A Connection Admission Control Framework for UMTS based Satellite Systems

An Adaptive Admission Control algorithm with pre-emption control mechanism for unicast and multicast communications in satellite UMTS

ANJU PILLAI

BSc (Hons), MSc

Submitted for the Degree of
Doctor of Philosophy

School of Engineering, Design and Technology

University of Bradford

2011

ABSTRACT

Name: Anju Pillai

Thesis Title: A Connection Admission Control Framework for UMTS based Satellite Systems: An Adaptive Admission Control algorithm with pre-emption control mechanism for unicast and multicast communications in satellite UMTS

Keywords: Satellite Networks, UMTS, Admission Control, Pre-emption Control, Unicast, Multicast, Multimedia.

In recent years, there has been an exponential growth in the use of multimedia applications. A satellite system offers great potential for multimedia applications with its ability to broadcast and multicast a large amount of data over a very large area as compared to a terrestrial system. However, the limited transmission capacity along with the dynamically varying channel conditions impedes the delivery of good quality multimedia service in a satellite system which has resulted in research efforts for deriving efficient radio resource management techniques. This issue is addressed in this thesis, where the main emphasis is to design a CAC framework which maximizes the utilization of the scarce radio resources available in the satellite and at the same time increases the performance of the system for a UMTS based satellite system supporting unicast and multicast traffic.

The design of the system architecture for a UMTS based satellite system is presented. Based on this architecture, a CAC framework is designed consisting of three different functionalities: the admission control procedure, the retune procedure and the pre-emption procedure. The joint use of these functionalities is proposed to allow the performance of the system to be maintained under congestion. Different algorithms are proposed for different functionalities; an adaptive admission control algorithm, a greedy retune algorithm and three pre-emption algorithms (Greedy, SubSetSum, and Fuzzy).

A MATLAB simulation model is developed to study the performance of the proposed CAC framework. A GUI is created to provide the user with the flexibility to configure the system settings before starting a simulation. The configuration settings allow the system to be analysed under different conditions.

The performance of the system is measured under different simulation settings such as enabling and disabling of the two functionalities of the CAC framework; retune procedure and the pre-emption procedure. The simulation results indicate the CAC framework as a whole with all the functionalities performs better than the other simulation settings.

ACKNOWLEDGEMENTS

The writing of this thesis has been a very long journey. It would not have been possible without the support of the people around me. I wish to express my sincere appreciation to those who have supported me in one way or the other during this amazing journey.

First of all, I would like to extend my sincere gratitude to my PhD supervisors, Prof. Yim Fun Hu, and Dr Rosemary Halliwell. I am very grateful to Prof. Hu for giving me the opportunity to pursue this PhD programme under her guidance. Her invaluable suggestions and discussions on my research work have enabled me to finish this work. I remain indebted for her understanding and support during my pregnancy days. She has always inspired me to work hard. I am also grateful to Dr Rosemary for lending her suggestions and insightful feedback and comments on my work. Her ever smiling face and calm nature has helped to ease the anxiety in the last phase of writing up of thesis.

I also wish to express my gratitude to the Inmarsat BGAN project for providing the funding. A special thanks to Paul Febvre from Inmarsat and Juan Rivera from ESA for their suggestions and help.

I would like to thank all my colleagues in the research group. It was great working with all of them. I would also like to thank my friends outside of the work space; Nanu, Karen, Era, and Preeti, who have always supported and wished the best for me.

Most of all, I would like to thanks my parents, my brother, and my sister for their constant love and support. A special thanks to my dad who has worked extremely hard all his life to give us the best. His strict discipline and strong focus on the studies has always inspired me. I would also like to thank my in-laws who have been very kind, supportive, and encouraging during this long journey.

Finally, I would like to acknowledge the two most important people in my life; my loving husband, Dr Prashant Pillai, and my lovely daughter, Navya Pillai (3 years old). I would like to thank my husband for his personal support and great patience at all times.

To my parents,

Mrs. Rajesh Singh and Mr. Dhan Singh

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

Acronym	Meaning
1G	1st Generation
2G	2nd Generation
3G	3rd Generation
3GPP	3rd Generation Partnership Project
ABR	Available Bit Rate
ACM	Adaptive Coding and Modulation
AL	Adaptation Layer
ALC	Admission Limit Curve
AM	Acknowledged Mode
AMPS	Advanced Mobile Phone System
AMSS	Aeronautical Mobile Satellite Service
ARQ	Automatic Repeat Request
AS	Access Stratum
ATM	Asynchronous Transfer Mode
BA	Bandwidth Adaptation
BCn	Bearer Connection
BCt	Bearer Control
BEF	Bandwidth Expansion Factor
BER	Bit Error Rate
BGAN	Broadband Global Area Network
B-ISDN	Broadband Integrated Service Digital Network
BoD	Bandwidth on Demand
BMSN	BGAN Multicast Service Node
BMSC	BGAN Multicast Service Centre
BS	Base Station
BSS	Broadcast Satellite Service
CAC	Call/Connection Admission Control
CBR	Constant Bit Rate
CBR	Call Blocking Rate
CC	Congestion Control
CCM	Constant Coding and Modulation
CDF	Cumulative Distribution Function
CDMA	Code Division Multiple Access
CDR	Call Dropping Rate
CEB	Classic Effective Bandwidth
CIM	Connection Inactive Mechanism

CN	Core Network
C/No	Carrier to Noise Ratio
CP	Complete Partitioning
CRP	Common Resource Pool
CS	Complete Sharing
CS	Circuit Switched
DBS	Direct Broadcasting Satellites
DMBS	Double Movable Boundary Strategy
DTH	Direct To Home
DVB-RCS	Digital Video Broadcasting – Return Channel via Satellite
DVB-S	Digital Video Broadcasting over Satellite
EBB	Extended Effective Bandwidth
ETSI	European Telecommunication Standards Institute
FCC	Fuzzy Congestion Controller
FDMA	Frequency Division Multiple Access
FEC	Forward Error Correction
FES	Fixed Earth Station
FGC	Fractional Guard Channel
FIFO	First In First Out
FIS	Fuzzy Inference System
FL	Fuzzy Logic
FLC	Fuzzy Logic Controller
FMT	Fade Mitigation Techniques
FQC	Fuzzy QoS Controller
FSS	Fixed Satellite Service
FTP	File Transfer Protocol
GA	Genetic Algorithm
GC	Guard Channel
GEO	Geostationary Earth Orbit
GGSN	Gateway GPRS Support Node
GMSC	Gateway MSC
GMM	GPRS Mobility Management
GPS	Global Positioning System
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communication
GTP	GPRS Tunnelling Protocol
GUI	Graphical User Interface
HLR	Home Location Register
HSPA	High Speed Packet Access

I-4	INMARSAT-4
IAI	Inmarsat Air Interface
ICLR	Individual Cell Loss Ratio
IMT-2000	International Mobile Telecommunications 2000
ITU	International Telecommunications Union
IP	Internet Protocol
ISDN	Integrated Services Digital Network
L1	Physical Layer
LCFD	Last Come First Dropped
LCFP	Last Come First Pre-empted
LEO	Low Earth Orbit
LOS	Line of Sight
LMSS	Land Mobile Satellite Service
MAC	Media Access Control
MBAC	Measurement Based Admission Control
MBMS	Multimedia Broadcast Multicast Services
MD-CAC	Markov Decision-based Call Admission Control
MDP	Markov Decision Process
MEO	Medium Earth Orbit
MFAC	Multimedia Fuzzy Admission Control
ML	Maximum Likelihood
MM	Mobility Management
MMSS	Maritime Mobile Satellite Service
MMPP	Markov Modulated Poisson Process
MPEG-4	Motions Pictures Expert Group – 4
MSC	Mobile Switching Centre
MSS	Mobile Satellite Service
MT	Mobile Terminal
NAS	Non-Access Stratum
NCC	Network Control Centre
NGWS	Next Generation Wireless System
NMT	Nordic Mobile Telephone System
NN	Neural Networks
NRT	Non Real Time
OBP	On Board Processing
OC	Optimized Centralized
OP	Optimized Proportional
PDCP	Packet Data Convergence Protocol
PDP	Packet Data Protocol

PID	Proportional Integrative Directive
PLP	Packet Loss Probability
PLMN	Public Land Mobile Network
PPBP	Path Prediction Based Preemption
PR	Peak Rate
PS	Packet Switched
PSAB	Primary Shared Access Bearer
QoS	Quality of Service
RAB	Radio Access Bearer
RAN	Radio Access Network
RANAP	Radio Access Network Application Protocol
RCBC	Reference Chaser Bandwidth Controller
RCCT	Random Capacity Change Time
RCST	Return Channel Satellite Terminal
RF	Radio Frequencies
RLC	Radio Link Control
RNC	Radio Network Controller
RNS	Radio Network Subsystem
RRC	Radio Resource Control
RRM	Radio Resource Management
RT	Real Time
RTD	Round Trip Propagation Delay
RT-VBR	Real Time – Variable Bit Rate
SAP	Service Access Point
SCCP	Signalling Connection Control Point
SGSN	Serving GPRS Support Node
SLA	Service Level Agreement
SM	Session Management
SMDP	Semi Markov Decision Process
SMS	Short Message Service
SS	Supplementary Service
ST	Satellite Terminal
S-UMTS	Satellite UMTS
TACS	Total Access Communications System
TDM	Time Division Multiplexing
TDMA	Time Division Multiple Access
TE	Terminal Equipment
THP	Traffic Handling Priority
TM	Transparent Mode

T-UMTS	Terrestrial UMTS
UE	User Equipment
UESS	UE Specific Signalling
UM	Unacknowledged Mode
UMTS	Universal Mobile Telecommunication System
USB	Universal Serial Bus
USIM	Universal Subscriber Identity Module
USRAN	UMTS Satellite Radio Access Network
UTRAN	UMTS Terrestrial Radio Access Network
VBR	Variable Bit Rate
VP	Virtual Partitioning
WDM	Wavelength Division Multiplex
WCDMA	Wideband Code Division Multiple Access

Chapter 1 : INTRODUCTION

1.1 Background

Today's world has shrunk in terms of communication. The ever advancing mobile communication system provides ubiquitous communication anywhere and anytime. Mobile communication systems are broadly categorized as terrestrial and satellite. These two systems have now been fully integrated to reap the benefits of both the technologies.

The history of terrestrial mobile telephone systems goes way back in early 1980s when the first generation (1G) mobile communication systems were launched based on analogue transmission. These systems used multiple cell sites and provided the ability to transfer calls from one site to another as the user travelled between cells during a conversation. Two major 1G systems were the Advanced Mobile Phone System (AMPS) which was used in the USA and many other countries, and Total Access Communications System (TACS) which was used in the UK as well as many other countries around the world. Another system that was commercially deployed was Nordic Mobile Telephone System (NMT), particularly in Scandinavia, Eastern Europe and Asia. However, the use of all these systems installed around the globe increased dramatically and the effects of the limited frequency allocations were soon noticed [1].

To overcome these problems, the development of the second generation (2G) systems was started addressing the need for more spectrally efficient cellular network. These 2G systems were based on digital technology as opposed to the analogue technology of 1G system. In 1991, the first pan-European 2G system, Global System for Mobile Communication (GSM) was launched which was standardized by European Telecommunications Standard Institute (ETSI). In addition to providing circuit-switched voice communication services, the GSM standard also supports international roaming and Short Message Service (SMS). Whilst GSM is excellent for voice calls, it is limited when it comes to data transfer and with the ever growing use of internet; GSM technology does not completely fulfil the needs of the users around the world [2]. Hence, the General Packet Radio Service (GPRS) was developed. GPRS is a packet oriented mobile data service aiming to enhance the GSM systems to enable them to transfer data at faster rate typically 32-48 kbps. GSM combined with GPRS is often defined as 2.5G, a technology between 2G and 3G.

In the last decade, there has been tremendous increase in the usage of the internet and the multimedia services. In order to cope with the demand of such services, International Telecommunication Union (ITU) started the process of defining the standard for third generation (3G) systems, referred to as International Mobile Telecommunications 2000 (IMT-2000). In Europe, ETSI was responsible for Universal Mobile Telecommunication Systems (UMTS) standardisation process. In 1998, the Third Generation Partnership Project (3GPP) was formed to continue the technical specification work. It uses Wideband Code Division Multiple Access (W-CDMA) as the underlying air interface. UMTS integrates packet and circuit data transmission and provides a data rate up to 2 Mbps, which has been increased by further 3GPP standards for High Speed Packet Access (HSPA) services. This capability together with the inherent IP support allows delivery of the interactive multimedia services and wideband applications such as video telephony and video conferencing [3] .

In parallel with the development of the terrestrial mobile communication systems, the growth of the satellite mobile communication systems has also been in constant progression. The International Telecommunications Union (ITU) has categorized radio services according to their broad applications: Fixed Satellite Service (FSS), Broadcast Satellite Service (BSS) or Mobile Satellite Service (MSS). FSS is the oldest and most used of all the satellite services. It is intended for communication through a satellite between earth stations that are fixed. INTELSAT is one of the major operators of FSS which includes services such as satellite newsgathering and as part of the internet backbone. BSS is responsible for broadcasting services such as broadcast of satellite television. It covers the area of Direct Broadcasting Satellites (DBS). MSS consists of earth stations which may be in motion and depending on the position of the earth station, the services can be divided into three major categories: Maritime Mobile Satellite Service (MMSS), Aeronautical Mobile Satellite Service (AMSS) and Land Mobile Satellite Service (LMSS) [4]. The major supplier of MSS services is INMARSAT. On 11th March, 2005, INMARSAT successfully launched the first of three INMARSAT-4 (I-4) satellites which delivers a 3-G compatible broadband data service to mobile users. With the successful launch of the first satellite, INMARSAT launched its new Broadband Global Area Network (BGAN) service, which became a global service in 2009 following the successful launch of the third I-4 satellite. BGAN is an IP and circuit-switched service which offers voice telephony and a sophisticated range of high-bandwidth services, including internet access, videoconferencing, LAN and other services, at speeds of up to 432 kb/s. This allows users to set up a broadband mobile office anywhere on the planet using a BGAN terminal. The provision of the multicast services such as video streaming is one of the key features to be supported by the BGAN system [5].

1.2 Motivations and Objectives

In recent years, there has been an exponential growth in the use of multimedia applications and an efficient multimedia delivery system requires high speed broadband networks. A UMTS enabled satellite system offers great potential for multimedia applications with its ability to broadcast and multicast a large amount of data over a very large area. However, there are many challenges in designing a satellite communication system such as Round Trip propagation Delay (RTD), atmospheric effects, channel losses and satellite life time. The limited transmission capacity along with dynamically varying channel conditions impedes the delivery of good quality multimedia service in S-UMTS systems. Hence, an efficient Radio Resource Management (RRM) technique becomes the key to deliver acceptable Quality of Service (QoS) to services while providing adequate resource utilization efficiency [6].

The main objective of this PhD project is to design a CAC framework for a UMTS based satellite system which aims at maximizing the utilization of the scarce radio resources available in the satellite and at the same time increases the performance of the system. The Call/Connection[1] Admission Control (CAC) technique grants/denies the access of the arrival calls/connection to the network, based on the pre-defined criteria. The following objectives have been set for this PhD work:

- Identification of the modification in the protocol stack of the terrestrial UMTS in order to support the satellite space segment for the S-UMTS network.
- Designing a unique CAC framework applicable to the S-UMTS networks consisting of the admission control procedure, retune procedure and the pre-emptive control mechanism supporting unicast and multicast traffic.
- Modelling of the CAC framework in a performance evaluation tool such as MATLAB.
- Performance analysis of the proposed algorithms in the CAC framework.

1.3 Contributions and Achievements

The following is a list of research contributions addressed in this thesis:

➢ **CAC Framework:** A CAC framework has been developed for the S-UMTS system consisting of three main functionalities; the admission control, the retune procedure and the pre-emption mechanism. The joint use of these functionalities have been proposed to allow the performance of the system to be maintained under different congestion states such as:

[1] The terms ‘calls’ and ‘connections’ are used interchangeably throughout the thesis.

- Requested link is not congested – Under such condition, only admission control procedure is invoked.
- Requested link is congested – Under such condition, initially the admission control procedure is invoked. However, due to congestion on the requested link, the connection will get blocked. In such a case, the retune procedure is triggered which gives the flexibility to the system to admit the rejected connection by retuning to another available link.
- All links are congested – Under this condition, the pre-emption procedure is invoked which allows the system to selectively remove the on-going connections with the requesting connection of a higher precedence.

- **Admission Decision Processes for Unicast and Multicast Traffic:** The separate admission decision processes have been defined for the unicast and the multicast traffic. For the unicast traffic, the different components of the CAC framework perform collectively to admit the requested connection in the system by initially checking the status of the requested link and if required for all other links. For the multicast traffic, the possibility of joining the requested multicast session on the same or different link is given priority over the creation of a new multicast session. This helps to increase the channel utilization in the multicast mode of transmission.
- **Adaptive CAC Algorithm for Unicast traffic:** The CAC algorithm designed for this thesis work is based on the equivalent capacity approach presented in the literature [7]. However, in order to make the

approach suitable to the S-UMTS system with varying link capacity and supporting different types of satellite terminals, an adaptive admission control algorithm is proposed. The algorithm not only takes into account the traffic characteristics of the requesting connection and the resources available in the system whilst making the admission decision, it also adapts to changes in the link condition experienced by each user due to the weather conditions, mobility and other factors.

- **Adaptive CAC Algorithm for Multicast traffic:** For multicast traffic, the adaptive admission control algorithm works in similar manner to that for the unicast traffic except that the data rate supported by a multicast group is re-evaluated every time a new multicast connection joins an existing multicast group. The data rate at which the multicast content is sent in a particular multicast group depends on the type of the satellite terminal. The data rate for the multicast group is selected such that the terminal of the lowest capability in terms of the supported data rate can also receive the multicast data. This allows the multicast data to be received by all the users belonging to a particular multicast group.
- **Retune mechanism:** The retune process is triggered when the requesting connection cannot be admitted on the given link. A greedy retune procedure is proposed which aims at reducing the number of connections retuned from one link to another by selecting the connections with the highest resource usage to retune.
- **Pre-emption procedure:** The pre-emption procedure is activated when the system is congested. The main objective of this procedure is to admit a higher priority connection by pre-empting or dropping existing lower

priority connections. Three pre-emption algorithms are proposed; Greedy, SubSetSum and Fuzzy. The Greedy pre-emption algorithm performs by pre-empting the connections with the lowest resource usage. The purpose of this algorithm is to reduce the bandwidth which is released by pre-empting the connections. However, in doing so, it 'greedily' pre-empts more connections. The SubSetSum pre-emption algorithm provides an improvement over the Greedy pre-emption algorithm. The algorithm drops one or more, lower priority connections such that the total resource consumption of these connections is just enough to accommodate the new connection request. This enables high priority connection requests to be admitted without dropping more than the necessary existing lower priority connections, thus minimizing the bandwidth released by the pre-emptable connections, and also reducing the number of connections pre-empted. Although, the SubSetSum algorithm provides an optimum solution, it is mathematically complex and requires more computation time. A further improvement over the SubSetSum pre-emption algorithm is provided by an intelligent Fuzzy pre-emption algorithm. It makes use of the expert system knowledge to provide a better system performance as compared to the Greedy and the SubSetSum pre-emption algorithms.

- **Pre-emption Algorithms with Connection Inactive Mechanism (CIM):** The CIM mechanism with the pre-emption control procedure is proposed to further enhance the performance of the system. In contrast with the simple pre-emption procedure where the selected pre-emptable connections are simply dropped, this mechanism changes the state of the connection to be *Non-Active* for a period of time which varies with the type of the connection. Under the *Non-Active* state, the resources used by the selected connections are released and at the end of the elapsed time, the pre-empted connections again request for the resources. In the case that the resources cannot be granted at the time, the *Non-Active* connection is finally dropped else it is admitted and the state of the connection is converted back to the *Active* state.

The following is a list of publications achieved during the course of this PhD programme:

- A. Pillai, Y. F. Hu, P. M. L. Chan, R. W. Millington, P. Febvre and J. R. Castro, "Adaptive Admission Control for MBMS Satellite Systems", Satellite and Space Communications, IWSSC, Tuscany, 2009, pp.326-330.
- A. Pillai, P. M. L. Chan and F. Hu, "Adaptive Connection Admission Control for MBMS Compatible Satellite Systems", 14th Ka and Broadband Communications Conference, Matera, Italy, 2008.
- A. Pillai, K. Xu, R. Lau, P. Pillai and R. W. Millington, “An SDL framework for protocol validation and performance evaluation of the Inmarsat BGAN protocol extension for multicast services”, 14th Ka and Broadband Communications Conference, Matera, Italy, 2008.
- P. Pillai, A. Pillai, K. Xu, V. Chook, P. M. L. Chan, R. W. Millington and P. Febvre, ”Design of a System SDL Framework for the Verification and Validation of the Inmarsat BGAN Protocol Stack”, 12th Ka and Broadband Communications Conference, 27-29 September, 2006, Italy

1.4 Organisation of Thesis

Following this introductory chapter which highlights the purpose and the scope of the thesis, the remainder of the thesis is organised as follows:

- **Chapter 2: Literature Survey of Admission Control and Pre-emption Control Mechanism** – This chapter provides a detailed review of the literature available on the admission control algorithms for various network architectures which are broadly classified as terrestrial and satellite networks. A review on the pre-emption control algorithms is also presented.
- **Chapter 3: S-UMTS System Architecture** – This chapter provides an overview of the S-UMTS system consisting of the user segment, the space segment and the ground segment. The modification of the terrestrial UMTS protocol stack in order to support the UMTS services via satellite is also described.
- **Chapter 4: CAC Framework Model** – This chapter focuses on the design of the CAC framework model for the UMTS based satellite system. The three main functionalities of the CAC framework: CAC Processor, the Retune Controller and the Pre-emption Controller are defined in detail in separate subsections. The CAC Processor contains the adaptive and the non-adaptive admission control algorithms. The admission decision processes for unicast and multicast traffic are also presented. In the Retune Controller subsection, the algorithm for the retuning is described. In the Pre-emption Controller subsection, the three proposed pre-emption algorithms; Greedy, SubsetSum and Fuzzy pre-emption algorithms, along with CIM mechanism are explained in detail.

