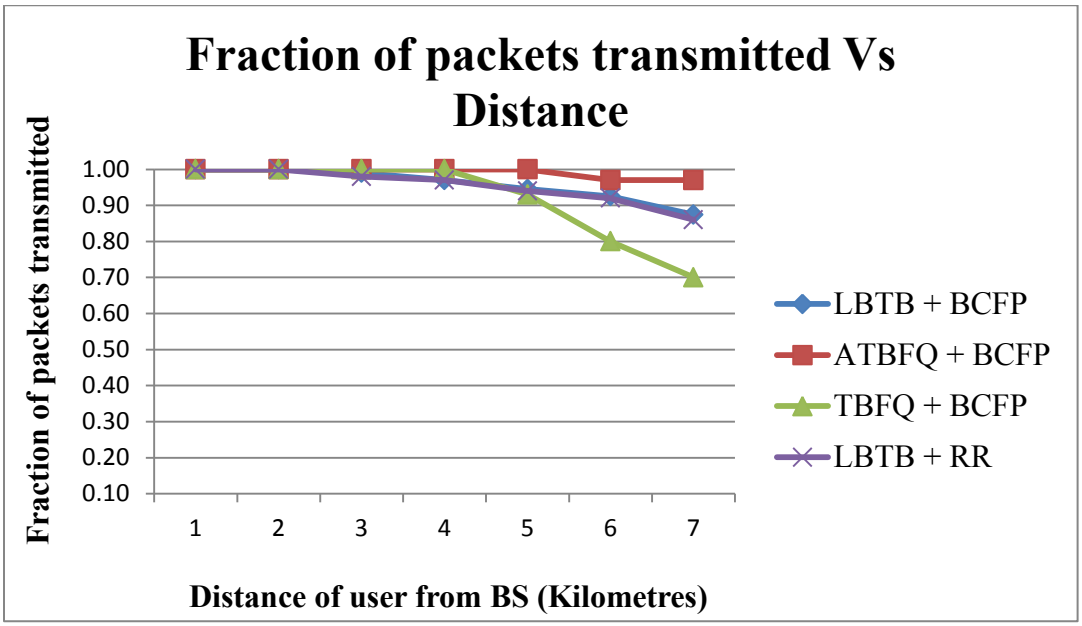


Figure 5.5: Fraction of packets transmitted at 40% offered load

Figure 5.6 shows the fraction of total packets transmitted for users with varying distances from base station at 90% offered load. It can be observed that the performance of LBTB+BCFP and LBTB+RR deteriorates. Even for users in close proximity to the base station, both the combinations lag behind TBFQ+BCFP as well as ATBFQ+BCFP. As opposed to 100% packet transmissions for users near to base station, in case of ATBFQ+BCFP and TBFQ+BCFP, the other two combinations manage to achieve almost 90% packet transmissions. The performance continues to deteriorate with the increasing distance of users from base station. Furthermore, for farthest distance of users from base

station, both LBTB+BCFP and LBTB+RR outperform TBFQ+BCFP by nearly 20 % and 10%, respectively.

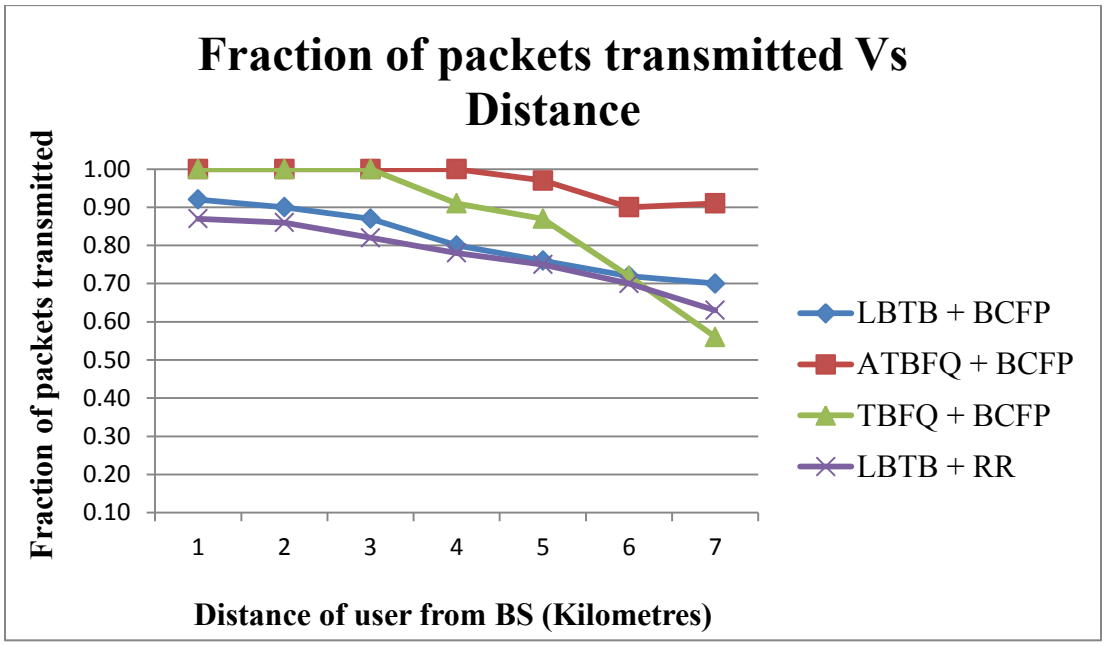


Figure 5.6: Fraction of packets transmitted at 90% offered load

5.4.4. Maximum Packet Transmission Disparity

In context of fraction of packets transmitted for all users in different zones, a fair scheduling algorithm should achieve almost equal fraction of packets transmitted for different zones. The difference between the fraction of packets transmitted for any two zones is called packet transmission disparity. The maximum disparity between the fraction of packets transmitted for any two zones is bounded by maximum packet transmission disparity. The maximum packet transmission disparity is defined as the difference between minimum fraction of packets transmitted for a zone and maximum packets transmitted for another zone. The maximum packet transmission disparity is the upper bound on the packet transmission disparity between any two zones. In the current scenario, the maximum packet transmission disparity will be the difference between the fraction of packets transmitted for zone 7 (maximum value) and fraction of packets transmitted for zone 1 (minimum value). Maximum packet transmission disparity, TD can be described as:

$$TD = f_7 - f_1$$

Where f_7 is the fraction of packets transmitted for zone 7, whereas f_1 is the fraction of packets transmitted for zone 1. Therefore, tighter or smaller value of bound signifies that a scheduling algorithm is more fair to the users in different zones as the maximum packet transmission disparity is smaller.