- **Chapter 5: Simulation Framework Model** – This chapter describes the MATLAB Simulink model of the proposed CAC framework for the S-UMTS network architecture developed for analysing the different proposed elements in the CAC framework. A Graphical User Interface (GUI) has also been developed which provides the user with the flexibility to configure the system settings before starting a simulation.
- **Chapter 6: Simulation Scenarios and Results** – This chapter describes the different simulation scenarios to verify the correct working of the CAC framework which consists of three different functionalities; the CAC Processor, the Retune Controller and the Pre-emption Controller. Based on the type of functionalities supported by the system, four simulation sets are defined; *CAC Processor without retune and pre-emption capability*, *CAC Processor with only retune capability*, *CAC Processor with only pre-emption capability*, and *CAC Processor with both retune and pre-emption capability*. Each of these simulation sets are tested by a set of simulation scenarios which are further divided into various test cases. The simulation parameters defined to analyse the system performances under different scenarios are also presented. The simulation results are obtained and analysed.
- **Chapter 7: Conclusion and Further work** – This chapter provides the conclusion and summarizes the contributions made by the research. It also provides any additional research activities for future development.

Chapter 2 : LITERATURE SURVEY OF ADMISSION CONTROL AND PRE-EMPTION CONTROL MECHANISM

2.1 Introduction

The Connection Admission Control (CAC) is an integral part of resource management to ensure Quality of Service (QoS) can be provided in a network. It is the process which takes a new connection request and decides whether there are sufficient available resources on the requested link to admit or reject the new connection request. A connection can only be accepted if sufficient resources are available to establish the connection end-to-end with the requested QoS. However, in doing so, the agreed QoS of existing connections in the network must not be affected by the new connection. This thesis also addresses the congestion control problem as a direct consequence of an emergency situation by adopting the policy whereby a low priority active connection is pre-empted to allow the admission of a higher priority connection. This chapter describes in detail the existing work on the CAC and pre-emption schemes and their suitability to the work in this thesis.

2.2 CAC Algorithms

The CAC is a key element in providing the guaranteed QoS in any network. A plethora of CAC algorithms have been proposed in the literature over a number of years. This section provides a comprehensive study on some of the existing CAC algorithms for different networks. The suitability of an admission control scheme varies from one type of network to another. The networks can be broadly classified into the terrestrial and the satellite networks. The various CAC schemes for these networks are reviewed in the following subsections.

2.2.1 Terrestrial Networks

All non-satellite networks are characterized as terrestrial networks. These networks can further be classified into wired and wireless networks.

2.2.1.1 Wired Networks

Many CAC algorithms have been proposed for Ethernet and other wired networks particularly in the context of Broadband Integrated Service Digital Network (B-ISDN) based on the Asynchronous Transfer Mode (ATM) technology [8]. A selection of many admission control schemes for wired networks is briefly described in the following subsections.

2.2.1.1.1 Parameter based CAC

In this type of scheme, the admission decision is based on the characteristics of the user traffic. Two types of parameter based admission control algorithms exist; Non-Statistical allocation also known as Peak Bandwidth allocation, and Statistical allocation. In the former case, the connection is admitted if the sum of the peak rates of existing connections and the requested connection is less than the total link capacity. The bandwidth assigned to the new request will be equal to its requested peak rate. This method is suitable for Constant Bit Rate (CBR) traffic which uses its peak rate at any given point of time. However, the network utilization efficiency for this type of allocation can be considerably lower for bursty or Variable Bit Rate (VBR) traffic. In Statistical allocation, the bandwidth allocated to the new connection is between the peak rate and the sustained rate. As a result, the sum of the peak rates of the existing connections may be greater than the total link capacity. While this method may be more economical when dealing with VBR traffic, it is more difficult to carry out effectively due to the potential of network congestion [9] [10].

2.2.1.1.2 Effective Bandwidth CAC

Using the *Effective Bandwidth* or *Equivalent Capacity* scheme, an appropriate effective bandwidth is assigned to each connection which is midway between the peak rate and the mean rate of the connection [8]. The feasibility of admitting a connection is determined by ensuring that the sum of the effective bandwidths of the existing connections and the requested connection is less than the total link capacity. The authors of [7] consider the bandwidth allocation issue in the high speed fast packet-switched network supporting connections with different bandwidth requirements using equivalent capacity approach. The bandwidth requirement or the equivalent capacity of both the individual and the multiplexed connections is computed which is then used for efficient bandwidth allocation problem. An effective bandwidth based admission control for a network supporting connections with different priority classes is presented in [11]. With the introduction of the priority classes, there is a need to modify the connection admission control policy. Hence, it is proposed that each connection is associated with more than one effective bandwidth, one for the corresponding priority level of the connection and one for each lower priority level. This is shown to perform better than the effective bandwidth with no consideration to the priorities. The authors of [12] perform the admission control by modelling the arrival traffic as a two-state Markov Modulated Poisson Process (MMPP) and calculating the effective bandwidth of the two-state MMPP. The results have shown high utilization compared with the conventional methods. An Extended Effective Bandwidth (EBB) approximation is proposed in [13] for an ATM network supporting both Markovian traffic as well as self-similar traffic. It compares the EEB approximation with the Classic Effective Bandwidth (CEB) approximation. The results indicate easy implementation of EEB approximation over CEB approximation in ATM systems.

2.2.1.1.3 Measurement based CAC

This type of scheme uses network measurements to calculate the load of the existing traffic unlike the parameter based CAC which uses the traffic characteristics. The admission decision is based solely on the current network state and it has no prior knowledge of the traffic statistics. A measurement-based approach lightens the burden on the users to accurately specify the parameters for the traffic flow, and therefore it is a more practical approach for achieving statistical multiplexing gain with VBR traffic [10]. A measurement based CAC for integrated services packet networks supporting real time applications is presented in [14]. The proposed algorithm focuses on the real time predictive services which provide fairly reliable delay bound as compared to guaranteed services. The results also indicate higher network utilization with predictive services. The authors of [15] presents an easy to implement measurement based admission control for real-time traffic in IP differentiated services networks that uses only measurements of the aggregate bandwidth without considering any per-flow information.

2.2.1.2 Wireless Networks

The admission control issue is more complicated in wireless networks than in wired networks due to the unique features of wireless networks such as channel multiple access interference, channel impairments, handoff requirements and limited bandwidth. In First Generation (1G) and Second Generation (2G) wireless systems, CAC was developed for a single service environment (voice) whereas in Third Generation (3G) systems, multimedia services such as voice, video, data and audio are offered with QoS profiles. Hence, more sophisticated CAC schemes have been developed for 3G networks [16]. The CAC schemes for wireless networks can be broadly classified based on the following design choice:

1) Type of traffic
2) Information scale
3) Optimization
4) Adaptive multimedia

Each of these design types are explained in detail in the following subsections.

2.2.1.2.1 Type of traffic

Based on the type and the number of traffic services supported by the system, the admission control algorithm can be broadly classified into the following:

- Single class CAC algorithm – In such case, only voice traffic is supported by the system. The most commonly used CAC scheme for voice traffic in literature is Guard Channel (GC) scheme. Under this scheme, a higher priority is given to the handoff connections over the new connections by reserving a number of channels known as guard channels for the handoff connections. A GC scheme for mobile radio telephone system is presented in [17]. The authors compare three schemes; one non-prioritized scheme and two prioritized schemes. In the non-prioritized scheme, no guard channels are assigned to the handoff calls. In the first priority scheme, handoff calls are exclusively assigned a fixed number of channels while the remaining channels are used for both new calls and handoff calls. The second priority scheme is similar to the first priority scheme, however the queuing of the handoff calls is allowed. Although the performance of the system is shown to increase with second priority scheme, the infinite queuing of the handoff calls is not applicable practically. Addressing this issue, [18] proposes a GC scheme with finite queuing of the new and handoff calls which is practical and shown to improve the performance of the system in terms of lower blocking probability. The traditional GC schemes considers the new and handoff call arrivals as a Markovian arrival process, [19] proposes the GC scheme using the self-similar traffic streams. Although GC scheme is useful in reducing the blocking probability of the handoff calls, it does so at the cost of the increase in the blocking probability of the new calls. To overcome the shortcoming of the GC, a Fractional Guard Channel (FGC) policy is studied in [20]. According to the FGC scheme, non-integral number of guard channels is assigned to the handoff calls by rejecting the new calls probabilistically depending on the current channel occupancy. This allows the system to reach its maximum capacity while meeting the QoS constraints.

- Multiple class CAC algorithms - The design of the multi-class CAC is more difficult than the single class CAC since issues like class prioritization, fairness and the resource sharing policy are considered [16]. Several policies have been proposed in the literature for the class based admission control.
    1) Complete Sharing (CS) - Under this scheme, the available link capacity is shared amongst the different traffic classes. A call request is admitted if there are enough resources. This policy gives the high channel utilization; however, it does not differentiate between calls of different priorities [21]. A statistical CAC mechanism for IEEE 802.16 network based on CS policy is proposed in [22]. In an IEEE 802.16 networks, the support for the Real-Time Variable Bit Rate service (RT-VBR) causes the bandwidth requirement to vary. The variability of the traffic and the channel state can results in the QoS degradation of the ongoing calls. This algorithm performs by reserving an additional bandwidth for the service whose traffic is variable, preventing the traffic overflow.
    2) Complete Partitioning (CP) - Under this scheme, the available link capacity is divided into as many partitions as the number of traffic classes. A new call request is rejected if there are not

sufficient resources in the partition for the given traffic class [23].

3) Virtual Partitioning (VP) – This scheme combines the advantages of the CS and CP schemes. Each priority traffic class is assigned a capacity according to the offered load. If the traffic class exceeds its nominal capacity, it is declared to be overloaded. While admitting the new connection, the traffic class is given a lower priority. Hence, the priority of the call is not fixed but depends on the state of the system [23-24].

4) Threshold based policies – Under the threshold policy, each class is associated with a threshold parameter. A call of a class is accepted if there are sufficient resources and the number of calls of the given class does not exceed the threshold set for that class [25-27]. A threshold based bandwidth allocation policy for 3G multi-class cellular networks is proposed in[28]. This policy prioritizes connections according to their QoS constraints by assigning a maximum occupancy or a threshold. Handoff calls are also given priority over new calls and hence, this policy provides a lower handoff dropping probability. The authors of [29] combines the threshold based policy with the CS policy and proposes a novel handoff control scheme known as *Dual Trunk Reservation with Queuing* for a voice/data integrated cellular system. It considers two thresholds, one to reserve the channels for the voice handoff and the other for managing the data traffic. A single queue is use to buffer the

data in case the resources are not available. The CS scheme is adopted to maximise the channel utilization. The thresholds are based on the number of channels rather than the number of calls and within a channel, the CS policy is applied.

5) Guaranteed Minimum - Under this scheme, each traffic class is assigned a portion of the total available resource; once used up, classes can then attempt to use the resources from a shared pool [30].

2.2.1.2.2 Information Scale

Depending on the amount of information used by the admission controller, three types of schemes are possible:

- Global - A global CAC algorithm takes the entire network information into consideration in every call admission decision. Although it gives the best possible performance, the global CAC algorithm is very complex to implement [16].
- Local – A local CAC algorithm considers only the information of the cell of the incoming call for making the admission decision. It is very easy to implement. However a local scheme proves to be least efficient. The authors in [31] proposes a local CAC algorithm which uses the global information, however its implementation in each cell uses only local information. Such an optimized local scheme provides almost the same performance as the global CAC in terms of network throughput while it is very easy to implement.
- Semi-Local - A semi-local CAC algorithm lies between the two extremes and considers a cluster of neighbouring cells for the

incoming call to make the decision. While a global scheme is most complex, a local scheme proves to be least efficient, making a semi-local scheme the better choice of the three types. A novel semi-local admission control algorithm is presented in [32] to support multimedia connections with dynamic bandwidth requirements while reducing call dropping probability and maintaining high network resource utilization. It uses the information from not only the originating cell but also from the neighbouring cells while making the admission decision.

2.2.1.2.3 Optimization

The admission control problem can be presented as an optimization problem. Based on the optimization technique used, the admission control policies can be divided into the following:

- Optimal – The optimal CAC policies have been extensively studied using the stochastic control technique based on the Markov Decision Process (MDP). A problem formulated as an MDP can be solved iteratively which converges to a unique optimal policy [33]. The MDP problem for single service systems is computationally complex, for multiple services systems it is even more complicated and expensive. Under such situation, a Semi Markov Decision Process (SMDP) has been applied. A Markov Decision-based Call Admission Control (MD-CAC) is proposed in [34] which maximized the system utilization while ensuring service differentiation and fairness across all call classes. The authors of [35] maximized the revenue by applying the MDP technique to rate-adaptive multimedia traffic to reallocate the system bandwidth to different service classes.

- Near-Optimal – The near - optimal CAC schemes are applied to the system with large number of states which are impractical to model as a Markov process. These near-optimal approaches, also known as intelligent algorithms, are adaptive and flexible, thus making them suitable to cope with the rapidly changing network conditions and bursty traffic that can occur in high-speed wireless networks to give an efficient network management scheme. Examples of some of the intelligent algorithms are described below.

  1) Genetic Algorithms (GA) – The genetic algorithms are directed random search techniques used to look for parameters that provide optimal solution to the problem. They are based on the principles of evolution and natural genetics. The notion of a GA is the survival of the fittest in nature, which implies that the 'fitter' individuals are more likely to survive and have chance of passing their features to the next generation. In [36], a CAC scheme using GA has been proposed for roaming mobile users with low handoff latency for a heterogeneous network of Next Generation Wireless Systems (NGWS). It provides high network utilization, minimum cost, minimum handoff latency and required QoS level. A cost function providing the best network performance with minimum cost is derived using Markov Decision Process Model which is then optimized using the GA determining the final handoff decision and accept-reject action for CAC.
  2) Fuzzy Logic (FL) – The FL techniques resembles the human decision making with an ability to generate precise solutions from

certain or approximate information. It does not require precise inputs, and can process any number of inputs. FL incorporates a simple, rule based approach to solving control problem rather than attempting to model a system mathematically. A FL based CAC for a GPRS network was proposed in [37]. It emphasized the use of FL to avoid the computational complexity introduced by many admission control algorithms and on the capability of the FL to accept rules expressed in natural language, and to combine the rules with facts through fuzzy inference to reach a conclusion. CAC using fuzzy logic for Wideband Code Division Multiple Access (WCDMA) is presented in [38]. This approach takes into account user mobility, limited radio spectrum, heterogeneous and dynamic nature of multimedia traffic, and QoS constraints without complex mathematical relations. The Fuzzy CAC decision for a new connection is based on the resource estimation in the target cell and the neighbouring cell and the mobility information where handoff calls are given higher priority than the new calls. This method gives low new call dropping probability, and high resource utilization efficiency. A method of fuzzy admission control for multimedia applications (MFAC) scheme is proposed in [39]. In this method, for multimedia applications, QoS and Congestion Control (CC) have more parameters and thus two additional fuzzy based controllers: Fuzzy QoS Controller (FQC) and Fuzzy Congestion Controller (FCC) have been introduced to the fuzzy based admission controller, allowing better estimation of QoS and CC parameters.

3) Neural Networks (NN) - Neural Networks have learning and adaptive capabilities that can be used to construct intelligent computational algorithms for traffic control. A neuro-fuzzy approach for CAC with QoS guarantee in multimedia high-speed networks is proposed in [40]. It is an integrated method that combines linguistic control capabilities of a fuzzy logic controller and the learning abilities of a neural network. This scheme provides higher system utilization. A novel learning approach to solve the CAC in multimedia cellular networks with multiple classes of traffic is presented in [41]. The near optimal CAC policy is obtained through a form of NeuroEvolution algorithm. This method guarantees that the specified Call Dropping Rate (CDR) remain under a predefined upper bound while retaining acceptable Call Blocking Rate (CBR).

2.2.1.2.4 Adaptive Multimedia

The development of the adaptive multimedia applications such as the Motion Pictures Expert Group (MPEG) -4, allows the bandwidth of a connection to be dynamically adjusted to the varying link capacity of the wireless network. An adaptive multimedia framework uses a Bandwidth Adaptation (BA) algorithm along with the CAC algorithm in order to provide the QoS provisioning. A novel bandwidth allocation scheme and the call admission control algorithm were proposed in [42] to balance the user satisfaction and the fairness among all the users for the adaptive multimedia services. The proposed algorithm works by assigning the bandwidth to the incoming calls step by step instead of giving them all the requested bandwidth at the very beginning. The dropping of the handoff calls due to insufficient resources is also avoided by degrading the bandwidth of the handoff calls. In order to avoid the inconvenience to the users because of the bandwidth degradation, a compensation mechanism is provided in the CAC scheme which keeps fairness among all the users. A new QoS provisioning method for adaptive multimedia in cellular wireless networks was proposed in [43]. The neural network is used to implement the algorithm and the proposed scheme dynamically adapts to the changes in the traffic condition by considering the status of the neighbouring cells.

2.2.2 Satellite Networks

Satellite systems have been growing popular due to the satellite large geographic coverage and fast deployment as compared to the terrestrial networks. The telecommunication infrastructure could easily be provided using satellites to cover areas where terrestrial network provision is deficient or simply not feasible, for example, maritime communications by InMarsat satellites. The other major advantage exhibited by the satellite systems is their inherent broadcast and multicast capabilities. However, there exist several constraints inherent to the satellite systems such as the high cost of the resources of the satellite and the long propagation delays. Given these issues, the goal of the Radio Resource Management (RRM) for a satellite system is to optimize the bandwidth utilization such that an acceptable quality of service is delivered.

In the recent past, research has been carried out to increase the competitiveness of satellites in terms of the transmission capacity versus the cost and the supported services, with regards to the evolution of the terrestrial technologies. Satellite systems use multi-beams which allow the frequency reuse to increase the transmission capacity. A further increase in the capacity is obtained by allocating a higher frequency band such as Ka band for satellite service provision. However, at such a high frequency band, the attenuation and scintillation effect of the atmospheric gases, clouds and rain become more severe which can induce the use of high system static margins for a given link. Using such high static link margins impairs the cost efficiency and hence, Fade Mitigation Techniques (FMT) is implemented in the physical layer to combat the propagation impairments at high frequency bands. FMT techniques allow the system to be designed with small static margins while overcoming the atmospheric attenuation. One of the popular FMT techniques used is Adaptive Coding and Modulation (ACM) which allows the performance of individual link to be optimized [44]. However, using ACM, the link capacity varies frequently as the physical layer configuration adapts to the time and location dependent channel conditions [45]. At the same time, some lower frequency bands satellites such as C and Ku still use Constant Coding and Modulation (CCM) technique in the physical layer. CCM allows the use of the same physical configuration for all ground stations receiving the same modulated carrier. Hence, the link capacity of such satellites remains constant. The design of an efficient CAC scheme for satellite networks has been extensively dealt with in the literature. This literature study has been divided according to the satellite link capacity:

- Fixed Link capacity – Satellites employing CCM in the physical layer
- Variable link capacity – Satellites employing ACM in the physical layer

2.2.2.1 Fixed Link Capacity

A resource allocation scheme along with CAC for broadband satellite networks over the TDMA based return link is proposed in [46]. Double Movable Boundary Strategy (DMBS) dynamically controls the boundary policy of the resource sharing amongst the different traffic class according to the variable network load conditions. The system supports two types of traffic: Constant Bit Rate (CBR) and Bursty data. The return frame allocator maintains two separate queues for the arriving connection requests for both types of traffic. The queue is considered finite for CBR traffic and is bounded by the maximum call establishment delay. On the other hand, the bursty traffic queue is assumed to be infinite. The return frame consists of a number of timeslots which is divided into three parts. The first consists of time slots assigned for CBR traffic, TS_{CBR}; the second consists of time slots assigned for bursty traffic, TS_B, while the third (the remaining slots) is a Common Resource Pool (CRP), TS_{CRP}, which can be assigned to either type of traffic. CAC for CBR connections is triggered periodically at the beginning of every frame. The CBR requests from the queue are granted time slots in the frame if there are enough unused timeslots available from TS_{CBR}. In case, there are additional requests remaining in the queue, the possibility of admitting the slots from TS_{CRB} is considered. This is based on the value of the data queue threshold, DQ_{TH}, of the queue storing the bursty traffic requests. The DQ_{TH} can be defined statistically in which case the value remains the same irrespective of the state of the system or it can be defined dynamically which allows the boundary of the CBR and CRP position on the frame to be dynamically adjusted to follow the variations in the traffic conditions. If the data queue occupancy is below the DQ_{TH} the waiting CBR calls are granted some extra resources from the TS_{CRP}; otherwise CBR request are denied access to the common resource in which case the requests are made to wait for the release of the resources. If the queue is completely full when a new CBR call arrives, the request will be blocked. It will also get blocked if the waiting time in the queue exceeds a certain maximum call establishment delay.

A predictive CAC algorithm is proposed in [47] for onboard packet switching satellite systems taking into account the unique features of satellite systems, such as limited onboard satellite buffer, and long propagation delays The CAC algorithm is implemented onboard the satellite and executed for each downlink whenever a new connection request arrives. The algorithm performs the online measurements for the established connections. For each traffic source, a sample data of On-Off traffic is collected for a number of active and idle periods. Using the Maximum Likelihood (ML) estimator, the source parameters such as the rate of flow from active to idle period and the rate of flow from idle to active period is estimated. Based on the estimated parameters, the Individual Cell Loss Ratio (ICLR) is predicted ahead of the current time. Two threshold values of ICLR have been defined: $ICLR_{CongestionOnset}$ which is the required ICLR for congestion onset, and $ICLR_{CongestionTermination}$ which is the required ICLR for congestion termination. The value of the predicted ICLR is compared against these thresholds for the admission decision. Under no congestion state, if $ICLR > ICLR_{CongestionOnset}$, the connection is rejected and the system is set into congestion state. Similarly, under congestion state, if $ICLR > ICLR_{CongestionTermination}$, the connection is rejected and the system remains congested. Otherwise the connection is accepted. A measurement based admission control (MBAC) for onboard processing satellite is also proposed by the authors of [48]. Unlike, [47] where the CAC is implemented onboard the satellite, this paper proposes a scalable CAC implemented on ground in Network Control Centre (NCC). The on-board measurements are performed on aggregate traffic rather than on single traffic for each downlink to reduce the computation complexities on the satellite, which are transmitted to the NCC. The CAC is based on the computation of the downlink effective bandwidth which is an estimation of the actual downlink bit rates used by the in-progress connections. Such estimations, based on Kalman filtering, take into account the traffic descriptors communicated by the traffic sources at the connection set-up and the received measures performed on the aggregate traffic. A combined CAC and bandwidth allocation scheme for DVB-RCS system implemented within Return Channel Satellite Terminal (RCST) layer 2 resource manager is proposed in [49]. Different traffic buffers are maintained pertaining to different sources. A connection is accepted in the buffer, if the sum of the peak rate of the given connection and the total bandwidth allocated to each traffic source within RCST does not exceed the total capacity assigned to a given RCST. The bandwidth allocation to each traffic flow is adjusted using control functions such as Reference Chaser Bandwidth Controller (RCBC) and Proportional Integrative Derivative (PID), which estimates the minimum bandwidth necessary at each traffic buffer to provide a given QoS measure, for example the Packet Loss Probability (PLP). An integrated CAC and Bandwidth on Demand (BoD) algorithm for a GEO-based broadband satellite network is presented in [50]. The BoD allows a connection to request resources on a demand basis, while a connection is already in progress. This can be invoked many times during the progress of some types of connections. However, the CAC is invoked only once during the connection set-up to determine if a new connection is admitted or rejected. While admitting a connection, j, a static amount of resource, $SR_{j,}$ is assigned to the connection which may vary from 0 to the peak rate depending on the type of traffic, the traffic descriptor and the requested QoS. The CAC also reserves a certain amount of resource, RR_j which may vary from 0 to ($PeakRate$ - SR_j). Although RR_j is assigned by the CAC, BoD manages this additional resource. If the connection does not request for any additional resource, the BoD controller provides this additional resource, RR_j to any other connection within the same beam on a best effort basis Using such reserved resource allows the delivery of QoS to services.

2.2.2.2 Variable Link capacity

The function of a CAC is to accept or reject a connection request based on the available capacity and the amount of resources needed by the arriving connections. Hence, the channel capacity estimation for a variable link capacity is a major challenge and the development of new CAC schemes is needed. In [51], the author performs separate CAC policies for the uplink and the downlink. The purpose of the CAC policies is to avoid link congestion. In the uplink, the blocking probability is calculated at each Satellite Terminal (ST) using a Proportional Integral Derivative (PID) controller. For each connection, a random number is chosen between 0 and 1, if the number is greater than the blocking probability, the connection is blocked. In the downlink, the CAC policy avoids congestion in the downlink, but also aims at fairness in sharing of the bandwidth between different STs during congestion. Different STs contribute unequally to the congestion which means that applying the same blocking probability would penalize the ST consuming less bandwidth. Hence, a ST specific blocking probability is derived by combining the downlink blocking probability with the consumed bandwidth ratio of each ST. For each connection, a random number is chosen between 0 and 1, if the number is greater than the ST specific blocking probability, the connection is blocked.

The authors of [52] provides an alternative to the work in [51] by replacing the PID controller with a fuzzy controller to derive a self-tuning connection blocking probability to accommodate different network conditions. A self-configuring fuzzy controller allows the use of an intuitive understanding as opposed to the conventional method such as PID, which uses several control parameters and whose effectiveness depends on the settings used for the parameters.

The authors of [53] compares a fixed admission control scheme against an adaptive admission control scheme for direct broadcast satellite systems supporting available bit rate (ABR) and variable bit rate (VBR) traffic. The resource allocated to VBR traffic is based on the statistical multiplexing gain and the work focuses on the capability of the VBR traffic to adjust its overall data rate based on the network conditions such as changes in the channel condition. The Bandwidth Expansion Factor (BEF) is determined such that the probability of the aggregate instantaneous rate of the VBR traffic exceeding the fraction of the resources assigned to the VBR services is not greater than a pre-specified probability value, γ. In the fixed admission control scheme, the BEF value is constant and does not vary with traffic or channel conditions. During the admission process, if the estimated probability value, γ_e, is greater than γ, then this indicates more dropping than estimated. A control action is taken by which the average overall rates of the VBR services, calculated as the source rate divided by the Forward Error Correction (FEC) code rate, are reduced by increasing the FEC code rate. Conversely, if γ_e is less than γ, indicating low utilization the action is taken to increase the average overall rate of the VBR services. The adaptive algorithm provides an improvement over fixed algorithm when the low utilization is detected. Under such conditions, the average overall rate and BEF is corrected by taking into account the bandwidth required for the pending request, B_p. The admission of a pending request is done periodically only if the quality of service of the ongoing services is not violated due to congestion or channel problems.

[54] presented an Admission Limit Curve (ALC) and a Random Capacity Change Time (RCCT) policy for capacity varying networks such as Low Earth Orbit (LEO) satellite system. The ALC defined the conditions under which the CAC policy may admit the connection request. It also ensures a lower bound on the call dropping probability. The dropping scheme proposed by the authors includes Last Come First Dropped (LCFD) which drops the connections in reverse order of acceptance when the capacity reduces. The authors in [55] focused on the impact of FMT on the resource management in

a satellite system with on-board processing capability operating at Ka band. The author proposed an extension of the basic RCCT mechanism reported in [54] to support multiple traffic class. Admitted calls are separated according to the traffic classes and the independent RCCT CACs operate in parallel for each traffic class. Similarly a LCFD dropping scheme is applied to each traffic class producing a list of calls to be dropped. A global dropping strategy is then applied to each of the call list to select the calls to be dropped.

[56-57] proposed a priority based CAC scheme for multi class services in satellite networks operating at Ka band. The system supports two types of services, Real-Time (RT) and Non-Real-Time (NRT) services. RT services are given higher priority than NRT services. It is assumed that the NRT services can use up to a certain portion of the total resources whereas RT services can use full available resource. Under a clear sky condition, the RT and NRT users are admitted into the system, if there are enough resources. However, under rain conditions, additional resources are required to mitigate the rain fading. These additional resources are calculated separately for RT and NRT users based on the outage probability of the user and are defined as the instantaneous fraction of redundant resources required at time, *t,* for error correction. The admission control procedure then takes into account these additional resources while making the admission decision.

[58-60] presented the resource allocation for a fully meshed satellite network which used bent-pipe GEO satellite channels operating at Ka Band. The mobile terminals continuously measure their signal fade level which is communicated to a controller, where the resource allocation technique resides.

In [59], a decentralized control approach was presented where a local admission control resided in each earth station. A master station allocates the total available bandwidth among the various earth stations. This can be made static using “a priori” information based on long-term weather statistics which then assigns a range of fade values to a given earth station. Alternatively, the assignment can be dynamic where the master station has an accurate knowledge of the level of fading for each station. Once the bandwidth has been assigned to the earth station, the admission of the different traffic is executed. In [60], a comparison is made between two centralized bandwidth allocation policies namely, Optimized Centralized (OC) and Optimized Proportional (OP). OC is based on the parametric optimization of a cost function while OP assigns the bandwidth on the basis of requests issued by the mobile terminals. The comparison shows that OC is preferable over OP, but at the expense of a higher computational complexity.

2.3 Pre-emption Algorithms

Mobile networks are designed in a way so that the performance of the system remains satisfactory under a certain maximum traffic load. However, beyond the maximum load, the system performance starts to deteriorate and eventually leading to the network failure. Under such a highly congested state, a pre-emption control mechanism can be triggered to prematurely stop one or more existing connections in order to admit the new connection based on a certain criteria. The conditions under which the high traffic is likely to take place eventually leading to a high congestion state can be broadly classified as follows [61]:

- Expected – These situations occur either regularly, for example, some festive period such as Christmas, or announced beforehand, such as some sport events, or open-air concerts. Such situations allow the network operator to counteract in time, hence minimizing the effects of high congestion on the system.
- Unexpected – These situations are caused by some unexpected event affecting many individuals, for example, major accidents such as railway catastrophes or some natural disasters such as, earthquakes, hurricanes etc. Such situations are unpredictable, hence the network operator cannot be prepared for such events well before in time. However, under such conditions it is important that the network does not fail completely. It should be able to provide services for the victim and rescue groups.

Under a highly congested state irrespective of the cause of the congested state whether expected or unexpected, it is necessary to subside the network failure. In order to achieve such an objective, the network should alter the normal resource management procedures, in particular admission control by including pre-emption control. A pre-emption control mechanism is based on the priority assigned to all connections. These priorities are assigned using certain criteria set by the system. In an emergency situation when the system is congested, upon arrival of a new call of high priority, the call is denied an access to the network. Using only the admission control procedure, the new call is blocked. However, by employing a pre-emption control procedure, the new call is admitted by dropping one or more lower priority connections to free the resources.

The authors in [62] proposed two pre-emption based resource allocation schemes, Last-Come-First-Preempted (LCFP) and Path-Prediction-Based-Preemption (PPBP); that can efficiently support multiple traffics such as voice, video, data etc., in an integrated heterogeneous wireless and mobile network. The traffic is divided into real-time traffic i.e. voice and video, which are delay sensitive and non-real time traffic i.e. data which is delay tolerant. It is assumed that real-time traffic has higher priority over non-real time traffic. Although different classes of real-time traffic have different bandwidth requirements, they are all assumed to have the same priority. An incoming real-time call, either originating or handoff call, can pre-empt ongoing non-real-time calls if there are any in the same cell of network. A higher priority real-time call is accepted in the network if there is enough resource for the call else the non-real time calls are checked for pre-emption. In the LCFP scheme, the order of pre-emption is based on the descending order of time when ongoing non-real-time calls were accepted by the system. Therefore, the last accepted non-real-time call is pre-empted first. On the other hand, in the PPBP scheme, the on-going non-real time call that reaches higher bandwidth network earliest is firstly pre-empted which is calculated using the location information of the non-real-time mobile user.

In [63] the pre-emption mechanism is used in a Traffic Engineering Scheme for Internet Protocol (IP) over a Wavelength Division Multiplex (WDM) optical network. The pre-emption provides the service differentiation by minimizing the blocking probability of higher priority requests. It helps to minimize the number of re-routing and signalling overheads in the network.

The authors of [64] proposed a pre-emption policy combined with the queuing policy and compares it with pure pre-emption policy. Under such scheme, the lower priority calls selected for pre-emption are put into a queue until the resources becomes available. Also, the system performance with single pre-emption scheme is compared with multiple pre-emption schemes where the calls are pre-empted multiple times. Three types of calls are considered, the handoff calls, the high priority calls consisting of emergency calls and the remaining lower priority calls. Two separate queues are maintained for handoff and lower priority pre-empted calls, which are scheduled according to the priority queuing with the assumption that the handoff calls have higher priority than the pre-empted calls. Once the call is resumed from the pre-emption queue, it can be pre-empted again which is the case of multiple pre-emption scheme. To avoid such a situation, a priority change mechanism is applied where the priority of the call is increased once it is resumed from the pre-emption queue such that it cannot be pre-empted again. Such a scheme is termed as a single pre-emption scheme. This scheme also gives more chances for the lower priority users to complete the call without affecting the system performance.

[65] proposed a threshold based pre-emption scheme for cellular network. The purpose of such scheme is to guarantee a certain amount of resources to the lower priority calls while allowing a higher priority call such as emergency calls an immediate access to the network. The amount of pre-emption is decided by a pre-emption threshold value which can be tuned according to the channel occupancy rate and traffic rate. If the number of channels occupied by the higher priority calls is less than the threshold, pre-

emption is allowed otherwise the higher priority call is blocked. Hence, the higher the threshold, the higher the resources used by the higher priority calls.

2.4 Summary

This chapter provides a comprehensive survey on the various admission control and pre-emption control algorithms. The state of the art presented for the CAC indicates that the existing CAC schemes has to be evolved or a new CAC scheme needs to be proposed to handle the new challenges for the ever-evolving technology and hence the networks. In the wire line networks, the admission control procedure was simple and uncomplicated. However, with the development of the wireless and the cellular networks, the role of the admission control became more significant and challenging. The most challenging issue was the mobility of the user and the effect of the wireless channel on the data transmission. CAC was initially developed only for single service environments (voice). However, with the advent of the 3G systems, multimedia services such as voice, video, data and audio with QoS profiles, the existing CAC schemes were not applicable. Hence, more sophisticated CAC schemes were proposed for 3G networks. Alongside, there has been a constant growth in the satellite mobile communication systems. Much research work was presented for the satellites with on-board processing capabilities. Recent works has been done on the satellites with variable link capacity. The work in this thesis focuses on the satellite network supporting unicast and multicast traffic.

Chapter 3 : S-UMTS SYSTEM ARCHITECTURE

3.1 Introduction

Satellite communications networks have evolved greatly from the time the first satellite was launched in 1957. Originally, satellites were designed to carry point-to-point communication. As satellites evolved, with their increasing size and power, the size of the earth station could be reduced, and hence their cost, leading to an increase in number of earth stations. This allowed the inherent feature of the satellite to be used: the transmission of the signal from one point to multi point distributed over a wide area [66]. The main advantages and disadvantages of satellite systems as compared to the terrestrial systems are summarized below [6]:

Advantages:

- *Ubiquitous coverage:* A satellite can provide a large coverage area. This is a significant feature especially in low density areas or over the sea where the realization of terrestrial infrastructures is not viable.
- *Support to mobile users:* A mobile user which is situated in the satellite coverage area can easily communicate with other fixed or mobile users.
- *Broadcast/Multicast capability:* It is possible to provide multicast and broadcast communications without the use of complex multicast routing protocols (such as those used in terrestrial networks).
- *Rapid deployment:* Once a satellite is launched it can immediately reach a high number of users.

Disadvantages:

- *Round Trip Propagation Delay (RTD):* RTD is defined as the propagation delay along a link (from ground to satellite and back to ground). In a satellite system, the data transmission is subjected to long propagation delays due to much longer distance of a link between the satellite and the earth station as compared to the terrestrial system. High RTD values can have severe effect on the performance of the system using real-time applications.
- *Atmospheric effects:* In satellite communications, the signal travels a long distance through different layers of the atmosphere which causes attenuation of the signal. The amount of attenuation depends on the transmission frequency such that signals travelling at a higher frequency (above 10 GHz) are mostly affected by the atmosphere. Atmospheric gases, rain, fog, and cloud are some of the examples of the effect of atmosphere.
- *Channel losses:* In satellite networks, *Bit Error Rate* (BER) is very high due to the atmospheric effects. The *Forward Error Correction* (FEC) coding is mostly employed to reduce the BER at the expense of lower information bit-rate.

Multimedia broadband services have been widely provided by terrestrial communication system. Universal Mobile Telecommunication System (UMTS) is the 3rd generation mobile communication system. UMTS network uses a high bit rate radio technology, Wideband Code Division Multiple Access (WCDMA) as the air interface, which is standardized by the Third Generation Partnership Program (3GPP). UMTS supports both circuit and packet switched connections [67]. Packet switched services provided by the UMTS network was initially based on point-to-point communication. 3GPP then standardised the Multimedia Broadcast Multicast Services (MBMS) framework which provides the delivery of multimedia services based on the point-to-multipoint communication in UMTS [68]. Since the satellites benefit from their inherent nature to multicast/broadcast, the integration of the satellite with the UMTS network has been studied.

INMARSAT BGAN is a UMTS-compatible satellite system. The BGAN system is the first ever 3GPP-based satellite communication network designed to deliver both voice and broadband data services simultaneously through a single, highly compact device on a global basis [69]. BGAN also extended the 3GPP architecture to allow MBMS to be delivered across a mobile satellite network efficiently [70].

This section provides an overview of the BGAN satellite system network architecture which has been considered while designing the proposed CAC framework.

3.2 S-UMTS Network components

The S-UMTS network architecture is derived from the standard Terrestrial UMTS (T-UMTS), however it is important to study the similarities and differences between the two network architectures. Figure 3-1 shows the T-UMTS network architecture which consists of the following network elements:

- **User Equipment (UE):** the combination of the subscriber's mobile equipment and the UMTS Subscriber Identity Module (USIM).

- **UMTS Terrestrial Radio Access Network (UTRAN):** UTRAN uses WCDMA as the radio access technology to connect to the UE. It consists of two network elements: the Node B and the Radio Network Controller (RNC). Node B is the radio transceiver unit for communication between radio cells. Each RNC connects to one or more Node B.

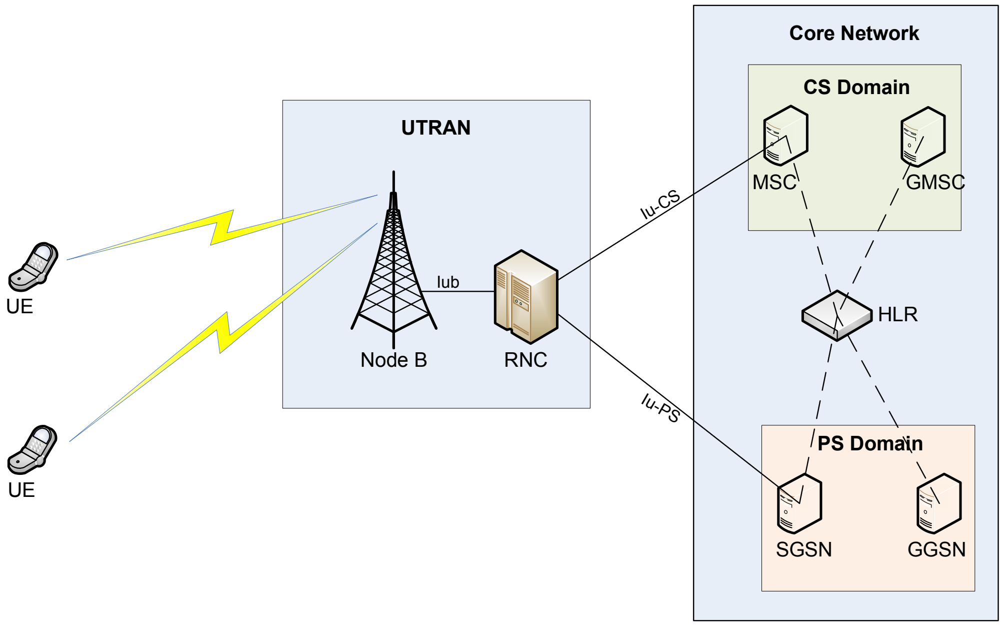

Figure 3-1 Terrestrial UMTS network architecture

- **Core Network (CN):** The CN consists of the circuit switched elements such as the Mobile Switching Centre (MSC), and the gateway MSC (GMSC) and the packet switched elements such as the Serving GPRS Support Node (SGSN) and the Gateway GPRS Support Node (GGSN).

Figure 3-2 shows the network architecture of the UMTS-compatible satellite system. There are several interfaces: *Iub* interface is used to connect the satellite gateway and the RNC within the USRAN*; Iu-CS* is the interface between the RNC and the CS domain for circuit switching data*; Iu-PS* is the interface between the RNC and the PS domain for packet switching data*; Iu-BM* is the interface between the RNC and the BM domain for multicast data*.*

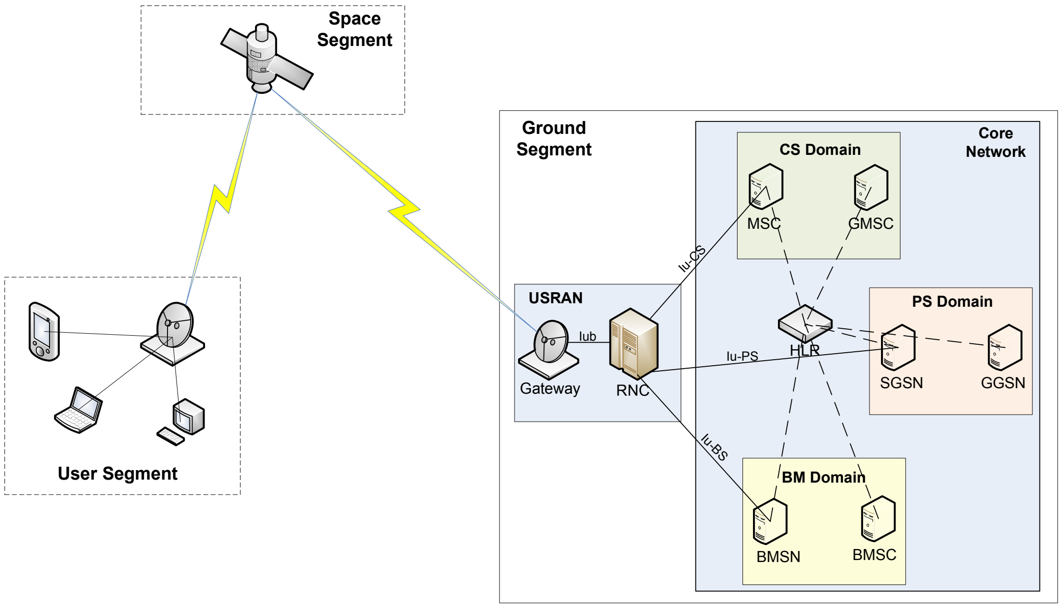

Figure 3-2 UMTS-compatible satellite network architecture [71-73]

The network architecture is similar to the standard T- UMTS but replaces the Node B with the satellite. Nevertheless, there are some similarities and differences between the satellite and terrestrial systems [74].

Similarities:

- Satellite systems make use of spot beams which is similar to the concept of 'cell' in terrestrial systems.
- Satellite systems also make use of frequency reuse amongst the spot beams.
- Both support circuit and packet switched data.

Differences:

- A spot beam covers a much larger area than a cell in the terrestrial system.
- The distance between the user and the RNC in satellite systems is much larger than the distance between the user and the base station in terrestrial system.

- The propagation delay in satellite systems is much longer than that in the terrestrial systems.

It can be seen that the S-UMTS network architecture consists of three entities: user segment, space segment and ground segment. Each of the segments is described in detail in the following subsections.

3.2.1 User Segment

The user segment consists of the user terminal components making up the UE equivalent of the T-UMTS. It is responsible for interfacing the user terminal component to the CN in order to access the services provided by the satellite system.