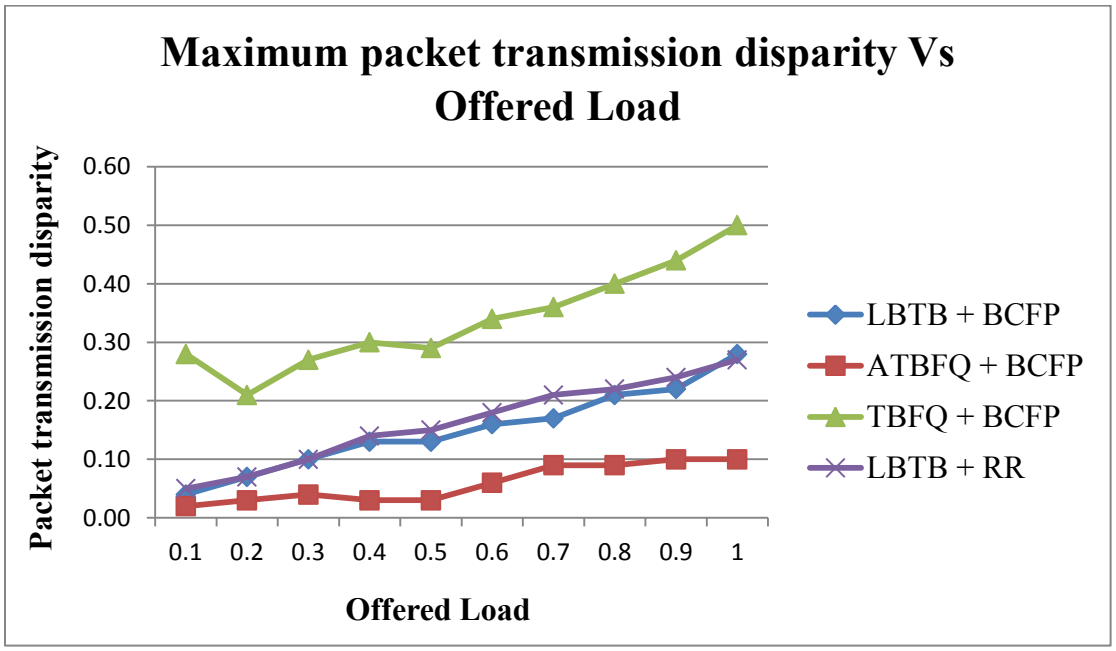


Figure 5.7: Packet transmission disparity at varying network loading conditions

Figure 5.7 depicts the difference between maximum and minimum values of fraction of packet transmissions for best and worst channel conditions for all the four combinations. Furthermore, Figure 5.7 shows that ATBFQ+BCFP is most fair among all the four combinations, having maximum transmission disparity of only 10% at high loading conditions. On the other hand, TBFQ+BCFP is least fair, having a very high maximum packet transmission disparity of almost 52% at 100% offered load. Furthermore, both combinations of LBTB clearly outperform TBFQ+BCFP for all values of offered load, whereas both combinations of LBTB lag behind ATBFQ+BCFP in this respect. Therefore, both combinations of LBTB are more fair than TBFQ+BCFP, whereas less fair than ATBFQ+BCFP.

Furthermore, both the combinations of LBTB achieve almost maximum packet transmission disparity of 28% at 100% offered load, which is considerably less than 52% maximum packet transmission disparity of TBFQ+BCFP at 100% offered load.

5.4.5. Fraction of packets dropped

Figure 5.8 shows the fraction of packets dropped for different loading conditions. It can be observed that ATBFQ+BCFP clearly outperforms all the other combinations whereas LBTB+BCFP clearly outperforms its other counterpart as well as TBFQ+BCFP. The reason for TBFQ+BCFP showing the highest fraction of packets dropped is that TBFQ defers the packet transmissions of users experiencing bad channel conditions, therefore it results in heavy backlog for flows experiencing bad channel conditions. This means high fraction of packets being dropped. On the other hand, since ATBFQ is able to serve maximum number of backlogged flows in a scheduling round because it considers the amount of radio resources remaining, every time it schedules the data of a flow. Therefore, heavy backlog does not produces in ATBFQ, hence least fraction of packets get dropped. LBTB first schedules the data of all the backlogged flows and then maps it one by one, therefore sometimes, LBTB is not able to map the scheduled data of all flows. Hence, the packet dropping ratios achieved by both combinations of LBTB are higher than that of ATBFQ+BCFP. Furthermore, since LBTB does not defer the transmission of flows experiencing relatively bad channel conditions because it exploits the link adaptation in Mobile WiMAX, hence it's packet dropping ratio is less than that of TBFQ+BCFP.

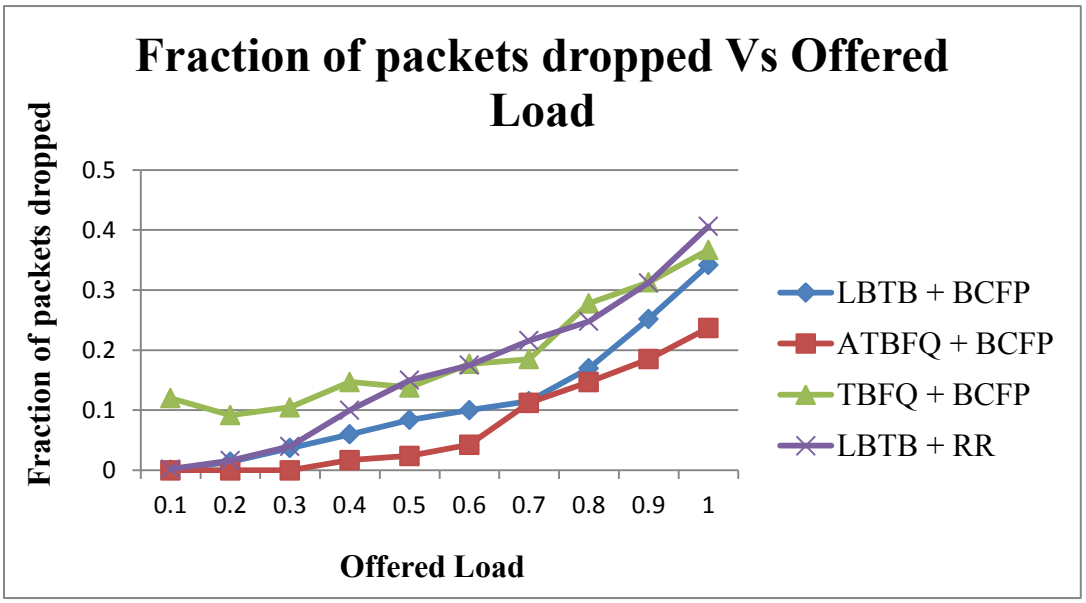


Figure 5.8: Fraction of packets dropped at different network loadings

5.4.6. Fairness

The notion of fairness in wireless networks refers to short-term fairness. According to the definition of short-term fairness, a scheduling discipline is short-term fair if the difference between the service received (number of bytes transmitted) by a pair of flows during a scheduling round (frame duration) of a scheduling discipline, is bounded. The difference between the service received by a pair of flow is called relative fairness index and it is upper bounded by a maximum value called relative fairness bound. In realistic scenarios, the difference between the service received by every other pair of flows will be different and can take any value, which does not tell anything quantitative about how much fair a scheduling discipline is.