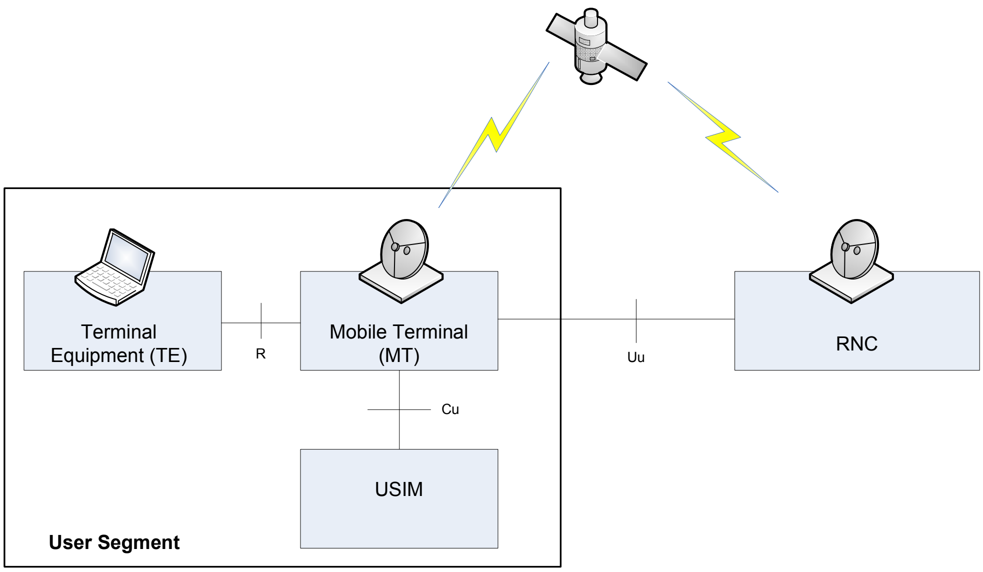

Figure 3-3 S-UMTS user segment interfaces [70-71, 73]

For the given system, the user segment consists of a transportable satellite modem known as the Mobile Terminal (MT) to which is connected a Terminal Equipment (TE) such as a personal computer, PDA etc. and a UMTS Subscriber Identity Module (USIM). The MT works as a communication bridge between the TE and the external networks via the satellite network. The Cu interface indicates the provision of inserting the USIM which must be inserted in the MT prior to use. The TE is the device on which the user applications are running. The connections between TEs and an MT can be setup via the R interface which can either use the wire line or wireless connection such as USB (Universal Serial Bus), Ethernet and Bluetooth. Both control plane signalling traffic and user plane data traffic are supported on the R interface. The Uu interface is the radio interface between the MTs and the RNC. The system supports 3 classes of MTs differing in the size of the antennas: Class1, Class2, and Class3. These support different maximum transmission rates of 432 kbps, 432 kbps and 216 kbps respectively when receiving data and maximum transmission rates of 432 kbps, 144 kbps and 72 kbps respectively when transmitting data [71, 73, 75].

3.2.2 Space Segment

The space segment consists of one or more satellites which provide the communications between the user and the ground segments. The following subsections concentrate on three aspects of the satellites: orbit, frequency band and transponder.

3.2.2.1 Satellite Orbits

Every satellite follows a trajectory around the earth known as orbit. Satellite systems can be classified into the following three types based on their orbital altitude from Earth [6, 66, 76-77]:

Low Earth Orbit (LEO)

LEO satellites orbit closest to Earth at an altitude of around 500-2000 km. As these satellites lie very close to Earth, they must travel very fast to avoid the gravitational pull. The orbital speed of LEO satellites is approximately 27,359 km per hour and it takes around 90 minutes to complete an orbit. Because of their low altitude, the LEO satellites have the smallest propagation delay (less than 10ms from Earth to satellite) and the transmitting stations require much less power as compared to the transmissions with satellites orbiting at a greater distance from the Earth's surface. However, due to the lower altitude a global coverage cannot be provided by LEO satellites. In order to provide global coverage and continuous satellite access, a large number of satellites are required in the network. The management of the network is very complex as these fast moving satellites require frequent handovers between them.

Medium Earth Orbit (MEO)

MEO satellites orbit at an altitude of around 8000-12000 km above the Earth's surface. The orbital periods of MEO satellites range from 2 to 12 hours. Compared to LEO satellites, MEO satellites have a higher altitude and hence there is a slightly longer delay in the propagation of about 20-40ms from Earth to satellite. A global coverage can be provided by using 10-12 MEO satellites. These are mostly used for localization and navigation system such as Global Positioning System (GPS).

Geostationary Earth Orbit (GEO)

A GEO satellite orbits in a geosynchronous orbit at an altitude of 35,786 km with an orbit period of 23 hours 56 minutes and 4.1 seconds. This is the actual time taken by Earth to rotate about its axis. Hence a GEO satellite appears to be stationary with respect to a point on Earth. Apart from the Polar Regions, the entire Earth can be covered using just three GEO satellites. A major disadvantage of GEO satellites is the high propagation delay of 120-135ms from Earth to satellite because of their high altitude.

3.2.2.2 Satellite frequency bands

Satellite communications employ electromagnetic waves to carry the information between ground and space. A particular range of frequencies is called a band. Although the most useful Radio Frequencies (RF) lie in the 300 MHz to 30GHz range, the optimum range of spectrum for space- to-earth communications lies between 1GHz and 30GHz. The propagation grows in difficulty due to higher levels of atmospheric attenuation effects above 30 GHz and the manmade noise becomes significant below 1GHz [3]. The main IEEE frequency bands used for satellite communications system are listed below [78-79]:

- L band is the range of frequencies between 1 and 2 GHz. This is mainly used for Mobile Satellite Service (MSS).
- S band is the range of frequencies between 2 and 4 GHz. This is also used for MSS.
- C band is the range of frequencies between 4 and 8 GHz. This is mainly used for Fixed Satellite Service (FSS).
- X band is the range of frequencies between 8 and 12 GHz. This is mainly used for military applications.
- Ku band is the range of frequencies between 12 and 18 GHz. These are mainly used for FSS and BSS purposes.
- K band is the range of frequencies between 18 and 26 GHz.
- Ka band is the range of frequencies between 26 and 40 GHz. This is mainly used for FSS and MSS.

- V band is the range of frequencies between 40 and 75 GHz. These bands are mainly used for military FSS.

FSS refers to a two way communication between two earth stations at a fixed location via satellite. It primarily uses two frequency bands, C and Ku. C band provides lower power transmission over a wider region requiring large receiving antennas. Ku band on the other hand, provides higher power transmission over a narrow region requiring a small receiving antenna.

BSS refers to a one way communication by which the satellite services are received by many earth stations. The Ku and Ka band are mainly used for BSS applications such as television broadcasting and Direct – To-Home (DTH) applications.

MSS refers to the two ways communication between earth stations that are in motion such as ships, cars, lorries etc. L band and S band are mainly used for MSS applications. At these lower frequencies, broader beams are transmitted from the satellite, enabling the reception by antennas even if they are not pointed towards the satellite.

3.2.2.3 Satellite transponder

A satellite transponder is a series of components that allows a communication link between the uplink channel (from the earth) and the downlink channel (to the earth). A typical communications satellite may contain several transponders [80]. A transponder is implemented in one of the two general types of configurations

- Frequency translation transponder also referred to as a non-regenerative repeater or bent-pipe. In this configuration, the uplink signal is amplified and retransmitted with only a translation in the carrier frequency. The uplink and downlink are co-dependent such that any degradation introduced in the uplink is also transferred to the downlink.

- On-board Processing transponder also referred to as regenerative repeater. In this configuration, the uplink signal is demodulated to the baseband signal which is then processed on-board. The baseband signal is the re-modulated to the download carrier and after final amplification, the download signal is transmitted on the ground. The uplink and downlink are independent since the modulation/re-modulation process removes the uplink noise.

On-board processing satellites are more complex and expensive than frequency translation satellites. However, they provide significant performance advantages.

In this thesis, a bent-pipe GEO satellite like INMARSAT-4 satellite, with no on-board processing is considered. The transmission between the RNC and the satellite is via C band whereas the transmission between the satellite and MT is via L band. The air interface consists of the Time Division Multiplex (TDM) and Time Division Multiple Access (TDMA) schemes in the forward direction and return direction respectively [73].

3.2.3 Ground Segment

The ground segment of a satellite system is responsible for monitoring/controlling the satellites, providing services and distributing information to the users. The S-UMTS ground segment consists of two main domains: the UMTS Satellite Radio Access Network (USRAN) and the Core Network (CN). The main task of the USRAN is to create and maintain Radio Access Bearers (RABs) for the data transmission between TE and CN via satellites. The CN of S-UMTS uses the standard 3GPP CN defined for T-UMTS which consists of two main domains: CS and PS domain to support the voice and the data transmission respectively [81-83]. In addition for the support of the multicast transmission, the INMARSAT specific BM domain is also included. The following entities belong to the three domains in the CN:

- **Mobile-services Switching Centre (MSC)**: The MSC has the switching and signalling function for TEs located in its coverage area. An MSC can route calls to the appropriate RNC, perform handover and communicate with other fixed networks.
- **Gateway MSC (GMSC)**: This is a special type of MSC. It connects to the telephone networks and routes the call to the MSC where the TE is located.
- **Serving GPRS Support Node (SGSN)**: This monitors the TE's location and sets up the connection between TE and the packet data networks using Packet Data Protocol (PDP).
- **Gateway GPRS Support Node (GGSN)**: This connects external packet switched networks to SGSNs.
- **BGAN Multicast Service Node (BMSN)**: This is an INMARSAT operated entity which emulates an integrated SGSN and GGSN with the necessary modifications to support the IP-multicast service.
- **BGAN Multicast Service Centre (BMSC)**: This consists of the multicast content provider.

- **Home Location Register (HLR)**: This is a database in charge of the management of TEs.

3.3 Protocol Architecture

This section describes the protocol architecture for the S-UMTS air interface. The work presented in this thesis is generic and is applicable to the satellite systems with variable link capacity such as DVB-S2 and Inmarsat BGAN. However, for the study of this thesis, the INMARSAT proprietary air interface, “INMARSAT Air Interface 2 (IAI-2)”, has been used. Although the S-UMTS network architecture is similar to the terrestrial UMTS network architecture, they differ in the way the data is transferred over the air interface. In order to support the transfer of data over the satellite in S-UMTS by ensuring the attributes of the satellite link such as high delay, variable error rate are countered, the existing air interface of the T-UMTS needs to be modified.

3.3.1 Terrestrial UMTS WCDMA air interface protocol stack

Figure 3-4 shows the protocol architecture of the terrestrial UMTS system air interface.

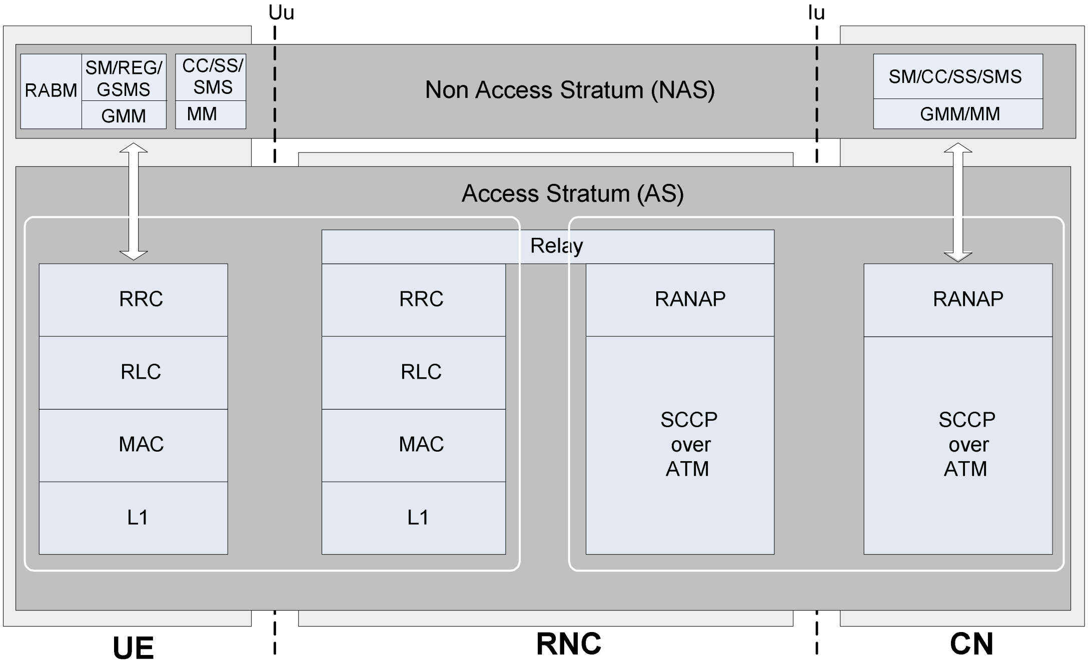

Figure 3-4 Protocol architecture of the T-UMTS WCDMA air interface

The WCDMA air interface technology is split into the following two functionalities [67]:

1) **Non Access Stratum (NAS)**: It groups together all protocols between the user and the CN. Some of the protocols of the NAS are:
   - GPRS Mobility Management and Mobility Management (GMM/MM): GMM supports mobility management functionality such as attach, detach, security, and routing area update for PS domain while MM supports for CS domain.
   - Session Management (SM): SM supports PDP context activation and deactivation. In CS domain, this is provided by the Call Control (CC).
2) **Access Stratum (AS):** This is divided into a *Uu* Stratum, which groups all protocols between the user and RNC, and an *Iu* Stratum, which groups all protocols between the RNC and CN.

The *Uu* stratum consists of the following protocols:

- Radio Resource Control (RRC): exists in the control plane only and provides information transfer service to NAS.
- Radio Link Control (RLC): offers logical link control over the radio interface for the transmission of higher layer signalling and Short Message Service(SMS) messages.
- Medium Access Control (MAC): provide priority handling between data flows and offers data transfer to RLC and higher layers.
- Physical Layer (L1): performs responsibilities associated with transmitting and receiving signals over wireless media.

The *Iu* stratum consists of the following protocols:

- Radio Access Network Application Protocol (RANAP): encapsulates and carries higher-layer signalling, handles signalling between the 3G-SGSN and UTRAN, and manages the GTP (GPRS Tunnelling Protocol) connections on the *Iu* interface.
- Signalling Connection Control Point (SCCP): routing protocol which provides reliable delivery of packets between end stations in a circuit-switched Public Land Mobile Network (PLMN).

3.3.2 Satellite UMTS air interface protocol stack

In order to introduce the satellite between the user and the RNC, the protocols on the *Uu* interface of the AS are modified as shown in Figure 3-5 [84] .

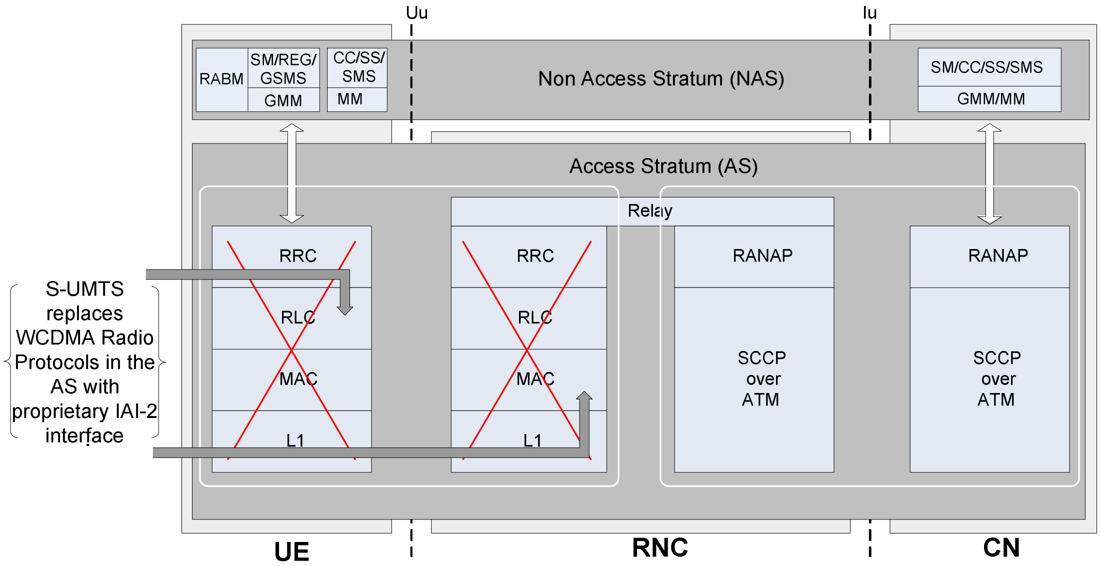

Figure 3-5 Modifications to the WCDMA protocols for S-UMTS [84]

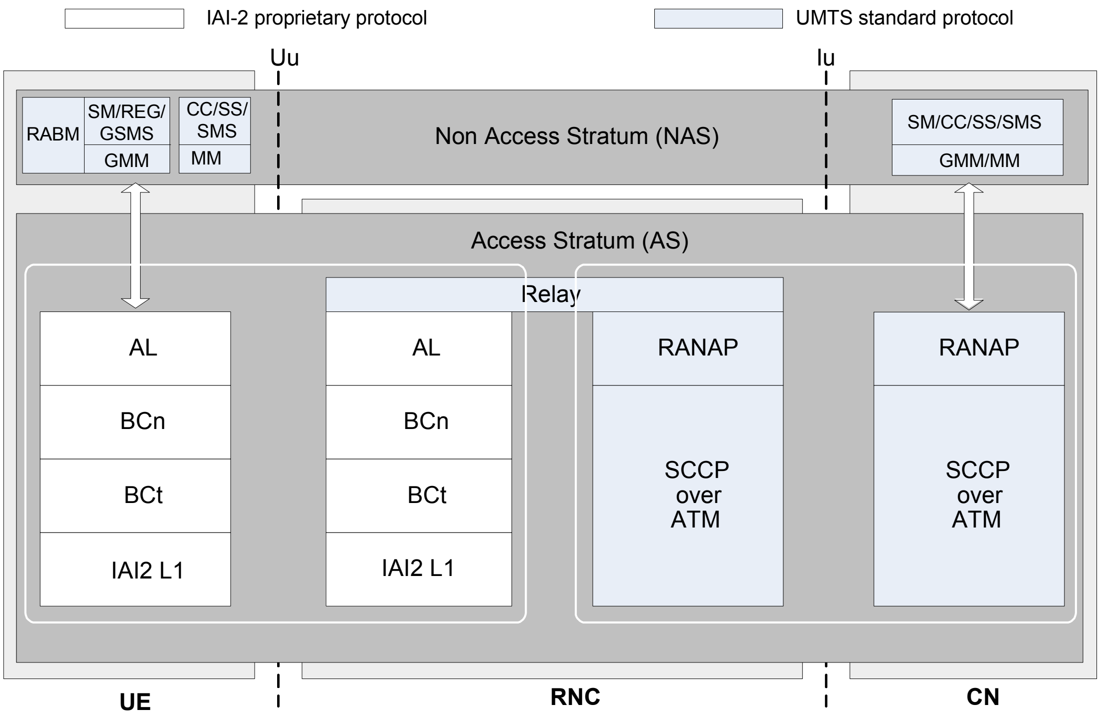

Figure 3-6 S-UMTS air interface protocol stack [84]

As it can be seen in Figure 3-6, the standard UMTS NAS layer signalling is supported over S-UMTS air interface. The new protocol stack consists of the following protocols:

- Adaptation Layer (AL)
- Bearer Connection Layer (BCn)
- Bearer Control Layer (BCt)
- Physical Layer (IAI2 L1)

3.3.2.1 Adaptation Layer

The Adaptation layer allows the lower layers of the terrestrial UMTS; RLC, MAC and Physical layer to be replaced by the equivalent S_UMTS air interface [85] protocol layers. It is a direct replacement of the RRC in the WCDMA protocol stack and performs many of the same functions as the RRC, namely:

- Interpretation of system information related to the AS and forwarding of system information related to the NAS;
- Spot beam selection (initial and re-selection);
- Establishment, maintenance, and release of UE specific Signalling (UESS) connection between the UE and the RNC;
- Establishment, maintenance, and release of radio bearer connections;
- NAS message transport;
- Connection mobility functions (handover);
- Integrity protection and control of ciphering;
- Paging.

3.3.2.2 Bearer Connection Layer

The Bearer Connection (BCn) layer provides a number of different data transport services to the upper layers which are used to carry the control and data signalling across the air interface [73, 86]. These transport services are provided to the upper layer via different Service Access Points (SAP). Three main data transport modes are supported; Acknowledged Mode (AM), Transparent Mode (TM), and Unacknowledged Mode (UM). The main functions provided by BCn layer are similar to those provided by the RLC in the WCDMA protocol stack namely [86]:

- Buffering and flow control of information from the interface to the layer above, AL;
- QoS policing ;
- Segmentation (of packets from AL) and reassembly (of packet segments from BCt) of information entering BCn;
- Automatic Repeat Request (ARQ) – retransmission of missing segments for AM traffic;
- Ciphering of data;

3.3.2.3 Bearer Control Layer

The Bearer Control (BCt) layer provides a mechanism for transporting data from various bearer connections over the physical bearers. It consists of a number of Bearer Control processes, each controlling a number of associated forward and return bearers. Typically one RNC Bearer Control process will control all the channels used in one spot beam. However, more than one control process per beam is allowed. In order to guarantee the fairness between established connections and optimize physical resource utilization, Radio Resource Management (RRM) functions are provided in the BCt layer. The main functions performed in the BCt layer include the following [87]:

- Selection of the appropriate Primary Shared Access Bearer (PSAB) for initial access to the RNC;
- Perform RRM techniques such as admission control, scheduling of resources in forward and return directions, and link adaptation, in order to maintain the QoS of each connection;
- Requesting and releasing of physical resources
- Perform sleep mode operation to allow UE to conserve power.

3.3.2.4 Physical Layer

The physical layer is responsible for transferring information bit stream via the satellite link. In order to optimize the satellite resources, the channels are utilized in the forward direction on a TDM basis, and in the return direction on a TDMA basis. Due to the limitations of the channel allocation mechanisms and the capabilities of the mobiles, channels are also limited in bandwidth, such that the resources are additionally operated in both directions on a Frequency-Division Multiple-Access (FDMA) basis. The air interface operates with multiple physical bearer configurations. Four types of forward bearer are considered in the system: F80T0.25Q1B, F80T1X4B, F80T4.5X8B, and F80T1Q4B based on the choice of modulation schemes as well as symbol and coding rates. These forward bearers are capable of carrying nominal data rates in the range between 4.5 kb/s and 512 kb/s [88].

3.4 Summary

This chapter described the system architecture for UMTS based satellite system. It is shown that the S-UMTS system is derived from the standard terrestrial UMTS system by replacing the base station with the satellite. A detailed description of the different entities comprising the S-UMTS system architecture such as user segment, space segment and the ground segment is presented. The chapter also focuses on the protocol architecture of the S-UMTS air interface. T-UMTS uses the WCDMA technology in the air interface. However, by replacing the base station with the satellite, a new set of protocols have been presented ensuring the unique characteristics of the satellite such as long delay and high bit error rate are considered for the transfer of the data.

Chapter 4 : CAC FRAMEWORK MODEL

4.1 Introduction

CAC is an integral part of the RRM which plays an important role in admitting the connection and ensuring the QoS requirements are met. This chapter proposes a CAC framework for a S-UMTS network such as BGAN, which satisfies the following RRM general requirements [89]:

1. The RRM algorithms shall support the following UMTS traffic classes:
    - Streaming
    - Interactive
    - Background;
2. The RRM algorithms shall support simultaneous operation of different types of service to the same terminal and to different mobile terminals with different capabilities;
3. The software architecture in which RRM algorithms are deployed shall allow support for different physical bearer types and the introduction of new bearer types without requiring a complete redesign;
4. Admission control algorithm should support service prioritisation and pre-emption schemes;
5. The RRM algorithms shall be able to support one or more Multicast Radio Access Bearer Services simultaneously;
6. The RRM algorithms shall be able to support Multicast Radio Access Bearer Service and Unicast Access Bearer Services simultaneously from the same MT.

4.2 CAC Framework Functional Model

The purpose of the basic admission control procedure irrespective of the underlying network is to admit any incoming connection such that it's specified QoS is satisfied without affecting the agreed QoS of the existing connections. A basic CAC model can be represented as shown in Figure 4-1.

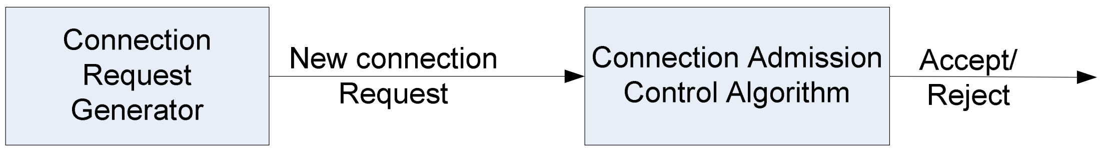

Figure 4-1 Basic CAC model

The Connection Request Generator statistically generates connection requests for one or more traffic sources. It also acts as a traffic descriptor providing a set of parameters characterizing the traffic source. This information is sent in the *New Connection Request* to the Connection Admission Control (CAC) algorithm. The CAC algorithm is an event triggered function which is initiated whenever a new connection request arrives. It makes the decision to accept or reject the connection request based on certain admission criteria such as that the QoS of the existing connections is not affected.

For the S-UMTS system under consideration, a novel CAC framework has been proposed to satisfy the requirements for the admission control procedure listed in Section 4.1. Figure 4-2 illustrates the proposed CAC framework functional model. The functionality of the Connection Request Generator remains the same as that for the basic CAC model shown in Figure 4-1. However, the simple connection admission control algorithm is replaced with a framework consisting of 3 different functionalities namely; CAC Processor, Retune Controller and Pre-emption Controller.

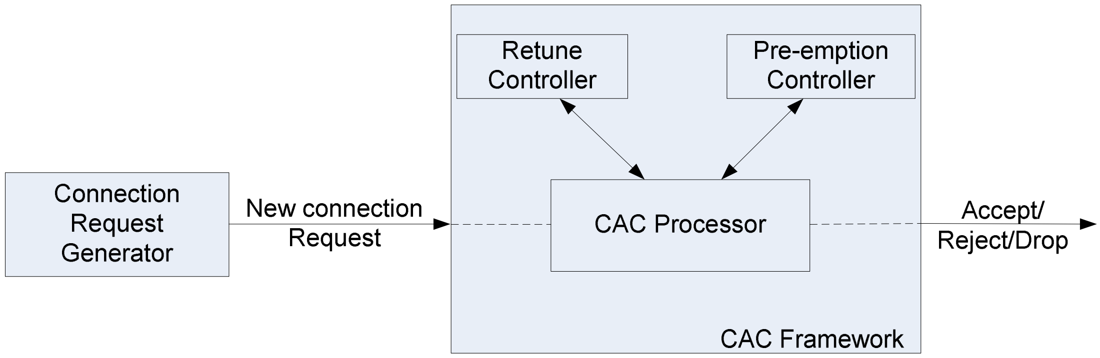

Figure 4-2 CAC Framework functional model

The CAC Processor is central to the CAC framework. It runs the admission control algorithm when triggered by a new connection request and decides whether the new connection can be admitted if there are enough resources. In case the resources are not sufficient on the given forward subband, the possibilities of admitting the new connection by the retuning the new connection from one subband to another or by pre-empting some of the existing connections is checked based on some pre-defined criteria. Hence, the output of the CAC Processor not only results in an *Accept*, or a *Reject*; but it can also result in an *Accept* by *Drop* of the existing connections. The role of the Retune Controller is to investigate the possibility of retuning the connections between different forward subbands such that the new connection can be admitted on the given forward subband. However, if all the available subbands are densely filled, the possibility of admitting the new connection by retuning does not exist; the CAC Processor then triggers the Pre-emption Controller. Based on the pre-emption policy used, one or more, lower precedence ongoing connections are pre-empted in order to admit the higher precedence new connection.

Figure 4-3 shows the overview of the flow of procedures in the CAC framework.

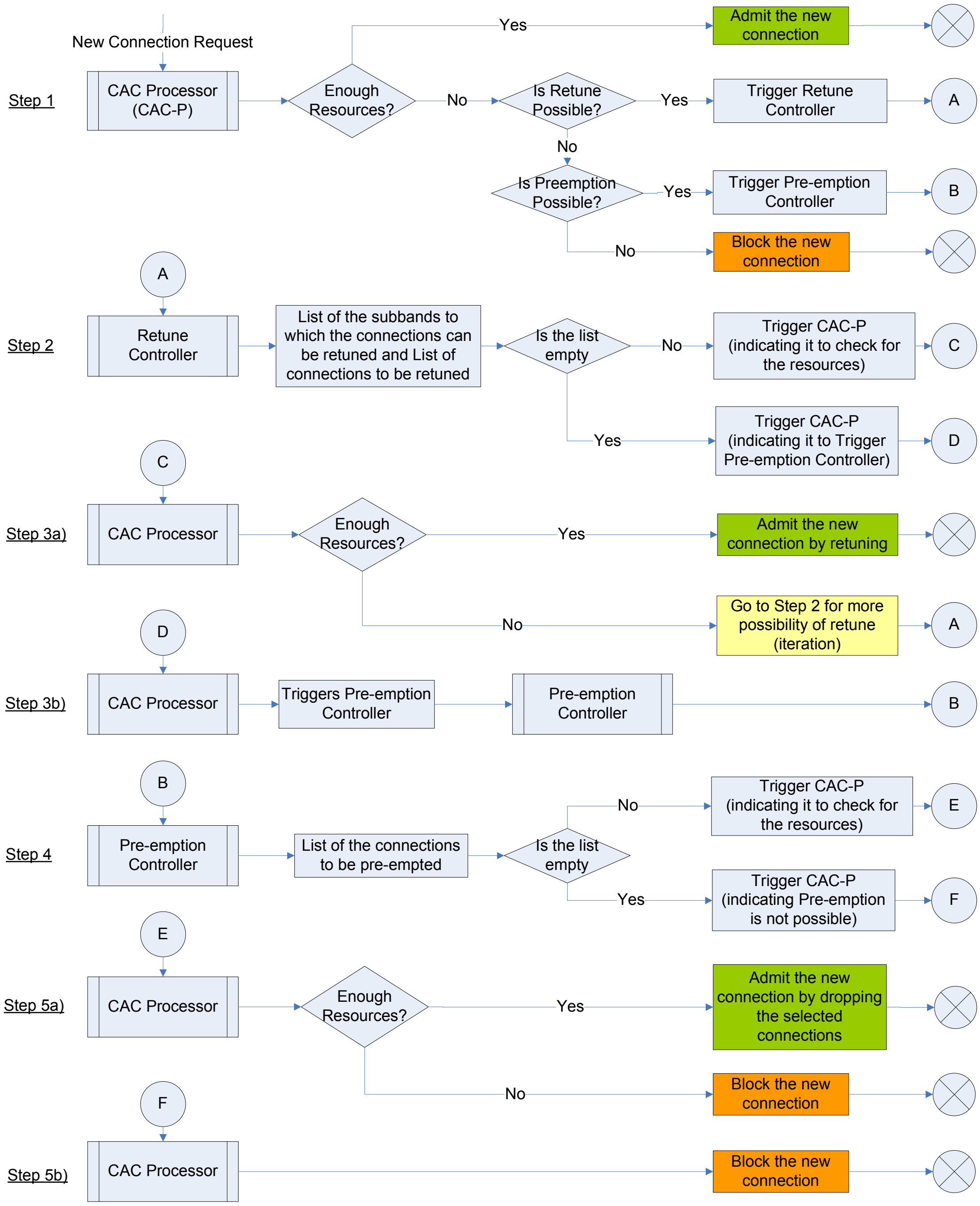

Figure 4-3 Flow of procedures in the CAC framework

The following steps describe the flow of procedures:

1. The CAC Processor is triggered by a *NewConnectionRequest* on a given forward subband. If there are enough resources, the connection is admitted. In case the resources are not sufficient, the possibility of

retuning and then the pre-emption is checked based on some pre-defined criteria. The new connection is blocked if the retuning or the pre-emption is not possible.

2. The CAC Processor triggers the Retune Controller which forms a list of the subbands to which the connections can be retuned and also the list of the connections which can retune. The Retune Controller responds back to the CAC Processor with the selected forward subband from the list or indicates that the list is empty.
3. The CAC Processor processes the list from the Retune Controller
    a. If the list is not empty, the CAC Processor checks for the resources on the selected forward subband. If there are enough resources, the new connection is admitted; else, Step 2 is repeated for another forward subband from the list.
    b. If the list is empty the Pre-emption Controller is triggered.
4. The CAC Processor triggers the Pre-emption Controller which forms the list of the connections to be pre-empted. It responds back to the CAC Processor with the list.
5. The CAC Processor process the list from the Pre-emption Controller
    a. If the list is not empty, the CAC Processor checks if there are enough resources to admit the new connection by dropping the selected connections from the list. If case the resources are sufficient, the new connection is admitted and the selected connections are dropped. Otherwise, the new connection is blocked.
    b. If the list is empty the new connection is blocked.

Each block of the CAC framework; the CAC Processor, the Retune Controller, and the Pre-emption Controller along with the Connection Request Generator block is described in detail in the following subsections.

4.2.1 Connection Request Generator

The Connection Request Generator is responsible for generating new connection arrivals. This section describes the type of multimedia traffic supported, the arrival process assumed for the traffic generation and finally the traffic model of the sources considered in the system.

4.2.1.1 Classification of the supported traffic sources

The system supports two modes of data transmission; unicast and multicast. The unicast mode of transmission involves point-to-point communication between the MT and the RNC. However, using the multicast transmission, the data can be sent from the RNC to a group of MTs in a single transmission. Irrespective of the type of the data transmission used, the QoS attributes associated with a traffic source type remains the same. Four different UMTS QoS classes also known as traffic classes are defined in 3GPP technical specification [90]. They are listed below with the conversational class having the most stringent QoS requirements and the background class having the lowest QoS requirements [91]:

1. Conversational class (e.g Voice)
2. Streaming class (e.g Video streaming)
3. Interactive class (e.g. Netted Voice[2], Web browsing)

[2] Netted Voice refers to the push-to-talk service.

4. Background class (e.g. email)

For the system model, three traffic classes are supported: the streaming class, the interactive class and the background class. Although, the traffic classes serves both Circuit Switched (CS) and Packet Switched (PS) connections, only PS connections are considered in the scope of this work. For the interactive traffic class, a UMTS QoS attribute known as Traffic Handling Priority (THP) is used to identify the priority of the connections within the interactive class. The THP parameter is only applicable for the interactive class and it can take three values: THP1, THP2, THP3, depending on the type of application which follows the following priority order: THP1>THP2>THP3. For the unicast mode of transmission, three classes of service are supported: the steaming class, the interactive class with THP1 and THP2, and the background class. For the multicast mode of transmission, only two classes of service are supported: the streaming class and the background class [73]. Table 4-1 provides the summary of the type of applications supported by the system.

Application Type	Mode of Transmission	Traffic Class	THP	Priority	Pre-emptable
Video Streaming	Multicast	Streaming	N/A	1	No
File Distribution	Multicast	Background	N/A	1	No
Video Streaming	Unicast	Streaming	N/A	1	No
Netted Voice	Unicast	Interactive	1	2	Yes
Web Browsing	Unicast	Interactive	2	3	Yes
Email	Unicast	Background	N/A	4	Yes

Table 4-1 Type of applications supported by the system

Depending on the type of the application, a parameter, *Priority*, is associated with each application which is used in the pre-emption process to decide the order in which the connections can be pre-empted such that the connection with *Priority* value at 4 is more likely to be pre-empted than the connection with *Priority* value 3 and so on. This parameter is based on the QoS requirements of the traffic class for example the streaming class has higher QoS requirements in terms of delay constraint than the interactive class and hence it is given a higher priority. Within the interactive class, netted voice application is assigned a higher priority than the web browsing since the THP parameter of netted voice is higher than that of web browsing. It is assumed that the multicast connections and the video streaming unicast connections cannot be pre-empted and, hence, all these connections are assigned the highest *Priority* with a value of 1. The parameter, *Pre-emptable*, indicates the pre-emption capability of the connection. As can be seen, only the connections of the type interactive and background traffic class are allowed to be pre-empted.

4.2.1.2 Call Arrival process

The call arrival process for each type of the traffic source is modelled as a Poisson process with an exponentially distributed inter-arrival time which is the time between two consecutive connection request generations. Also, the connection holding time, the time for which the connection is active in the system, is exponentially distributed [92].

Figure 4-4 shows the connection request generation at time T_1, T_2, T_3, T_4, and T_5 according to the Poisson distribution for a traffic source with a mean arrival rate, λ.

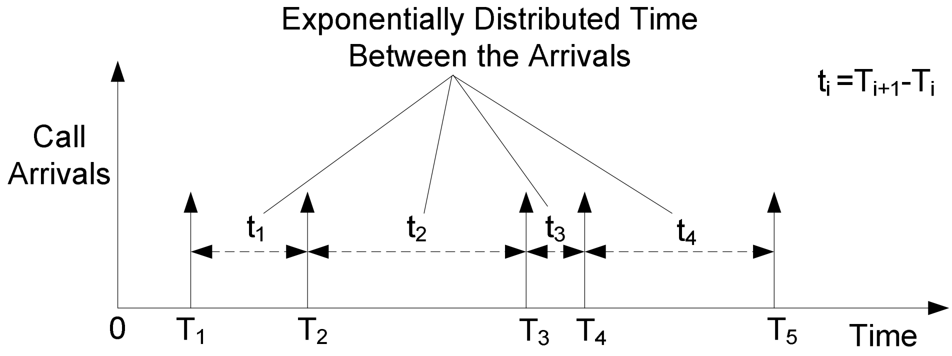

Figure 4-4 Poisson Distributed Call Arrival Process

Since the inter-arrival times, t_1, t_2, and so on are exponentially distributed, the occurrence of the connection arrivals can be calculated as follows:

The Cumulative Distribution Function (CDF) of an exponential distribution is defined as, $F(t) = 1 - e^{-\lambda t}$. Let R represents a random number over the interval [0, 1]:

$$\begin{aligned}
&F(t) = 1 - e^{(-\lambda t)} = R \\
&e^{-\lambda t} = 1 - R \\
&-\lambda t = \ln(1 - R) \\
&t = \left(-\frac{1}{\lambda}\right)\ln(1 - R)
\end{aligned} \tag{4.1}$$

Thus, Equation (4.1) is used to generate inter-arrival times samples, t_1, t_2, and so on based on the mean inter-arrival time, $1/\lambda$.

4.2.1.3 Traffic model

Each traffic source is modelled as an ON-OFF process whereby the burst period (ON) is followed by a silent period (OFF). During the ON mode, the packets are transmitted at Peak Rate (PR) whereas in OFF mode, no packets are transmitted [93]. A simple ON-OFF traffic model for the connection session is shown in Figure 4-5.

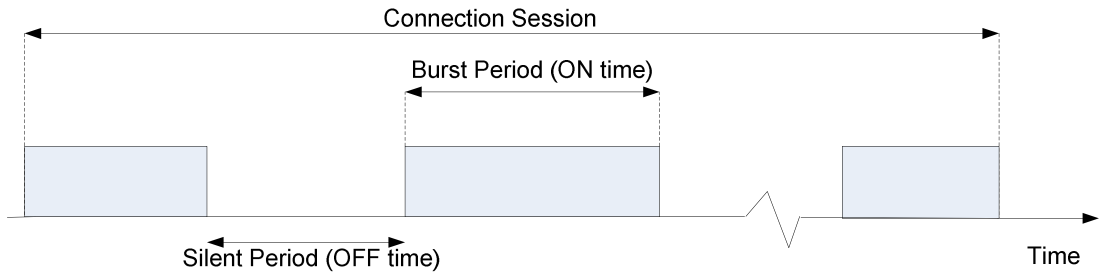

Figure 4-5 A simple ON-OFF traffic model

4.2.2 CAC Processor

The CAC Processor is the predominant functional entity of the CAC framework shown in Figure 4-2. It is an event driven function which encompasses the admission control algorithm. The algorithm is executed at the arrival of a new connection request and decides whether the connection request can be admitted. Two admission control algorithms are presented which are tested against each other in the given system:

- Non-adaptive CAC algorithm
- Adaptive CAC algorithm.

The difference between the two algorithms lies in the calculation of the resource consumed by a connection. In the proposed adaptive CAC algorithm, the algorithm adapts to the changing link condition of the user which may vary depending on the weather condition, user mobility etc. It also considers the class of the MT. The system supports three classes of MTs: Class1, Class2, and Class3, which differ in their capability to transfer the data. In contrast to the adaptive CAC, the non-adaptive CAC algorithm does not consider these parameters. It only checks the available capacity to decide if the new connection request can be admitted in the system. The two algorithms are explained in more detail later in Section 4.2.2.1.3 while describing the method to calculate the total resources used by the active connections.

The following subsections describe in detail the functionality of the CAC Processor and the flow of the admission control algorithm.

4.2.2.1 CAC Processor Functional Block Diagram

The functionality of the CAC Processor involves the following tasks:

- Registering the new connection into the system.
- Performing the admission control algorithm based on Complete Sharing (CS) technique where different types of traffic classes share a common pool of resources [94]. The algorithm requires measurements of the following:
    - The bandwidth required by the connection using the characteristics of the traffic source to which the connection belongs.
    - The total amount of resources used by the on-going active connections in the system.

In order to perform the above mentioned functions, the CAC Processor can be presented as a functional block diagram shown in Figure 4-6.

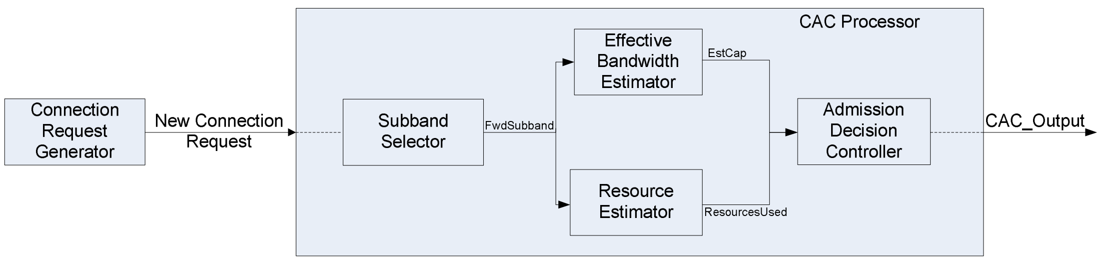

Figure 4-6 Functional Block Diagram of CAC Processor

The following are the functional blocks of the CAC Processor:

- Subband Selector – registers the new connection by selecting a forward subband, *FwdSubband*, for the new connection from a list of available forward subbands.
- Effective Bandwidth Estimator – estimates the effective bandwidth, *EstCap*, required by the connection based on the various traffic characteristics such as traffic class, priority, QoS value of the connection etc.
- Resources Utilization Estimator – calculates the total resources used, *ResourcesUsed*, by the active connections on the given forward subband.
- Admission Decision Controller – produces the output, *CAC_Output*, which may indicate admission, rejection, retuning or pre-emption of connection/connections.

The following subsections describes each of the functional block of the CAC Processor in detail.

4.2.2.1.1 Subband Selector

The Subband Selector is responsible for performing the registration of a new MT into the system by selecting a forward subband from the list of available subbands.

Figure 4-7 depicts an example where an MT can generate one or more connections depending on the number of Terminal Equipments (TEs) connected to the MT and the applications running on each TE. As can be seen, TE1 and TE2 are connected to the same MT via Conn1, Conn2, Conn3, and Conn4 respectively. Hence, when the MT sends its first connection request, for example Conn1, for admission to the network, the

Subband Selector registers the new MT to the system by selecting a forward subband. Any subsequent requests for the connection admission by the same MT will be allocated the same forward subband by the Subband Selector.

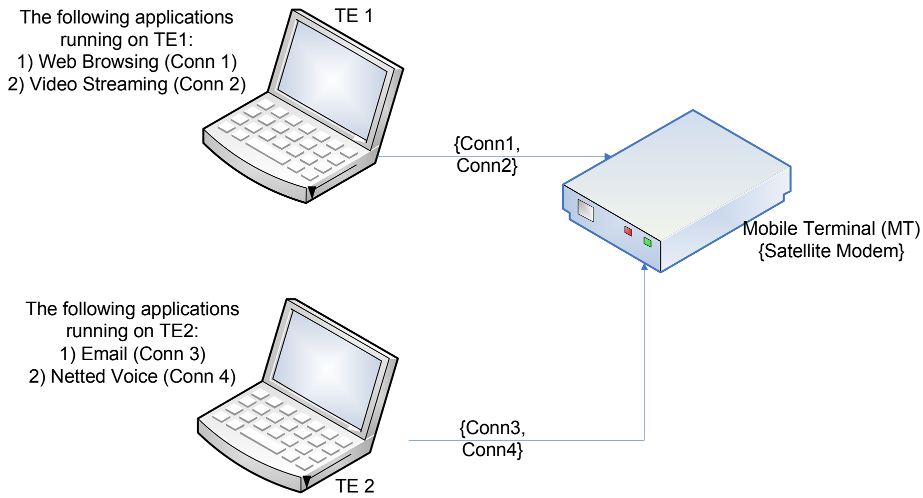

Figure 4-7 Illustration of one or more connections belonging to the same MT

Figure 4-8 summarizes the flow of the procedure in the Subband Selector.