Therefore, Jain's fairness index [22] is used to compute the short-term fairness of a scheduling discipline in quantitative sense and tells about how much fair the scheduling discipline is. If a system allocates resources to n contending users such that user i receives allocation x_i in a scheduling round, then Jain's fairness index, FI will be:

$$FI = \frac{(\sum_{i=1}^{n} x_i)^2}{n \times \sum_{1=1}^{n} {x_i}^2}$$

Where FI will always lie in (0,1). Since, relative fairness index considers fairness metric as the service received by flows in a scheduling round, therefore here x_i will be the service received by a flow i in a scheduling round. Since Jain's fairness index tells about how much fair the scheduling algorithm is, no matter which fairness metric is selected, therefore it is necessary to normalize the fairness metric. Hence, x_i will be normalized as [23]:

$$x_i = \frac{Number\ of\ bytes\ transmitted\ in\ a\ scheduling\ round}{Number\ of\ bytes\ in\ queue\ of\ flow\ i}$$

In all the C++ simulations developed for this thesis, fairness index is measured in every frame interval and is averaged over the simulation time of 3000 seconds. The snapshot of fairness index is taken at every 300 seconds of simulation time.

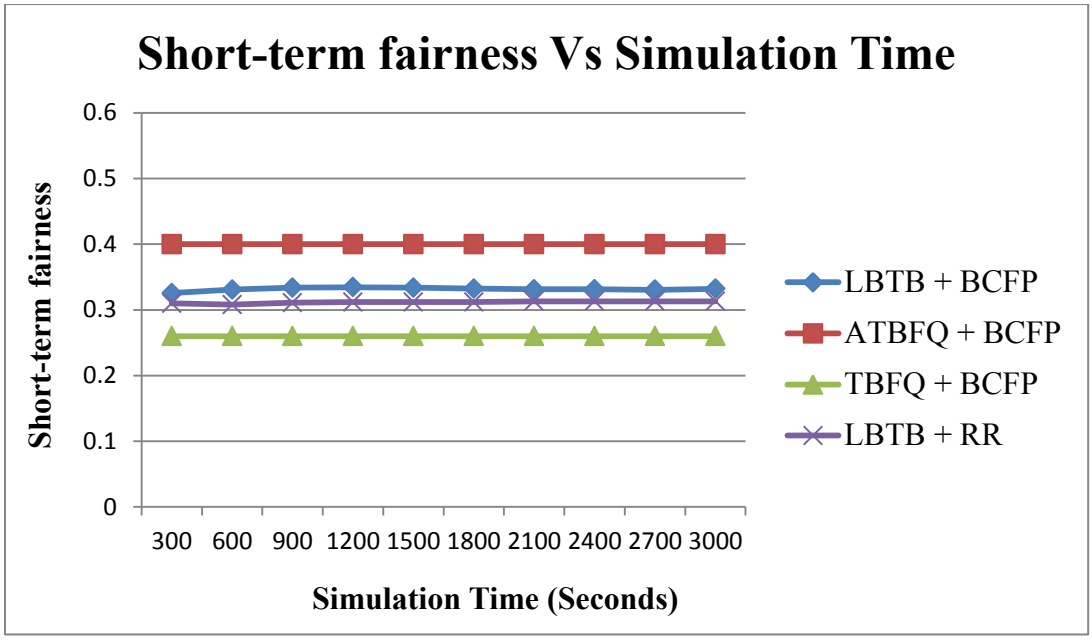


Figure 5.9: Snapshot of short term fairness at every 300 seconds

As it can be clearly observed from Figure 5.9, both combinations of LBTB outperform TBFQ+BCFP. While TBFQ+BCFP treats 25% of the users fairly, both combinations of LBTB outperform TBFQ+BCFP by nearly 10%. On the other hand, there is not much difference between two different combinations of LBTB, whereas ATBFQ+BCFP treats approximately 40% of the users fairly in a scheduling round.

5.4.7. Wastage of Physical Radio Resources

The wastage of physical radio resources is an important constraint to be considered because in OFDMA systems, these resources are limited and hence should be minimally wasted. Usually, wastage of radio resources in such kind of systems is evaluated by measuring two different kinds of metrics: Unoccupied resource blocks and over-allocated resource blocks.

- **Unoccupied Resource Blocks**

Unoccupied Resource Blocks (RBs) are the fraction of RBs, which are left-over because scheduled data of a user cannot be completely mapped on the left-over RBs. Figure 5.10 shows the fraction of unoccupied RBs for BCFP working in conjunction with different

scheduling algorithms. It can be observed that when BCFP works in conjunction with LBTB, then it outperforms all the other combinations, achieving least fraction of unoccupied RBs. During high loading conditions, LBTB+BCFP as well as LBTB+RR almost achieve 30% wastage in RBs and the performance remains consistent. On the other hand, both TBFQ+BCFP as well as ATBFQ+BCFP show highly fluctuating behaviour in unoccupied RBs.

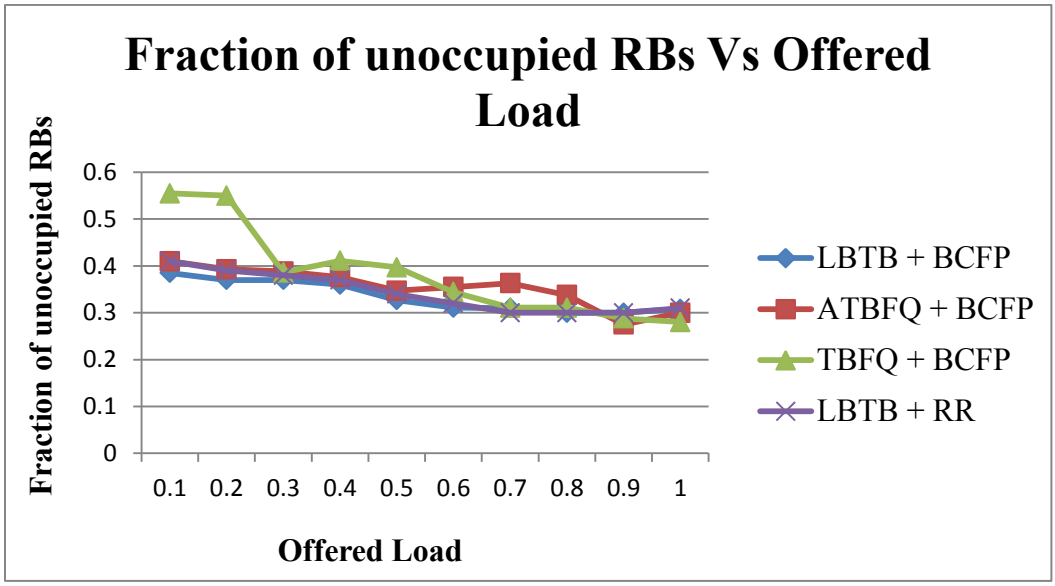


Figure 5.10: Fraction of unoccupied RBs for varying network loading conditions

- **Over-allocated Resource Blocks**

Over-allocated Resource Blocks (RBs) are the fraction of RBs which get wasted when the absolute rectangular area occupied by burst of a user consists of more resource blocks than required to map the scheduled data of user. Since the area occupied by the burst of a user has to be absolute rectangular, the over-allocated RBs cannot be occupied by any other user. Figure 5.11 shows the fraction of over-allocated RBs at varying network loading conditions for BCFP, working in conjunction with different scheduling algorithms. In most of the network loading scenarios, over-allocation is less than 10% and is achieved for all the combinations. On the other hand, during most of the network loading scenarios, ATBFQ+BCFP demonstrates the least fraction of over allocation. Both combinations,

LBTB+BCFP as well as TBFQ+BCFP, show slight fluctuating behaviour during high loading conditions, leading to over allocation of more than 10%. This means that the operation involving moving RBs to left-most time symbol column of a burst in BCFP, in the event of over-allocated RBs is able to keep the over-allocation to the minimum. On the other hand, when the round robin burst construction algorithm works in conjunction with LBTB, then the over-allocation remains constant for all network loading scenarios

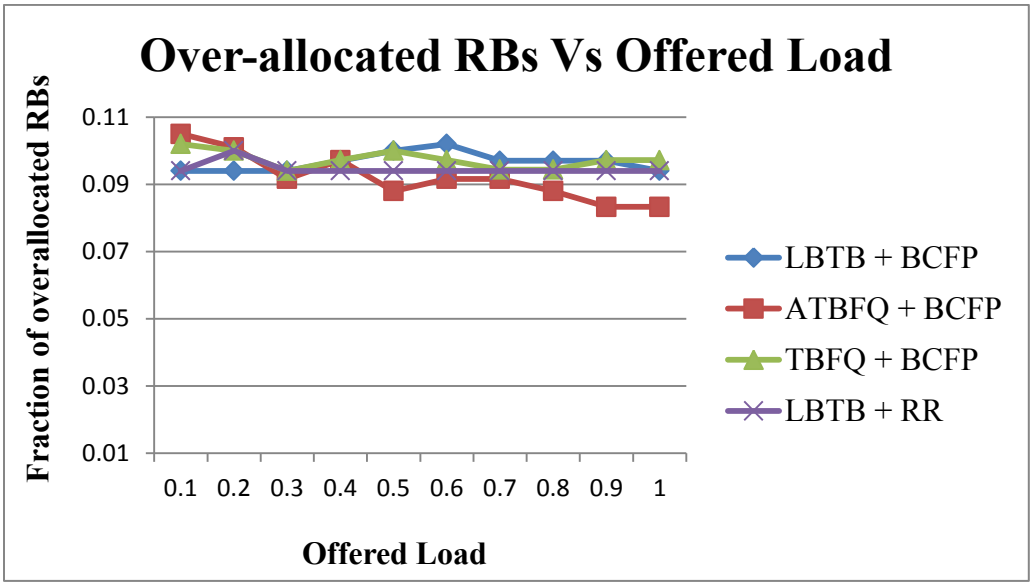


Figure 5.11: Fraction of over allocated RBs at varying network loading conditions

5.4.8. Fairness in average wake-up time (Power Consumption)

Since in all the C++ simulations developed for this thesis, it is assumed that Connection Identifier (CID) of a user is not included in the DL-MAP message of user, the wakeup time of user starts from start of DL-MAP to the end of user's burst. BCFP tries to achieve fairness in the wakeup time when CID of user is not included in DL-MAP. Figure 5.12 shows the fairness in average wakeup times of 16 users for the four combinations. Since the average wake-up time is always in the integral multiples of OFDMA time symbols in all the C++ simulations developed, the average wake-up time is rounded off to the nearest integer. The average wakeup time is 2 time symbol columns only for LBTB+BCFP, whereas for all other combinations, the average wakeup time is 1 time symbol columns. Although, in all the cases, 100% fairness in average wakeup times of 16

users is achieved, but the 16 users experience more average wakeup (average power consumption) in case of LBTB+BCFP.