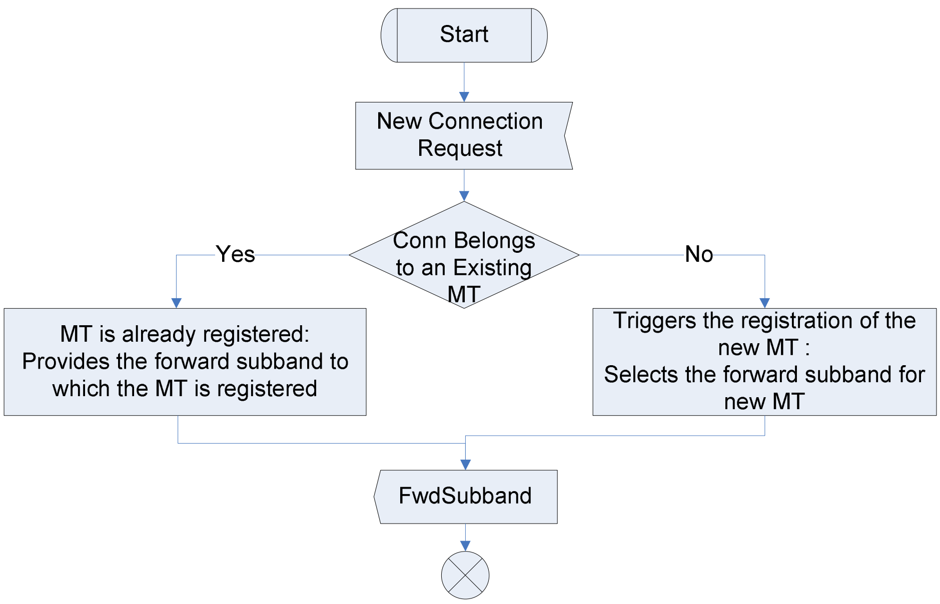

Figure 4-8 Forward subband selection process in Subband Selector

In order to select the forward subband from a list of available subbands, the following two methods have been considered:

- *MinConnSubSel* – Using this method, the forward subband with the minimum number of active connections is selected from the list of available subbands. Such selection aims to evenly spread the traffic on the available subbands and hence, performing a simple load balancing between the subbands.
- *Random* – This method randomly selects a forward subband from the list of available subbands.

4.2.2.1.2 Effective Bandwidth Estimator

The Effective Bandwidth Estimator estimates the bandwidth requirement of the connection based on their statistical characteristics, *EstCap*. As mentioned in Section 4.2.1, all traffic sources are modelled as ON-OFF process, the following parameters are used to calculate the effective bandwidth of the connection:

- Mean Burst Period (t_{ON}): represents the period during which the source is active.
- Source Utilization (ρ): represents the fraction of the time the source is active. It is calculated as follows:

$$\rho = \frac{t_{ON}}{t_{ON} + t_{OFF}} \qquad (4.2)$$

- Peak Rate (R_{peak}): represents the peak rate of the connection.

In order to calculate the Effective Bandwidth of the connections, the two most popular methods in literature are considered [95]:

1) Mean Bit Rate – Using this technique, the effective bandwidth is calculated as the mean sending rate of each type of traffic source and is given by:

$$EstCap = \rho * R_{peak} \tag{4.3}$$

2) Equivalent Capacity – this is based on the mathematical model using fluid flow approximation [7]. The approximation assumes the single source feeding a finite capacity queue of size K, then the equivalent capacity or the effective bandwidth of the source is the service rate of the queue that corresponds to the cell loss of ε and is calculated using Equation (4.4).

$$EstCap = \frac{a - K + \sqrt{(a-K)^2 + 4Ka\rho}}{2a} R_{peak} \tag{4.4}$$

where,
$$a = \ln\left(\frac{1}{\varepsilon}\right) t_{ON} \left(1 - \rho\right) R_{peak}$$

4.2.2.1.3 Resources Utilization Estimator

The function of the Resources Utilization Estimator is to calculate the total resources used on the given forward subband. Depending on the type of the admission control algorithm, adaptive or non-adaptive, the total used resources vary. As explained previously in Section 4.2.2, the adaptive CAC algorithm takes into account the link condition and the class of the MT while calculating the resources used by the connections on a forward subband as opposed to the non-adaptive CAC algorithm which does not consider these parameters.

The forward frame in the physical layer carries the data on a forward subband from the RNC to MT. Each forward frame is 80ms long and consists of a number of FEC blocks. The number of FEC blocks varies according to the type of forward bearer used. Table 4-2 shows the supported types of forward bearer [88].

Identifier	Frame Duration	Symbol Rate	Modulation	FEC Blocks per Frame	Beam Type
F80T1X4B	80 ms	33.6 kS/s	16-QAM	4	Regional
F80T4.5X8B	80 ms	4.5 x 33.6 kS/s	16-QAM	8	Narrow
F80T1Q4B	80 ms	33.6 kS/s	QPSK	4	Regional

Table 4-2 Types of Forward Bearers

Each type of forward bearer supports a range of code rates and is bounded by the lowest and the highest code rates which are used to deliver error-free data from the RNC to the MT under different radio link conditions. The code rate is a fractional number which indicates the portion of the total amount of information that is useful. Figure 4-9 shows the difference in the adaptive and the non-adaptive CAC algorithm at the physical layer. For a forward frame of 80ms duration belonging to a given forward bearer type, if the total number of FEC bits in the frame are *C* bits, then after applying FEC coding, the number of the data bits become less than *C* since the remaining bits are used as parity bits which act as redundant bits and are essential to the error detection and correction procedure.

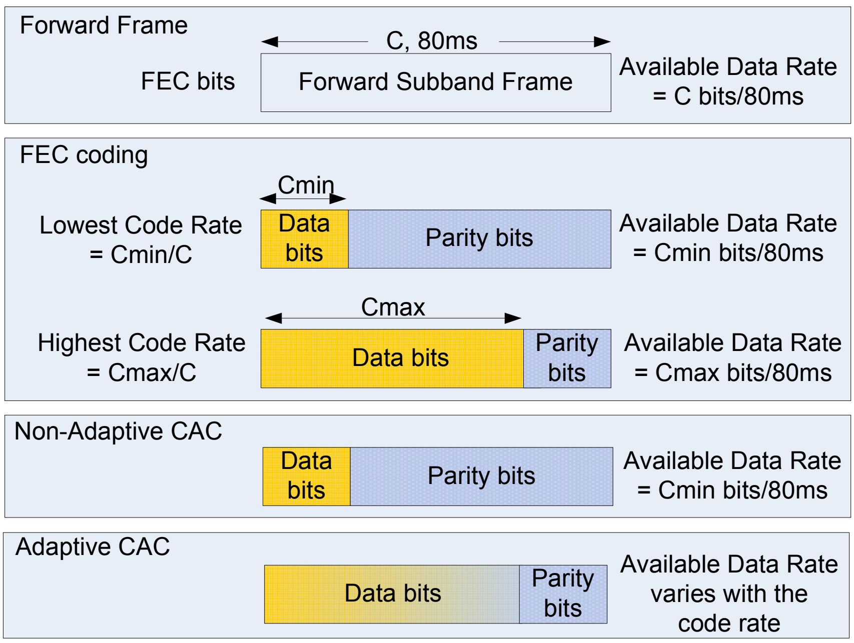

Figure 4-9 Difference in adaptive and non-adaptive CAC algorithm at the physical layer

Hence, for a given forward bearer type, if the lowest code rate (LowestCR) and the highest code rate (HighestCR) supported is given by C_{min}/C and C_{max}/C respectively, then the *UsefulDataBits* that can be delivered over a radio link are,

$$UsefulDataBits_{LowestCR} = TotalCapacity * LowestCR = C_{min} bits \quad (4.5)$$

$$UsefulDataBits_{HighestCR} = TotalCapacity * HighestCR = C_{max} bits \quad (4.6)$$

The change in the link condition causes a change in the FEC code rates. Another factor which affects the transmission of the data is the class of the MT. For the given system, three types of MTs differing in their size of the antenna with each supporting a range of code rates are considered termed as MT Class; Class1, Class2 and Class3. An MT belongs to one of the three classes and the code rates supported by each MT class vary for a given type of forward bearer. The non-adaptive CAC algorithm considers the lowest code rate such that it is supported by all the MT classes for a given type of forward bearer. However, such consideration may lead to an over-estimation of the resources used. On the other hand, the adaptive CAC algorithm takes into consideration the class and the changing link condition of the MT to calculate the resources used by the system. This helps in correctly estimating the available resources and avoids the over or under estimation of the resources used. Hence, the data rate for the non-adaptive CAC algorithm remains fixed at C_{min} *bits/80ms* whereas the data rate for the adaptive CAC algorithm varies between C_{min} *bits/80ms* and C_{max} *bits/80ms* as shown in Figure 4-9.

In order to calculate the total resources used on a given forward subband, a list of all the active connections on the forward subband is maintained and the total resources used is calculated as follows:

$$ResourcesUsed = \sum_{AllConnections} \frac{EstCap}{coderate} \qquad (4.7)$$

where, *EstCap* is the output from the Effective Bandwidth Estimator and, *coderate* is a fixed value in non-adaptive CAC algorithm but varies with the changing link condition and the class of the MT for the adaptive CAC algorithm.

4.2.2.1.4 Admission Decision Controller

The Admission Decision Controller takes the inputs, *EstCap* and *ResourcesUsed* for each connection and produces the output, *CAC_Output*, which performs one of the many possible actions.

Table 4-3 and Table 4-4 summarize the type of actions indicated by *CAC_Output* for unicast and multicast traffic respectively. The *CAC_Output* varies for the unicast and the multicast traffic since the admission decision process differs for both types of traffic which is described in the next section. As can be seen, the actions can be broadly classified as: Admit, Trigger, Block and Drop.

Actions	***CAC_Output*** **– Unicast**
1)	Admit : • New connection on the given *Fwdsubband.* • New connection on another *Fwdsubband* . • New connection on given *Fwdsubband* by retuning some other MT. • New connection on given *Fwdsubband* by dropping one or more connections.
2)	Trigger: • Retune Controller to check the possibility of retuning new connection. • Retune Controller to check the possibility of retuning any other MT on the given *Fwdsubband.* • Pre-emption Controller to check the possibility of pre-empting one or more connections.
3)	Block the new connection
4)	Drop one or more connections

Table 4-3 *CAC_Output* for Unicast traffic

Actions	***CAC_Output*** **– Multicast**
1)	Admit : • Join the existing multicast session on the given *Fwdsubband.* • Join the existing multicast session on another *Fwdsubband.* • Create new multicast session on given *Fwdsubband.* • Create new multicast session on given *Fwdsubband* by retuning some other MT. • Create new multicast session on given *Fwdsubband* by dropping some connections.
2)	Trigger: • Retune Controller to check the possibility of retuning new multicast connection. • Retune Controller to check the possibility of retuning any other MT on the given *Fwdsubband*. • Pre-emption Controller to check the possibility of pre-empting one or more connections.
3)	Block the new multicast connection
4)	Drop one or more connections

Table 4-4 *CAC_Output* for Multicast traffic

4.2.2.2 Admission Decision Process for Unicast and Multicast connections

The flow of the admission decision process varies for the two modes of transmissions; unicast and multicast. Hence, there is a need for two separate admission control procedures. Figure 4-10 shows the overall flow of the admission decision process.

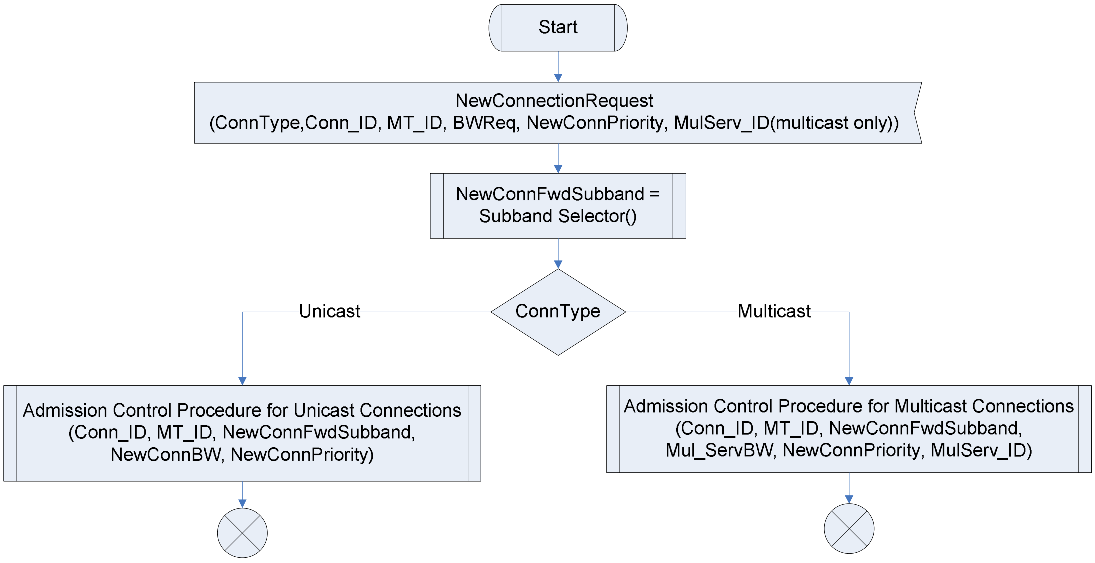

Figure 4-10 Overall Admission Flow

The admission process is triggered by an incoming connection request, *NewConnectionRequest*, containing the following parameters:

1) *ConnType* – Indicating unicast or multicast Connection
2) *Conn_ID* – A unique ID assigned to each connection request
3) *MT_ID* – A unique ID assigned to each MT. Each MT can have more than one connection.
4) *BWReq* – Indicates the amount of resources required by the connection.
5) *NewConnPriority* – Priority assigned to the new connection according to the type of connection shown in Table 4-1.

6) *MulServ_ID* – Indicates the multicast service the user wants to receive. Each multicast service supported by the system is assigned a unique ID.

The following sequence of steps illustrates the working of the overall admission decision process:

1. On receiving the new connection request, the SubbandSelector() procedure is initiated to select the forward subband for the new connection request, *NewConnFwdSubband*. The SubbandSelector() checks the parameter, *MT_ID*, in the connection request to see if the MT is already present in the system.

2. If the MT is already present in the system, the *NewConnFwdSubband* is the forward subband already selected for the existing MT.

3. However, if the new connection request belongs to an MT that is not present in the system, a new forward subband needs to be assigned to this new MT. Depending on the type of the method chosen, either *MinConnSubSel* or *Random*, the *NewConnFwdSubband* is assigned a forward subband as explained in Section 4.2.2.1.1.

4. Once the *NewConnFwdSubband* has been selected, the appropriate admission control procedure is triggered depending on whether the type of connection request, *ConnType*, is unicast or multicast.

4.2.2.2.1 Admission Process for Unicast Connections

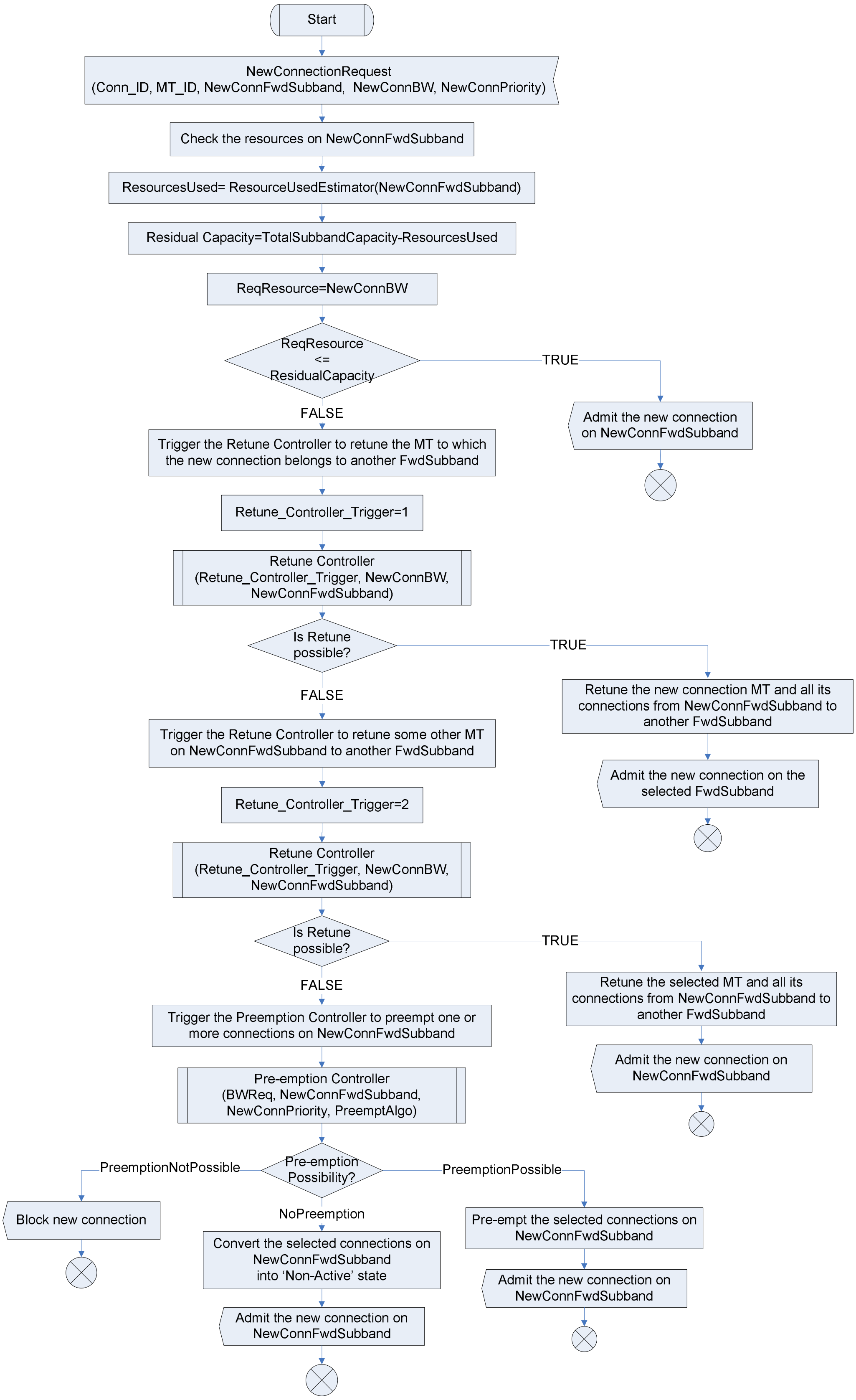

Figure 4-11 Admission Decision Process for unicast connections

Figure 4-11 shows the admission decision process for the unicast connections. The following sequence of steps illustrates the working of the admission control procedure for the unicast connections:

1. On receiving the new connection request, it first checks if there are enough resources on *NewConnFwdSubband* by checking the following condition:

$$ReqResource \leq TotalCapacity_{NewConnFwdSubband} - ResourcesUsed \quad (4.8)$$

   where,

$$ReqResource = NewConnBW$$
$$ResourcesUsed = ResourcesUsedEstimator\left(NewConnFwdSubband\right)$$

   The *ResourcesUsedEstimator* calculates the resources used on the *NewConnFwdSubband* based on the type of admission control algorithm selected, adaptive or non-adaptive as explained in Section 4.2.2.1.3.

2. In case the condition is true in Equation(4.8), the new connection is admitted on the *NewConnFwdSubband*. Otherwise, the CAC Processor triggers the Retune Controller to check the possibility of retuning the new connection to another available forward subband. Since, the MT to which the new connection belongs may have more connections running on *NewConnFwdSubband*, all the connections of the MT must be checked for retuning.

3. In case of possible retune, the new connection along with all other connections of the MT is retuned to the selected forward subband.

4. If the possibility of retuning the new connection fails, the CAC processor again triggers the Retune Controller to check for the possibility of retuning

some other MT on *NewConnFwdSubband* in order to accommodate the new connection.

5. If the possibility of retuning another MT exists, the selected MT on *NewConnFwdSubband* is retuned with all its connections to the selected forward subband, while the new connection is admitted to *NewConnFwdSubband*.
6. However, if the retuning of another MT fails, the CAC Processor triggers the Pre-emption Controller to check for the possibility of pre-empting one or more connections on *NewConnFwdSubband* such that new connection request can be accommodated.
7. In case the pre-emption is possible, the selected connections on *NewConnFwdSubband* are pre-empted and the new connection is admitted.
8. However, if the pre-emption with Connection Inactive Mechanism (CIM) is selected (explained in detail in Section 4.2.4), the selected connections are not pre-empted. Instead the connections are converted into '*Non-Active*' state so that the resources used by the selected connections are released and thus, the new connection is admitted.
9. In case the pre-emption is not possible, the new connection is blocked.

4.2.2.2.2 Admission Process for Multicast Connection

Figure 4-12 shows the admission decision process for multicast connections. A new multicast connection request is triggered when a user tries to join a particular multicast service (for instance the 'BBC'). In the multicast mode, several users can listen to the same multicast service without increasing the load on the system in contrast to the unicast mode, where each user is assigned individual resources.