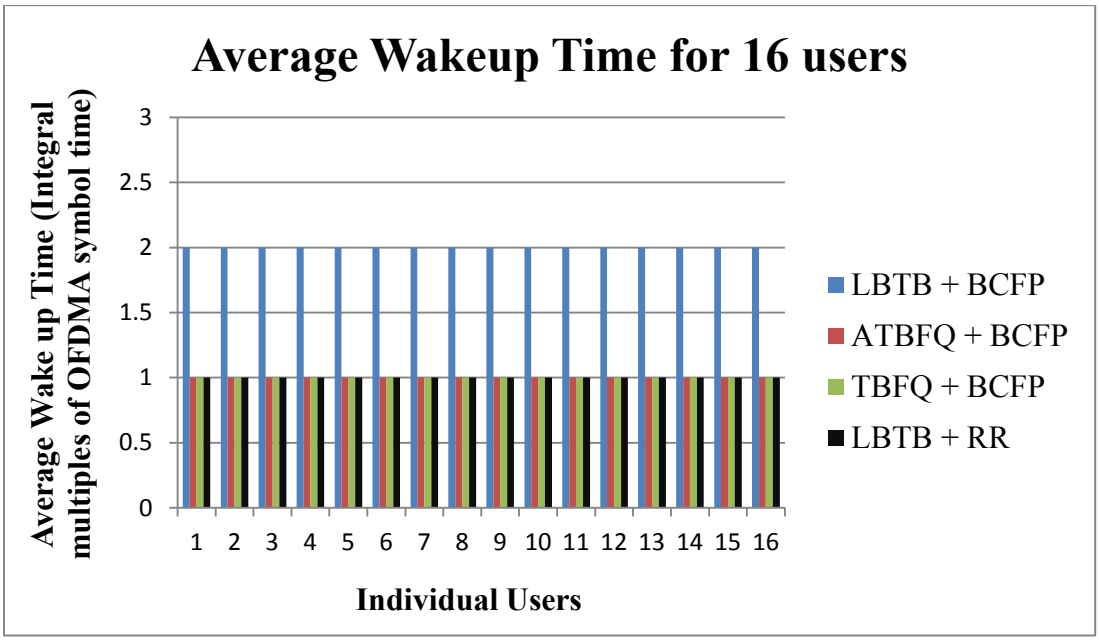


Figure 5.12: Average wake-up time for 16 users

5.5. Results Analysis for proposed burst construction algorithm

For the purpose of evaluating the performance of BCFP algorithm alone, out of total six in-house developed C++ simulations, two in-house developed C++ simulations are used to compare the performance of BCFP alone with round robin burst construction algorithm. In these simulations, the two burst construction algorithms do not work in conjunction with any scheduling algorithms. In the current simulations scenario, scheduled bandwidth of 25 users is generated randomly between their minimum guaranteed rate and peak rate, in every scheduling round. During each scheduling round, BCFP and round robin burst construction algorithms are not able to map the data of all users. Therefore in this case, the performance of BCFP and round robin burst construction algorithms is evaluated in terms of average wakeup times, unoccupied Resource Blocks (RBs) and Over-allocated RBs, for different number of users for which the data is mapped on the resource matrix, in every scheduling round.

Figure 5.13 shows the fraction of unoccupied Resource Blocks (RBs) that are wasted when data of 25 users is mapped using both BCFP as well as round robin burst construction algorithm. It can be observed that BCFP clearly outperforms round robin burst construction algorithm for different number of mapped users. The minimum fraction of unoccupied RBs, which a round robin algorithm can achieve is almost 17%, whereas the minimum fraction of unoccupied RBs achieved by BCFP is even less than 10% and remains consistent for different number of mapped users. On the other hand, the maximum fraction of unoccupied RBs when users are mapped using round robin algorithm, is almost 40 %, which is a lot. The results also clarify that as more number of users are present in the system, more scheduled data is available for mapping and hence less unoccupied resource blocks get wasted.

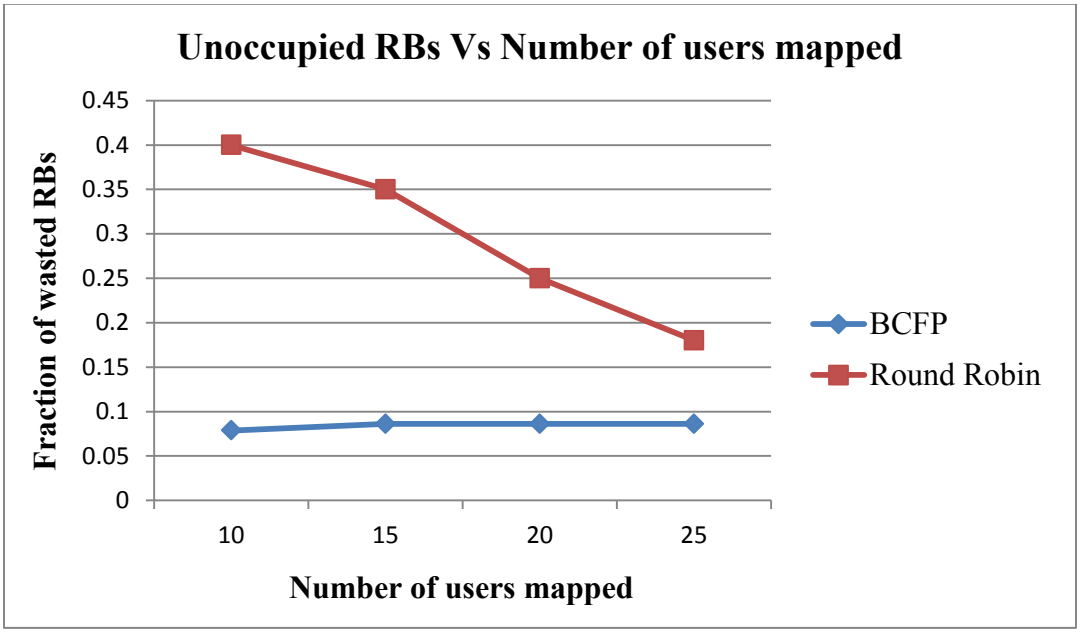


Figure 5.13: Fraction of unoccupied RBs for different number of users packed

Figure 5.14 shows the fraction of over-allocated RBs that are wasted when data of 25 users is mapped using both BCFP as well as round robin burst construction algorithms. It can be observed that BCFP clearly outperforms round robin burst construction algorithm for different number of mapped users. The minimum fraction of over-allocated RBs, which a round robin algorithm can achieve is almost 5.5%, whereas the minimum fraction of over-allocated RBs achieved by BCFP is even less than 2% and increases with increased number of mapped users. This clarifies that as the number of mapped users increase, the absolute

rectangular bursts to be mapped also increase, which results in increased fraction of over-allocated RBs. On the other hand, round robin algorithm achieves 9% and BCFP achieves 5.5% over-allocated RBs.

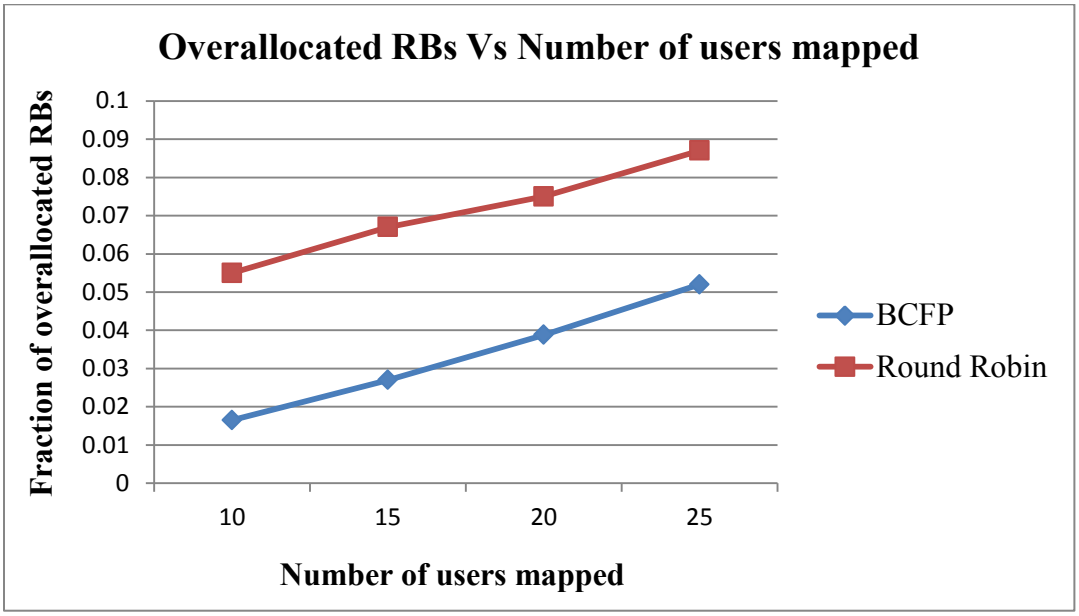


Figure 5.14: Fraction of over allocated RBs for different number of users packed

Figure 5.15 shows the fairness in average wakeup times for 25 users for BCFP as well as round robin burst construction algorithms. The average wakeup time for all the users in BCFP is 2 time symbol columns, which is less than the average wakeup time for all the users in round robin algorithm, i.e. 4 time symbol columns. The 100% fairness in average wakeup times for all users in both BCFP as well as round robin algorithms can be observed in figure 5.15.

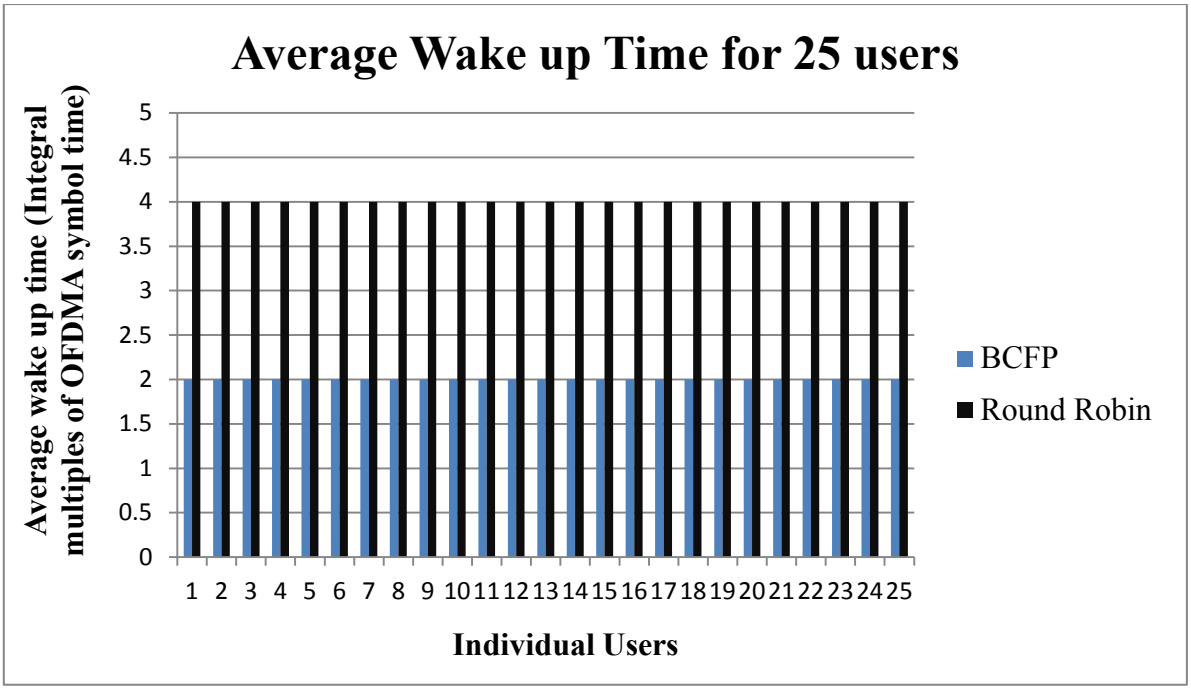


Figure 5.15: Average wake-up time for 25 users

Chapter 6:

Conclusions and Future work proposals

6.1 Conclusions

In this thesis, a 4G wireless network scheduling algorithm called Leaky Bucket Token Bank (LBTB) is proposed for Mobile WiMAX networks. Furthermore, a burst construction algorithm called Burst Construction for Fairness in Power (BCFP) is also proposed to pack the scheduled data into bursts and map the bursts to OFDMA resource matrix. The performance of the joint proposed algorithms is compared with the other scheduling algorithms called Token Bank Fair Queuing (TBFQ) and Adaptive Token Bank Fair Queuing (ATBFQ) each used jointly together with the BCFP burst construction algorithm i.e. ATBFQ+BCFP and TBFQ+BCFP. The performance is compared in terms of average cell throughput, average queuing delay, packet dropping ratio, fraction of packets transmitted for varying distances of all users from base station and short-term fairness. The LBTB scheduling algorithm has taken all the necessary enhancements made in the physical layer of Mobile WiMAX into consideration.

It has been shown that LBTB+BCFP achieves quite high cell throughput whereas on the other hand, it provides acceptable QoS to the cell edge users. This proves that LBTB+BCFP treats users with varying distances from the base station, more fairly. This has also been confirmed by simulation results. Although LBTB+BCFP either mimics or lags behind ATBFQ+BCFP in terms of performance in these aspects it works under the environment of highly variable capacity which is considered to be the real wireless environment.

Simulation results prove that LBTB+BCFP either outperforms or mimics the 4G benchmark combination ATBFQ+BCFP in terms of delay performance of highly delay sensitive applications. Therefore, LBTB+BCFP provides best QoS to different class of services.

It has been also proved with the help of simulation results that LBTB+BCFP treats 35% of the users fairly, whereas ATBFQ+BCFP treats 40% of the users fairly. The fraction of fairly treated users accounts for the fair distribution bandwidth. Therefore, LBTB+BCFP is able to distribute bandwidth fairly among 35% of the users, whereas this fraction is 40% in case of ATBFQ+BCFP.

In all the aspects, TBFQ+BCFP lags behind the two mentioned combinations. This also verifies the theory that TBFQ being the scheduling algorithm developed for 3G networks is not suitable for 4G networks. Since, all the three scheduling algorithms work in conjunction with BCFP hence it is difficult to comment on the effect of BCFP on the whole combination. Therefore, it is necessary to cascade another burst construction algorithm with one of the scheduling algorithms so that the impact on over-all performance can be observed and whether that impact degrades the performance or not.