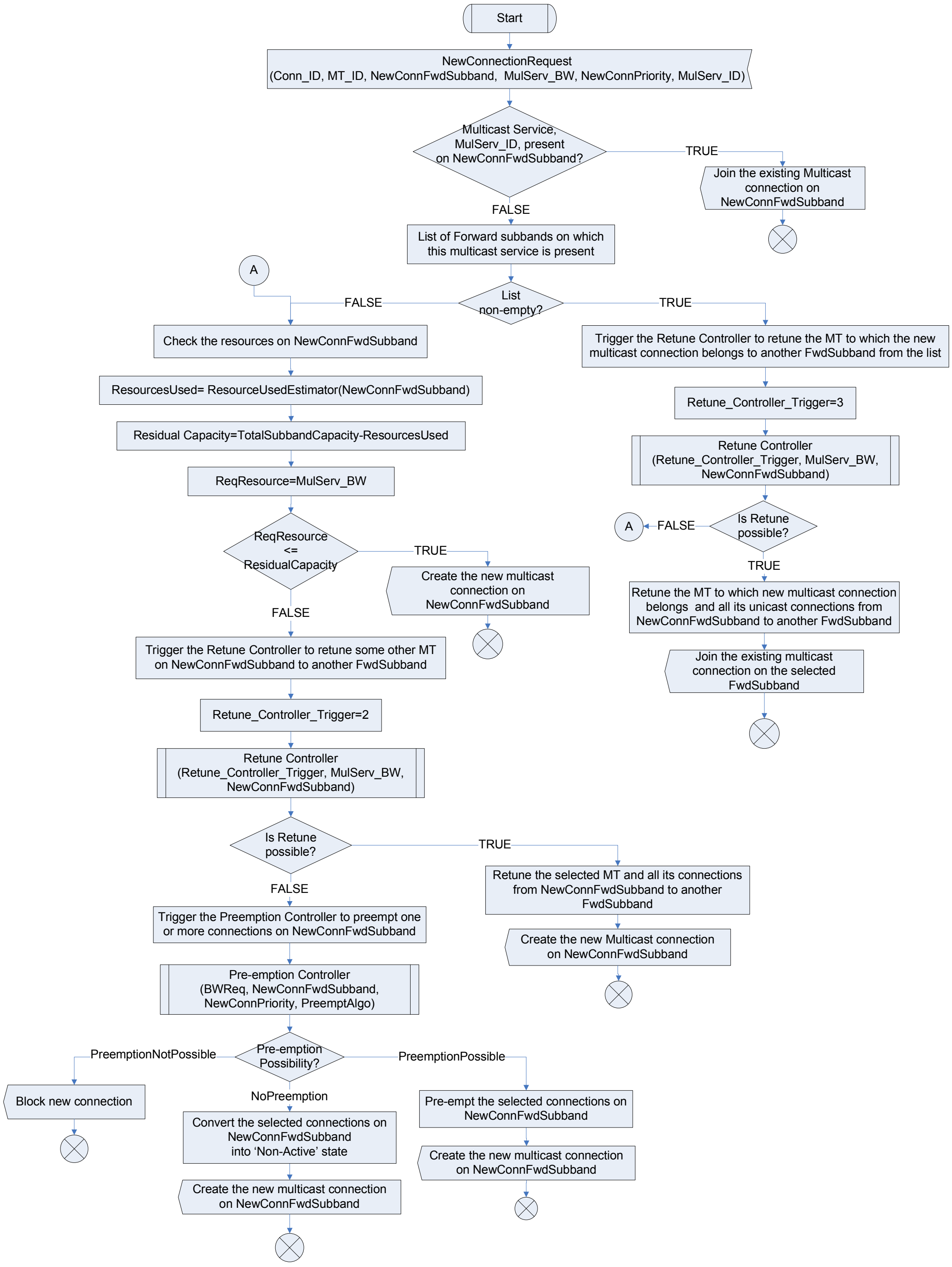

Figure 4-12 Admission Decision Process for Multicast Connections

The following steps detail how the admission process deals with a multicast connection request.

1. On receiving the new connection request to join a multicast service, it first checks whether the requested multicast service, *MulServ_ID*, is already running on *NewConnFwdSubband*.
2. In such a case, the user joins the existing multicast connection on *NewConnFwdSubband*. Since the multicast connection is already set up, by joining this multicast connection the user is not using additional resources.
3. However, if there is no existing multicast connection on *NewConnFwdSubband*, it checks whether the multicast connection for *MulServ_ID* exists on any other available forward subband. A list of such forward subbands is maintained.
4. If the list is empty which means there is no multicast connection running for *MulServ_ID*, the possibility of creating the new multicast connection on *NewConnFwdSubband* is checked. Go to Step 6.
5. However, if such multicast connection exists on other forward subband, *SelectedFwdSubband*, the CAC Processor triggers the Retune Controller to check the possibility of retuning the MT to which the new connection belongs. In case the retune is possible, the MT joins the existing multicast connection on the *SelectedFwdSubband* by retuning all its unicast connections from *NewConnFwdSubband* to the *SelectedFwdSubband*. However, if the possibility of retuning the MT fails, go to Step 6.
6. In order to create a new multicast connection, it is checked if there are enough resources on *NewConnFwdSubband* by checking the following condition:

$$ReqResource \leq TotalCapacity_{NewConnFwdSubband} - ResourcesUsed \quad (4.9)$$

where,

$$ReqResource = MulServ\_BW$$
$$ResourcesUsed = ResourcesUsedEstimator\left(NewConnFwdSubband\right)$$

The *ResourcesUsedEstimator* calculates the resources used on the *NewConnFwdSubband* based on the type of the admission control algorithm selected; adaptive or non-adaptive as explained in Section 4.2.2.1.3.

7. In case the condition is true in Equation (4.9), the new multicast connection is created on the *NewConnFwdSubband*.

8. Otherwise, the CAC Processor triggers the Retune Controller to check the possibility of retuning some other MT on *NewConnFwdSubband* in order to create the new multicast connection.

9. If the possibility of retuning another MT exists, the selected MT on *NewConnFwdSubband* is retuned with all its unicast connections to the *SelectedFwdSubband*, while the new multicast connection is created on *NewConnFwdSubband*.

10. However, if the retuning of another MT fails, the CAC processor triggers the Pre-emption Controller to check for the possibility of pre-empting one or more unicast connections on *NewConnFwdSubband* such that the new multicast connection can be accommodated.

11. In case pre-emption is possible, the selected unicast connections on *NewConnFwdSubband* are pre-empted and the new multicast connection is created.

12. However, if the pre-emption with Connection Inactive Mechanism (CIM) is selected (explained in detail in Section 4.2.4), the selected connections are not pre-empted instead the connections are converted into '*Non-Active*' state so that the resources used by the selected connections are released and thus, the new multicast connection is created.

13. In case pre-emption is not possible, the new multicast connection is blocked.

4.2.3 Retune Controller

The Retune Controller is the process which is triggered by the CAC Processor when it is required to check the possibility to retune an MT and its associated connections from one forward subband to another forward subband.

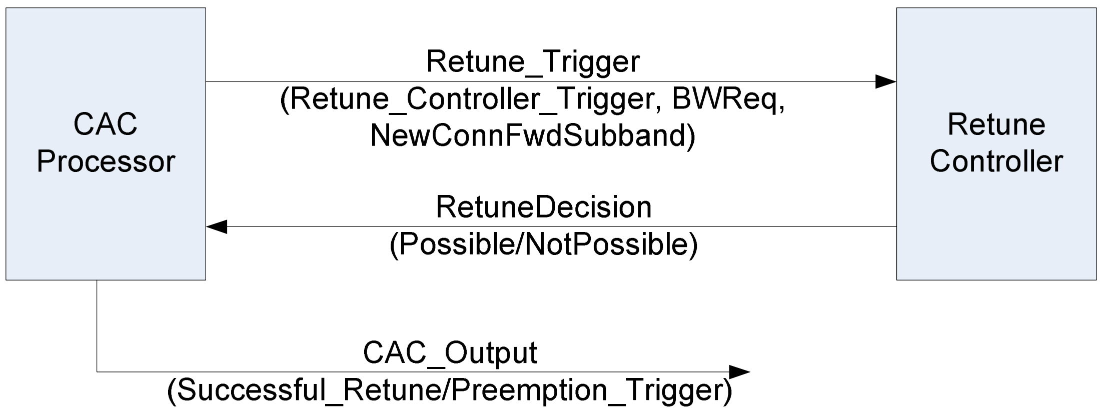

Figure 4-13 Flow of Interaction between CAC Processor and Retune Controller

Figure 4-13 shows the flow of interaction between the CAC Processor and the Retune Controller. The CAC processor triggers the Retune Controller with *Retune_Trigger* containing the following parameters:

1) *Retune_Controller_Trigger* – It is a variable which takes values between 1-3, each value indicating the type of retune required as shown in Table 4-5.

Value	***Retune_Controller_Trigger***
1	Possibility of retuning the new unicast connection request send by a given MT and all other existing connections of the given MT, from *NewConnFwdSubband* to another forward subband
2	Possibility of retuning other MT from *NewConnFwdSubband* to another forward subband
3	Possibility of retuning, the new multicast connection request send by a given MT and all other existing connections of the given MT, from *NewConnFwdSubband* to another forward subband

Table 4-5 Different values assumed by *Retune_Controller_Trigger* variable

2) *BWReq* – It indicates the bandwidth requirement of the new connection request.

3) *NewConnFwdSubband* – It indicates the forward subband assigned to the new connection request.

On Receiving the *Retune_Trigger*, the Retune Controller check for the possibility of retune and sends the decision back to the CAC Processor as *RetunePossible/RetuneNotPossible*. In case the retune is possible, the CAC processor admits the new connection by successful retuning; else the Pre-emption Controller process is triggered by sending the *Preemption_Trigger* message to check for the possibility of pre-emption. Figure 4-14 shows the flow of Retune Controller process.

The following sequence of steps illustrates the working of the Retune Controller process:

1. On receiving *Retune_Trigger* from the CAC processor, the value of *Retune_Controller_Trigger* is checked. If the value is 2 indicating the possibility of retuning the other MT on *NewConnFwdSubband*, go to Step3.

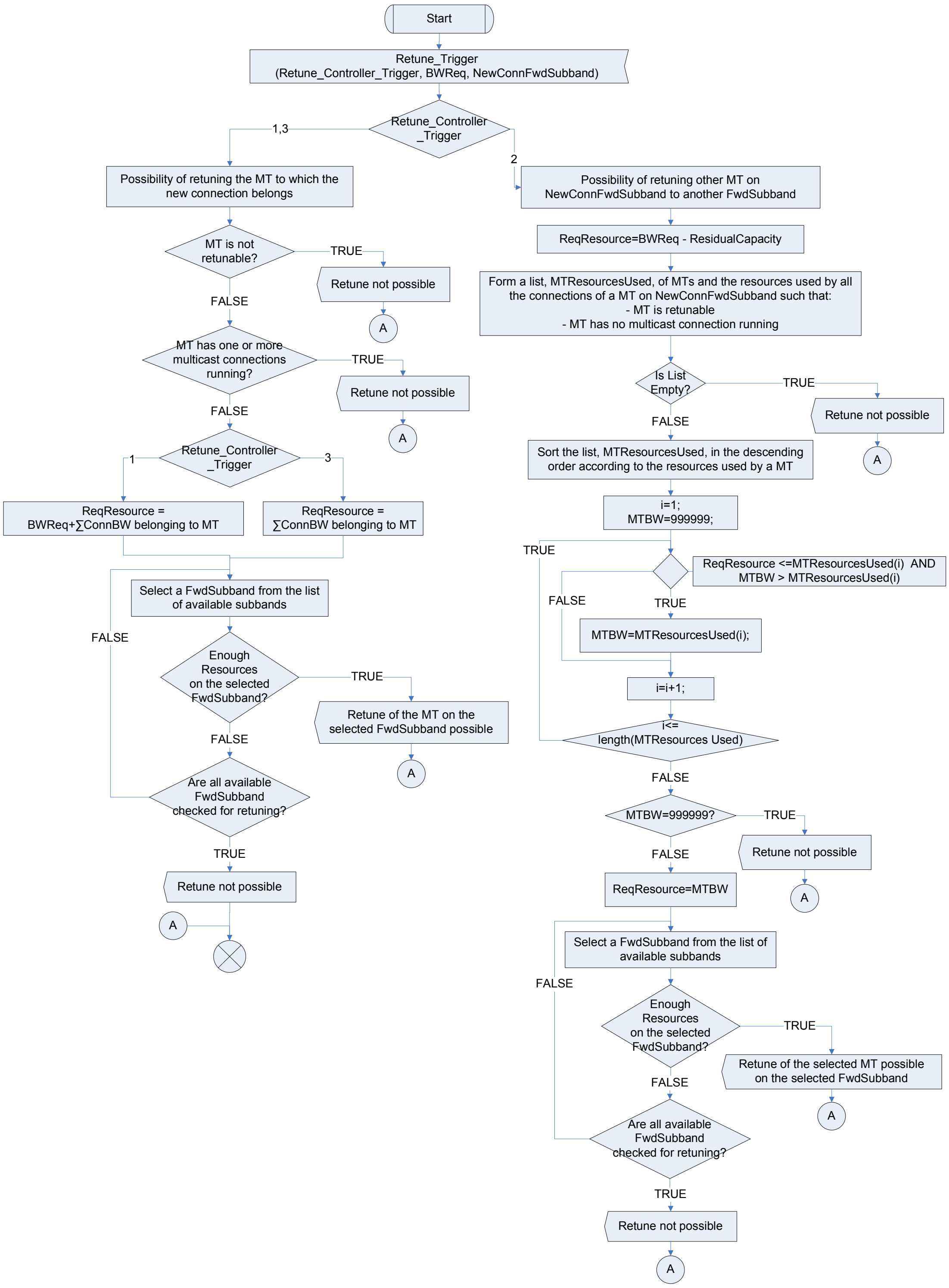

Figure 4-14 Flow of Retune Controller Process

2. However, if the value of *Retune_Controller_Trigger* is either 1 or 3 indicating the possibility of retuning the new unicast or multicast connection request sent by a given MT respectively. It is first required to check if the given MT can be retuned. The following conditions must be true for the MT to be able to retune:

   - The MT is configured to be *Retunable*;
   - The MT has no existing multicast connection running. It is assumed that the MT cannot retune while running a multicast connection.

   a. In case either of the above conditions is false, MT cannot be retuned. However, if both conditions are true, a list of available forward subbands is maintained. For the new unicast connection request, all available forward subbands are checked for the possibility of retune. However, for the multicast connection request only those forward subbands running the requested multicast service are checked.

   b. Once the forward subband, *SelectedFwdSubband*, is chosen, it is checked whether there are enough resources on the *SelectedFwdSubband* by checking the following condition:

$$ReqResource \le TotalCapacity_{SelectedFwdSubband} - ResourcesUsed \quad (4.10)$$

   where,

$$ReqResource = \begin{cases} BWReq + \sum_{ALL} ConnBW_{MT} & for\ unicast \\ \sum_{ALl} ConnBW_{MT} & for\ multicast \end{cases}$$

$$ResourcesUsed = ResourcesUtilizationEstimator\left(SelectedFwdSubband\right)$$

c. In the case there is sufficient resource, the retune of the MT on the *SelectedFwdSubband* is possible. Otherwise, Step 2b is repeated for all other available forward subbands. If all the available forward subbands have been checked for the possible admission of the new connection without any success, the retune is not possible.

3. This step is used if the value of *Retune_Controller_Trigger* is 2. In order to select an MT to retune from *NewConnFwdSubband* to another forward subband, a list of MTs on the *NewConnFwdSubband* is maintained such that:

    - The MT is configured to be '*retunable*'
    - The MT has no multicast connection running

    a. If such a list is empty, no retune is possible. Otherwise, the list is sorted in descending order according to the resource used by the MTs. It is checked whether the bandwidth utilized by the MT with the highest resource usage is greater than the bandwidth requirement, *BWReq*, of the new connection. If the condition is false, the retune is not possible.

    b. However, if the condition is true, the sorted list is traversed such that the MT with just enough resources to accommodate the new connection is selected.

    c. Once the MT is selected, a new forward subband, *SelectedFwdsubband*, is selected using the SubbandSelector() procedure. The admission control algorithm checks if there is enough resources on the *SelectedFwdSubband* to admit all the connections of the selected MT by checking the following condition:

$$ReqResource \le TotalCapacity_{SelectedFwdSubband} - ResourcesUsed \quad (4.11)$$

where,

$$ReqResource = \sum_{ALL} ConnBW_{SelectedMT}$$
$$BWReq \le ReqResource$$

d. In case there is sufficient resource, the retune of the selected MT and all its associated connections on the *SelectedFwdSubband* is possible. Otherwise, step 3c is repeated for all other available forward subbands. If all the available forward subbands have been checked, the possibility of retuning the selected MT does not exists.

4.2.4 Pre-emption Controller

The functionality of the Pre-emption Controller is to find the connections which can be pre-empted according to certain pre-emption criteria, such that the new connection can be admitted on the given forward subband. The necessity of pre-emption arises when it is essential to admit a higher priority connection in a congested system where there are no resources left. Under such a situation, it is required to pre-empt one or more lower priority connections in order to admit the new higher priority connection.

Figure 4-15 shows the flow of interaction between the CAC Processor and the Pre-emption Controller.

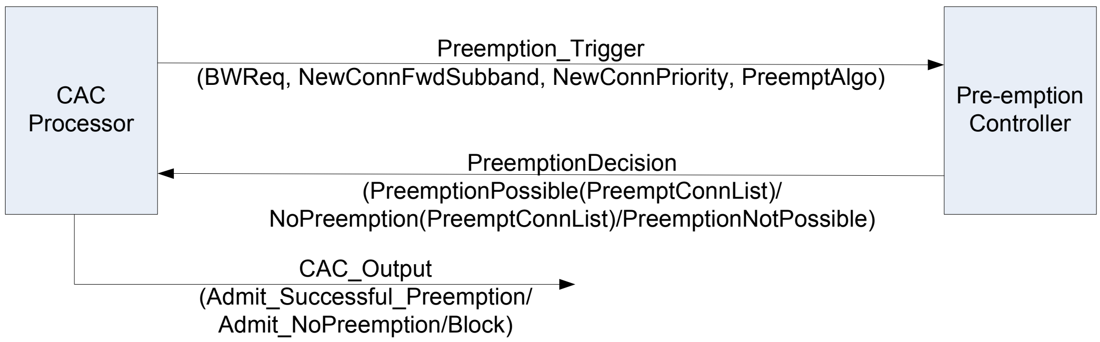

Figure 4-15 Flow of Interaction between CAC Processor and Pre-emption Controller

The Pre-emption Controller is triggered by the CAC Processor which then runs the pre-emption algorithms to decide whether the pre-emption is possible or not. The following parameters are sent in the trigger:

- *BWReq* – The bandwidth requirement of the new connection either unicast or multicast.
- *NewConnFwdSubband* – The forward subband on which the new connection attempts to get admitted.
- *NewConnPriority* – The priority of the new connection in accordance with the priorities shown in Table 4-1.
- *PreemptAlgo* – It is an integer value indicating the type of the pre-emption algorithm applied. Three pre-emption algorithms have been proposed:
    - Greedy
    - SubSetSum
    - Fuzzy

  Each pre-emption algorithm can be treated with the Connection Inactive Mechanism (CIM). The CIM allows the connections to be converted into the *Non-Active* state for a certain period of time by releasing the resources as opposed to dropping the connections. A more detailed description on CIM is provided later in the section 4.2.4.1. The parameter, *PreemptAlgo,* can take 1-6 values as shown in Table 4-6.

Value	Pre-emption Algorithm
1	Greedy without CIM
2	Greedy with CIM
3	SubSetSum without CIM
4	SubSetSum with CIM
5	Fuzzy without CIM
6	Fuzzy with CIM

Table 4-6 Values taken by *PreemptAlgo* parameter in *Preemption_Trigger*

Depending on the type of the pre-emption algorithm selected, the Pre-emption Controller forms a list of the connections to be pre-empted, *PreemptConnList*. If the list is empty, indicating that the pre-emption is not possible, the Pre-emption Controller sends the '*PreemptionNotPossible*' output to the CAC Processor. Otherwise the list is sent to the CAC Processor either in '*PreemptionPossible*' output when no CIM is applied or '*NoPreemption*' output which is the case when CIM is applied. Table 4-7 summarizes the input message to the CAC Processor from the Pre-emption Controller and the corresponding output.

Input	Output
PreemptionPossible(PreemptConnList)	Admit the new connection by pre-empting the connections listed in *PreemptConnList*
NoPreemption(PreemptConnList)	Admit the new connection by converting the connections into '*Non-Active*' state listed in *PreemptConnList*
PreemptionNotPossible	Block the new connection

Table 4-7 Input and Output messages of the CAC Processor

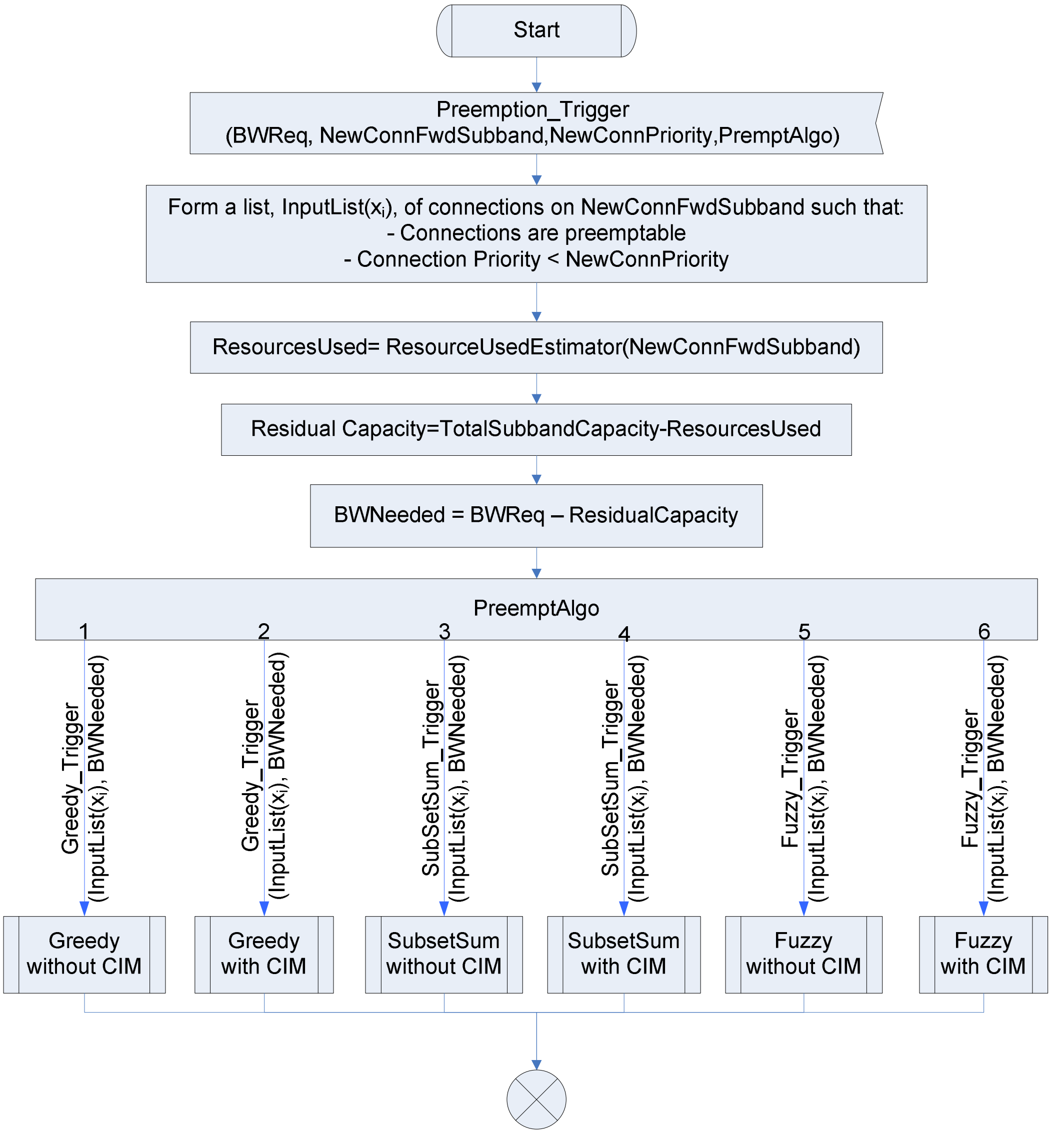

Figure 4-16 Overall flow of pre-emption process

Figure 4-16 shows the overall flow of the pre-emption process. The following sequence of steps illustrates the working of the overall pre-emption process:

1. On receiving the *Preemption_Trigger* from the CAC Processor, a list of connections on *NewConnFwdSubband* is maintained, *InputList(x_i)* such that for each connection:

    - Connection is configured '*Preemptable*';
    - *ConnectionPriority<NewConnPriority*, where the *ConnectionPriority* is the priority of the selected connection.