Since, one of the prime goals of BCFP is to achieve fairness in average power consumption of a mobile station therefore; another well-known fair algorithm called Round Robin (RR) is used as a burst construction algorithm. In this algorithm, the fairness criteria is selected to be average power consumption. Hence, the round robin algorithm is cascaded with the LBTB i.e. LBTB+RR. Simulation results prove that LBTB+RR shows degrading performance in terms of different aspects. This verifies the theory that BCFP is working in synchronization with LBTB and changing the burst construction algorithm definitely affects the performance of whole combination. Furthermore, all the four combinations perform similar in terms of wastage of radio resources. Therefore, both the burst construction algorithms are independently executed in an environment where they are not cascaded with any of the scheduling algorithms. This reveals the true performance of both burst construction algorithms.

It has been proved that although, BCFP as well as RR result in fairness in average power consumption but average power consumption is more in RR whereas, BCFP clearly outperforms RR in terms of wastage of radio resources. Hence, in a true sense BCFP clearly outperforms RR in every aspect.

6.2. Recommendations for Future research works

- It would be interesting to evaluate the performance of LBTB in a Software Defined Radio (SDR) test bed. SDRs are the reconfigurable radios in which the physical as well MAC layer can be re-tailored using reconfigurable hardware. Evaluating the performance in such an environment gives more rigorous treatment of the evaluation, and therefore more reliable results, and actual performance of scheduling discipline.
- The development of LBTB+BCFP can be further extended to the scenario of Cognitive Radios. A scenario of a cognitive radio relay can be considered, which is providing coverage to different types of devices, working in different frequencies with a completely different set of physical layer parameters: digital modulation, multiple access scheme, and transmitted power. The challenge in extending the development of LBTB+BCFP is in designing the architecture of such algorithm, physical layer parameters to be considered.
- Since the scheduling algorithm running in the base station has 5ms to make a decision about scheduling bandwidth to different users within cell coverage, the complexity of scheduling algorithm is a big issue. Therefore, it would be interesting to extend this work, for reducing the complexity of LBTB+BCFP.
- The work on LBTB can be further extended to the scenario of using cognitive radio in cellular world. One of the areas, where cognitive radio finds application in cellular world, is spectrum trading. The idea of spectrum trading revolves around spectrum pooling. In spectrum pooling, a closed group of cellular operators deposit their unused spectrum in a pool, over a short- term duration. From the spectrum collected in a pool, it is re-distributed using some mechanism.

References

[1] P. Ramanathan and P. Agrawal, "Adapting packet fair queuing algorithms to wireless networks," in *Proceedings of the fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking,* Dallas, Texas, USA, 1998, pp. 1-9.

[2] You-Chiun Wang; Yu-Chee Tseng; Wen-Tsuen Chen; Kun-Cheng Tsai, "MR-FQ: a fair scheduling algorithm for wireless networks with variable transmission rates," in *Information Technology: Research and Education, 2005. ITRE 2005*, pp.250-254, 27-30 June 2005.

[3] Chakchai So-In; Jain, R.; Al Tamimi, Abdel-Karim, "eOCSA: An algorithm for burst mapping with strict QoS requirements in IEEE 802.16e Mobile WiMAX networks," in *Wireless Days (WD), 2009 2nd IFIP* , Paris, France, pp.1-5, 15-17 Dec. 2009.

[4] Lin, P.; Benssou, B.; Ding, Q.L.; Chua, K.C., "CS-WFQ: a wireless fair scheduling algorithm for error-prone wireless channels," in *Proceedings of the Ninth International Conference on Computer Communications and Networks*, Las Vegas, Nevada, USA, 2000., pp.276-281.

[5] Chakchai So-In; Jain, R.; Al Tamimi, A.-K., "OCSA: An algorithm for burst mapping in IEEE 802.16e mobile WiMAX networks," in *Proceedings of the Fifteenth Asia-Pacific Conference on Communications*, Shanghai, China, 2009., pp.52-58, 8-10 Oct. 2009.

[6] Hosein, P., "Cross-Layer Design for Data Burst Construction in the Downlink of IEEE 802.16 Systems," in *Proceedings of the IEEE Global Telecommunications Conference*, New Orleans, Louisiana,USA, pp.1-5, Nov. 30 2008-Dec. 4 2008

[7] Parekh, A.K.; Gallager, R.G., "A generalized processor sharing approach to flow control in integrated services networks-the multiple node case," in *Proceedings of the Twelfth Annual Joint Conference of the IEEE Computer and Communications Societies*, San Francisco, California, USA, pp.521-530 vol.2, 1993.

[8] A. Demers, S. Keshav and S. Shenker, "Analysis and simulation of a fair queueing algorithm," in *Symposium Proceedings on Communications Architectures \& Protocols,* Austin, Texas, USA, 1989, pp. 1-12.

[9] Bennett, J.C.R.; Hui Zhang, "WF^2Q: worst-case fair weighted fair queueing," in *Proceedings of the Fifteenth Annual Joint Conference of the IEEE Computer Societies*, San Francisco, California, USA, vol.1, pp.120-128, 24-28 Mar 1996.

[10] Golestani, S.J., "A self-clocked fair queueing scheme for broadband applications," in *Proceedings of the Thirteenth Conference on Networking for Global Communications*, Toronto, Ontario, Canada, pp.636-646 vol.2, 12-16 Jun 1994.

[11] Yunkai Zhou; Harish Sethu, "On the relationship between absolute and relative fairness bounds," in *IEEE Communications Letters*, vol.6, no.1, pp.37-39, Jan. 2002.

[12] Bhagwat, P.; Bhattacharya, P.; Krishna, A.; Tripathi, S.K., "Enhancing throughput over wireless LANs using channel state dependent packet scheduling," in *Proceedings of the Fifteenth Annual Joint Conference of the IEEE Computer Societies*, San Francisco, California, USA, vol.3, pp.1133-1140, 24-28 Mar 1996.

[13] Fragouli, C.; Sivaraman, V.; Srivastava, M.B., "Controlled multimedia wireless link sharing via enhanced class-based queuing with channel-state-dependent packet scheduling," in *Proceedings of the Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies*, San Francisco, California, USA, vol.2, pp.572-580, 29 Mar-2 Apr 1998.

[14] Songwu Lu; Bharghavan, V.; Srikant, R., "Fair scheduling in wireless packet networks," in *IEEE/ACM Transactions on Networking*, vol.7, no.4, pp.473-489, Aug 1999.

[15] Ng, T.S.E.; Stoica, I.; Hui Zhang, "Packet fair queueing algorithms for wireless networks with location-dependent errors," in *Proceedings of the Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies*, San Francisco, California, USA, vol.3, pp.1103-1111, 29 Mar-2 Apr 1998.

[16] Wong, W.K.; Leung, V. C M, "Scheduling for integrated services in next generation packet broadcast networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference*, New Orleans, Louisiana, USA, 1999., vol.3, pp.1278-1282.

[17] Sheng-Lin Wu; Chen, W.-S.E., "The token-bank leaky bucket mechanism for group connections in ATM networks," in *Proceedings of the International Conference on Network Protocols*, Columbus, Ohio, USA, 1996., pp.226-233, 29 Oct-1 Nov 1996.

[18] Wong, W.K.; Haiying Zhu; Leung, V. C M, "Soft QoS provisioning using the token bank fair queuing scheduling algorithm," in *IEEE Wireless Communications*, vol.10, no.3, pp.8-16, Jun 2003.

[19] Wong, W.K.; Tang, H.; Shanzeng Guo; Leung, V. C M, "Scheduling algorithm in a point-to-multipoint broadband wireless access network," in *Proceedings of the 58th IEEE Vehicular Technology Conference*, vol.3, pp.1593-1597, 6-9 Oct. 2003.

[20] Ohseki, T.; Morita, M.; Inoue, T., "Burst Construction and Packet Mapping Scheme for OFDMA Downlinks in IEEE 802.16 Systems," in *Proceedings of the IEEE Global Telecommunications Conference*, Washington, DC, USA, pp.4307-4311, 26-30 Nov. 2007.

[21] Xin Jin; Jihua Zhou; Jinlong Hu; Jinglin Shi; Yi Sun; Dutkiewicz, E., "An Efficient Downlink Data Mapping Algorithm for IEEE802.16e OFDMA Systems," in *Proceedings of the IEEE Global Telecommunications Conference*, New Orleans, Louisiana, USA, pp.1-5, Nov. 30 2008-Dec. 4 2008.

[22] R. Jain, D. Chiu, and W. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer systems," *DEC Research Report* TR-301, Digital Equipment Corporation, Maynard, MA, USA, Sept 1984.

[23] Bokhari, F.A.; Yanikomeroglu, H.; Wong, W.K.; Rahman, M., "Fairness Assessment of the Adaptive Token Bank Fair Queuing Scheduling Algorithm," in *Proceedings of the 68th IEEE Vehicular Technology Conference*, Calgary, Alberta, Canada, pp.1-5, 21-24 Sept. 2008.

[24] F. A. Bokhari, "Adaptive Token Bank Fair Queuing scheduling in the downlink of 4G wireless networks," MASc Thesis, Carleton University (Canada), Canada, 2008.

[25] J. Korhonen, Introduction to 3G Mobile Communications, Boston, Massachusetts, USA: Artech House, 2001.

[26] M. Ergen, Mobile Broadband - Including WiMAX and LTE, Berkley, California, USA: Springer Science+Business Media, 2009.

[27] R. Prasad, OFDM for Wireless Communications Systems, Boston, Massachusetts, USA: Artech House, 2004.

[28] F. De Rango, A. Malfitano, S. Marano, N. Krichene and N. Boudriga, Cross-Layer End-to-End QoS Architecture: The Milestone of WiMAX; QoS in Mobile WiMAX, Chichester, United Kingdom, John Wiley and Sons, 2010.

[29] L. Nuaymi, WiMAX : Technology for Broadband Wireless Access, Chichester, United Kingdom, John Wiley and Sons, 2007.

[30] M. Steer, Microwave and RF Design: A Systems Approach : Beta Edition, Raleigh, North Carolina, USA: SciTech Publishing, 2010.

[31] B. Sklar, Digital Communications: Fundamentals and Applications, New Delhi, India: Pearson Education, 2007.

[32] W. Forum, "WiMAX : A way forward in India," WiMAX Forum, 2010.

[33] "IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Broadband Wireless Access Systems," *IEEE Std 802. 16-2009 (Revision of IEEE Std 802. 16-2004),* pp. 1-2080, 2009.

[34] Chakchai So-In; Jain, R.; Tamimi, A.-K., "Scheduling in IEEE 802.16e mobile WiMAX networks: key issues and a survey," in *IEEE Journal on Selected Areas in Communications*, vol.27, no.2, pp.156-171, February 2009.

[35] S. Yang, OFDMA System Analysis and Design, Boston, Massachusetts, USA: Artech House, 2009.

[36] Desset, C.; de Lima Filho, E.B.; Lenoir, G., "WiMAX Downlink OFDMA Burst Placement for Optimized Receiver Duty-Cycling," in *Proceedings of the IEEE International Conference on Communications*, Glasgow, United Kingdom, pp.5149-5154, 24-28 June 2007.

[37] W. K. Wong, "Packet scheduling in wireless systems by token bank fair queuing algorithm", PhD Thesis, The University of British Columbia (Canada), Canada, 2005.

[38] Kulkarni, S.S.; Rosenberg, C., "Opportunistic scheduling policies for wireless systems with short term fairness constraints," in *Proceedings of the IEEE Global Telecommunications Conference*, vol.1, pp.533-537, 1-5 Dec. 2003.

[39] A. Leon-Garcia and I. Widjaja, Communication Networks: Fundamental Concepts and Key Architectures, New York, USA: McGraw-Hill Higher Education, 2004.

[40] Bae, J.J.; Suda, T., "Survey of traffic control schemes and protocols in ATM networks," in IEEE, vol.79, no.2, pp.170-189, Feb 1991.

[41] Bokhari, F.A.; Wong, W.K.; Yanikomeroglu, H., "Adaptive Token Bank Fair Queuing Scheduling in the Downlink of 4G Wireless Multicarrier Networks," in *Proceedings of the IEEE Vehicular Technology Conference*, Singapore, pp.1995-2000, 11-14 May 2008.

[42] Ahmadzadeh, A.M and Sanchez-Garcia, J.E and Saavedra-Moreno, B and Portilla-Figueras and Salcedo-Sanz, S., "Capacity estimation algorithm for simultaneous support of multi-class traffic services in Mobile WiMAX," in *Computer Communications*, vol 35, no. 1, pp. 109-119, January 2012.

[43] Hong-Shen Wang; Moayeri, N., "Finite-state Markov channel-a useful model for radio communication channels," in *IEEE Transactions on Vehicular Technology*, vol.44, no.2, pp.163-171, Feb 1995.

[44] Naser, H.; Mouftah, H.T., "A joint-ONU interval-based dynamic scheduling algorithm for Ethernet passive optical networks," in *IEEE/ACM Transactions on Networking*, vol.14, no.4, pp.889-899, Aug. 2006.

[45] M. S. Taqqu, W. Willinger and R. Sherman, "Proof of a fundamental result in self-similar traffic modeling," in *Computer Communication Review.,* vol. 27, pp. 5-23, Apr, 1997.