2. The amount of bandwidth required by the pre-emption of the connections is then calculated as follows:

$$BWNeeded = BWReq - ResidualCapacity \qquad (4.12)$$

where,

$$ResidualCapacity = TotalCapacity_{NewConnFwdSubband} - ResourcesUsed$$
$$ResourcesUsed = ResourcesUsedEstimator\left(NewConnFwdSubband\right)$$

3. Depending on the value of the parameter, *PreemptAlgo,* the appropriate pre-emption algorithm is triggered.

4.2.4.1 Connection Inactive Mechanism (CIM)

The CIM procedure is proposed to be used in conjunction with the pre-emption algorithm. This mechanism allows the connections which have been selected for pre-emption to be converted into *Non-Active* state for a certain period of time as opposed to dropping the connections in the first place. Since the selected connections under *Non-Active* state releases the resources used hence, the equivalent resource is used to admit the higher priority connection.

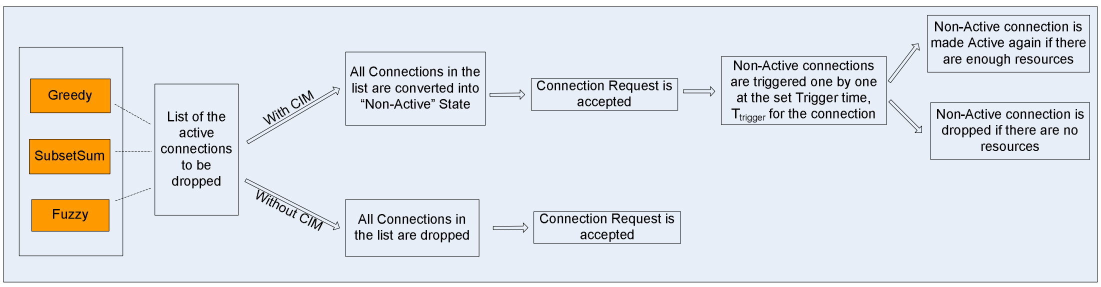

Figure 4-17 Pre-emption process with and without CIM

Figure 4-17 illustrates the pre-emption process with and without the CIM procedure. As it can be seen, the pre-emption algorithm produces a list of active connections to be dropped. In case where CIM is not enabled, all the connections are dropped and the higher priority connection is admitted into the system. However, if CIM is enabled, all the connections in the list are converted into *Non-Active* state and the higher priority connection is admitted. Each *Non-Active* connection is subjected to a certain amount of delay before the connection can request for the resources again. According to [90], there is no restriction on the transfer delay for interactive and background traffic class. Hence, certain time delays have been assumed for these types of traffic classes for CIM. Table 4-8 shows the minimum and the maximum transfer delay values assumed for the different traffic type.

Traffic	Traffic Class	THP	T_{min} (sec)	T_{max} (sec)
Netted Voice	Interactive	1	30	60
Web Browsing	Interactive	2	60	90
Email	Background		90	120

Table 4-8 Minimum and Maximum transfer delay assumed for the Interactive and Background class traffic

The time at which the connection request for the *Non-Active* connection is triggered again is given by,

$$t + T_{min} \le T_{Trigger} \le t + T_{max} \quad (4.13)$$

where, t is the time at which the connection is made *Non-Active*.

Once the $T_{Trigger}$ is reached in the simulation environment, the request for the admission is sent to the CAC Processor to request for the resources. It is treated as a new connection request only. If there are enough resources, the *Non-Active* connection is changed into active connection otherwise it is pre-empted.

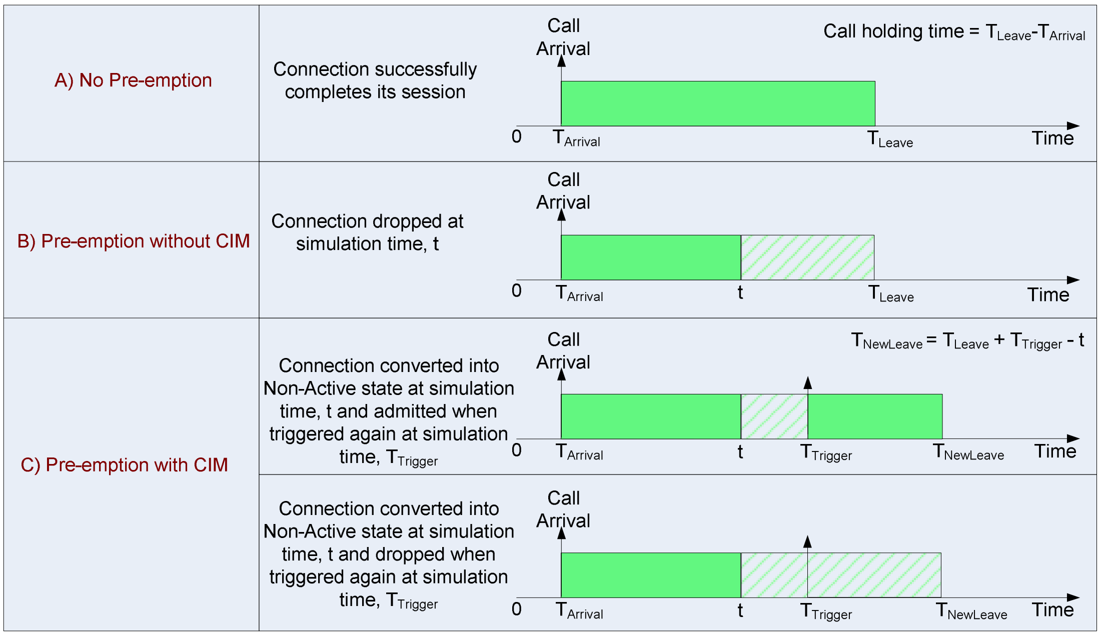

Figure 4-18 CIM procedure explained on a time scale

Figure 4-18 shows the CIM procedure on a time scale in a simulation environment. The connection once accepted into the system is expected to finish its session time indicated by the call holding time of the connection given by, T_{Leave}-$T_{Arrival}$. Hence, the connection arrives at time, $T_{Arrival}$, and leaves the system at T_{Leave}. Such is the case when no pre-emption exists. However, in case of pre-emption when no CIM is applied, the connection is pre-empted at time, *t*, in between its session. In case of pre-emption with CIM, the connection is converted into *Non-Active* state and will no longer use its resources. The connection request is made again at time, $T_{Trigger}$. If there are enough resources at the time, the connection is accepted into the system which then finishes its remaining session time given by,

$$T_{NewLeave} = T_{Leave} + T_{Trigger} - t \tag{4.14}$$

Otherwise, the connection request is dropped.

4.2.4.2 Greedy pre-emption algorithm

The Greedy pre-emption algorithm performs by pre-empting the connections with the lowest resource usage. The purpose of the algorithm is to reduce the bandwidth which is released by pre-empting the connections. However, in doing so, it greedily pre-empts more connections.

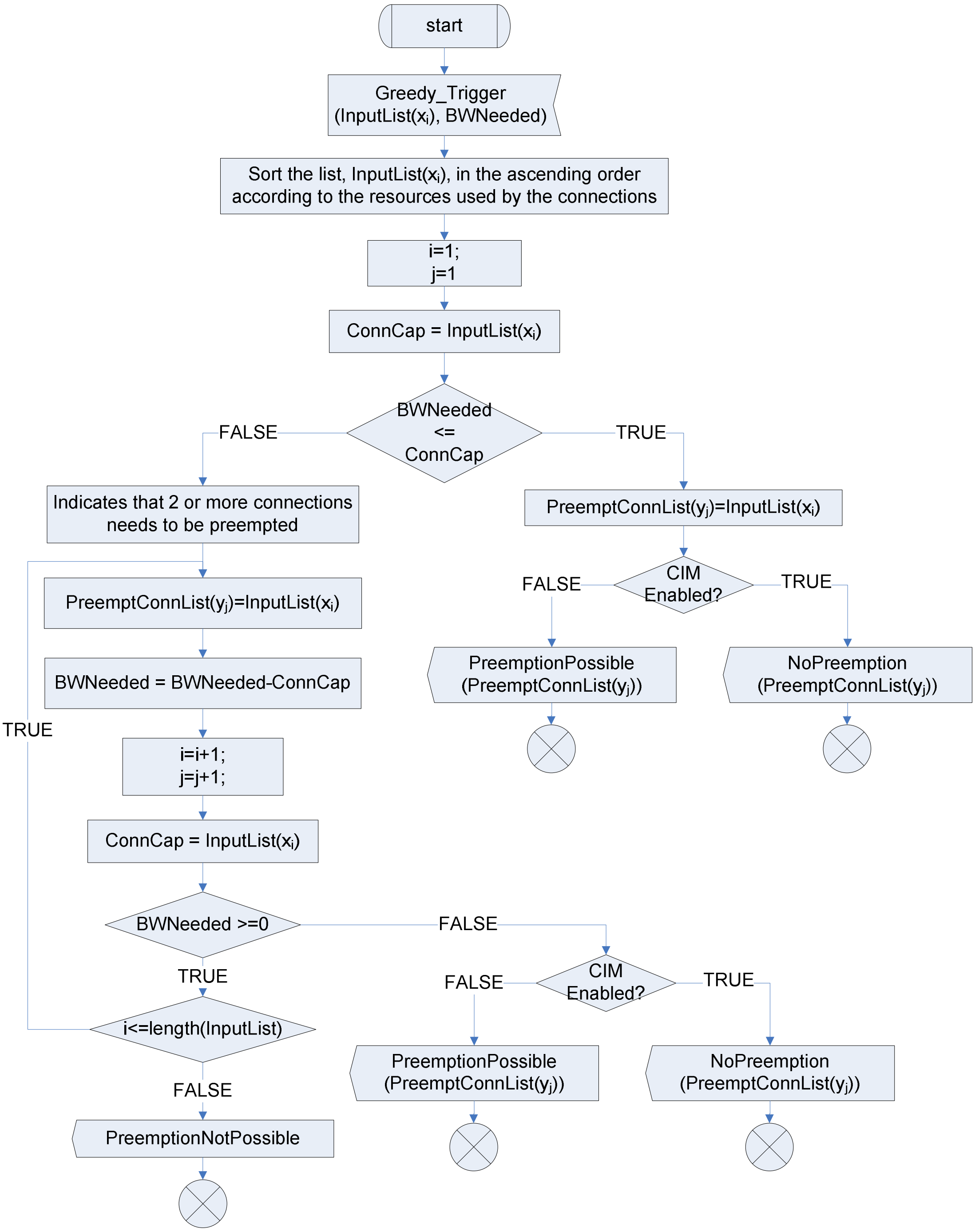

Figure 4-19 Flow Chart of Greedy pre-emption algorithm

Figure 4-19 shows the flowchart for the Greedy pre-emption algorithm. The following sequence of steps illustrates the working of the algorithm:

1. On receiving the *Greedy_Trigger* with *InputList*(x_i) and *BWNeeded* as input parameters, the list, *InputList*(x_i), is sorted in the ascending order of the resources used by the connections.

2. The following condition is then checked for the first connection in *InputList*(x_i)

$$BWNeeded \leq ConnCap \tag{4.15}$$

   where, *ConnCap* is the capacity of the first connection.

3. If the condition in Equation (4.15) is true, the *PreemptConnList* consists of only one connection and depending on whether CIM is enabled, one of the following signals are sent back to the CAC Processor:

    a. *PreemptionPossible(PreemptConnList)* if no CIM is enabled

    b. *NoPreemption(PreemptConnList)* if CIM is enabled

4. However, if the condition is false in Equation (4.15), more than one connection is required to be dropped in order to admit the new connection. Hence, the algorithm iterates in a loop forming the list, *PreemptConnList*, by selecting the connections from the sorted list until one of the following condition becomes true:

    a. $BWNeeded \leq \sum_{AllSelectedConnections} ConnCap$

    b. Sorted list is fully traversed

5. If condition 4.a is satisfied, depending on whether CIM is enabled, one of the following signal is sent back to the CAC Processor:

a. *PreemptionPossible(PreemptConnList)* if no CIM is enabled

b. *NoPreemption(PreemptConnList)* if CIM is enabled

6. However, if condition 4.b is satisfied indicating that the pre-emption is not possible, then, the *PreemptionNotPossible* signal is sent back to the CAC Processor.

4.2.4.3 SubSetSum pre-emption algorithm

The SubSetSum pre-emption algorithm performs by selecting the pre-emptable connections so as to minimize the bandwidth which is released by pre-empting the connections in an optimum manner. The algorithm is based on a SubSetSum problem which states that given a set *A* of positive integers such that $A = [a_1, a_2, \ldots\ldots, a_n]$ and a positive integer called the target sum, *s*, where, $s \le \sum_{i=1}^{n} a_i$ there exists a column vector $X = [x_1, x_2, \ldots\ldots, x_n]^T$, $x_i \in [0,1]$ such that *AX* is as large as possible but not greater than *s*.

Adapting the SubSetSum problem to the pre-emption problem is equivalent to finding a subset of connections belonging to a set of pre-emptable connections, $A = [a_1, a_2, \ldots\ldots, a_n]$ where *n* is the total number of pre-emptable connections on a forward subband and a_j is the resource consumed by connection *j*, such that the sum of this subset of *A*, i.e. *AX* where *X* is defined as before, can accommodate the resource, *r*, where, $r \le \sum_{i=1}^{n} a_i$, needed by the connection. In this case, the target sum is no longer a fixed value but a value with a minimum bound of *r* and a maximum bound of $s \le \sum_{i=1}^{n} a_i$. Since there can be more than one solution for *X*, the one which gives the minimum value of AX greater than r will be adopted. Any element x_i equalling to 1 in the chosen solution of X will lead to connection i being dropped. Hence, the algorithm drops one or more, lower priority connections such that the total resource consumption of these connections is just enough to accommodate the new connection request. This enables high priority connection requests to be admitted without dropping more than necessary existing lower priority connections, thus minimizing the bandwidth released by the pre-emptable connections and also reducing the number of connections pre-empted. Hence, SubSetSum pre-emption algorithm is an improvement over the Greedy pre-emption algorithm.

Figure 4-20 shows the flowchart for the SubSetSum pre-emption algorithm. The following sequence of steps illustrates the working of the algorithm:

1. On receiving the *SubsetSum_Trigger* with *InputList*(x_i) and *BWNeeded* as input parameters, the list, *InputList*(x_i), is sorted in the descending order according to the resources used by the connections and then according to the priority of the connection.

2. A new list is defined, *OutputList* which is initialized as, $OutputList = \{y_1\}$ where $y_1=0$.

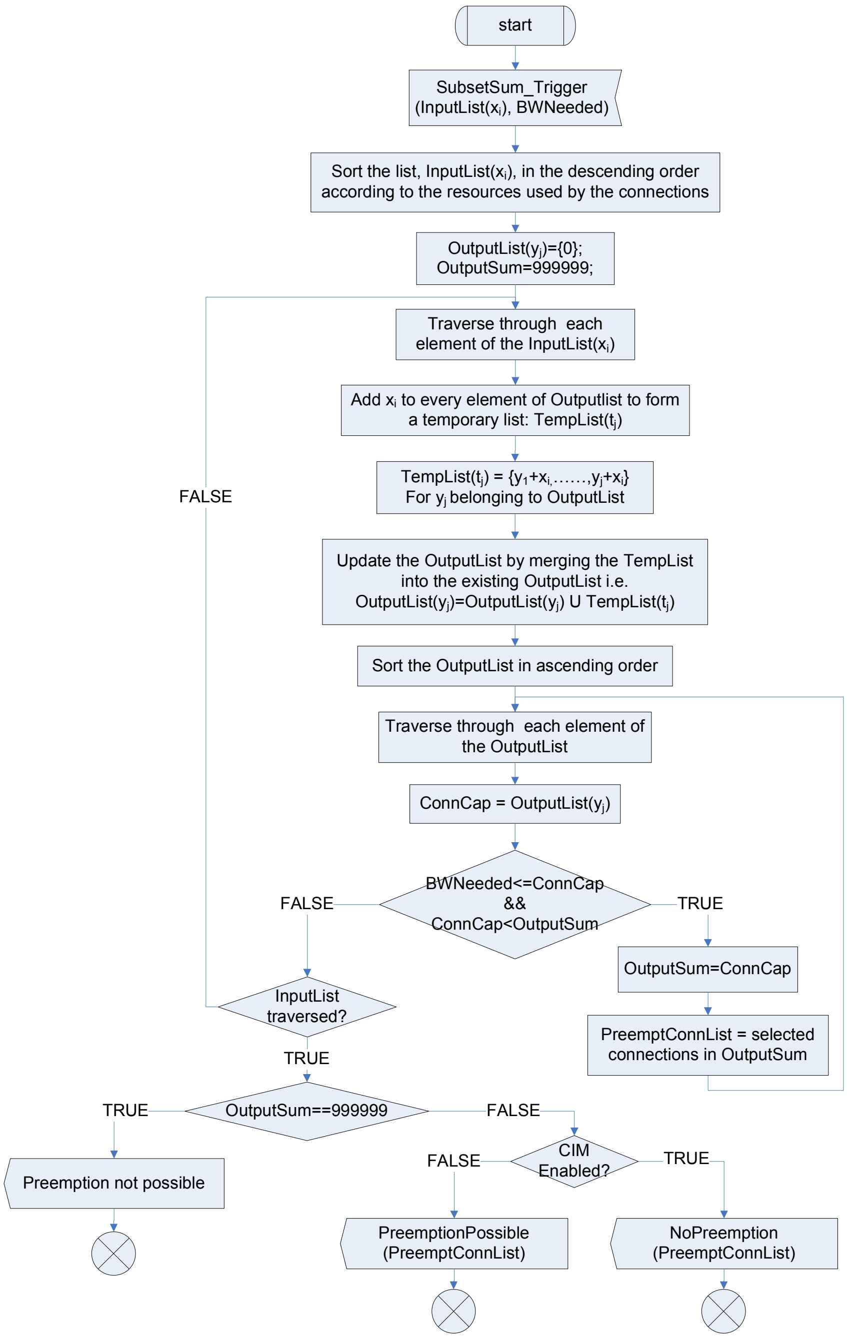

Figure 4-20 Flow Chart of SubSetSum pre-emption algorithm

3. For each $x_i \in InputList$

   a. Define a temporary list; *TempList* whose elements are formed by adding the element x_i to every element of *OutputList* i.e.

   $$TempList = \{y_1 + x_i \ldots\ldots\} \qquad \forall \qquad y_j \in OutputList$$

   b. $OutputList = OutputList \cup TempList$

c. Sort the *OutputList.*

d. Traverse each element of the *OutputList* and check the following condition: $BWNeeded \leq ConnCap$ where, *ConnCap* is the capacity represented by the element of the *OutputList*.

e. Once the condition is true, store the value in a variable *OutputSum* and store the corresponding connections in *PreemptConnList*.

4. Once all the elements of the *InputList* are traversed, the value of *OutputSum* is checked. If the value of *OutputSum* is the same as its initialization value, indicating that the pre-emption is not possible, then the *PreemptionNotPossible* signal is sent back to the CAC Processor.

5. Otherwise, depending on whether CIM is enabled, one of the following signal is sent to the CAC Processor:

   a. *PreemptionPossible(PreemptConnList)* if no CIM is enabled

   b. *NoPreemption(PreemptConnList)* if CIM is enabled

4.2.4.4 Fuzzy pre-emption algorithm

Fuzzy pre-emption algorithm is proposed to overcome the shortcomings of the SubSetSum algorithm and to further enhance the performance of the system. Although, SubSetSum algorithm provides an optimum solution by minimizing the bandwidth released by the pre-emption of the connections and by reducing the number of connections to be pre-empted, however, it is mathematically complex and requires more computation time. Also, it does not consider other factors such as priority of the connection while deciding which connections to be pre-empted. Fuzzy algorithm is proposed as it has many advantages over the SubSetSum algorithm such as the following:

- Fuzzy logic is conceptually easy to understand since it is based on natural language.
- It is more flexible since it does not require precise inputs to solve the problem. It allows the truth values or membership values to be indicated by a value in the range [0 1] instead of the Boolean values 0 and 1.
- Human expertise knowledge can be used while defining the fuzzy rules.
- The algorithm considers more than one criterion while making the decision to pre-empt the connection such that the most optimum and fairer solution is provided.

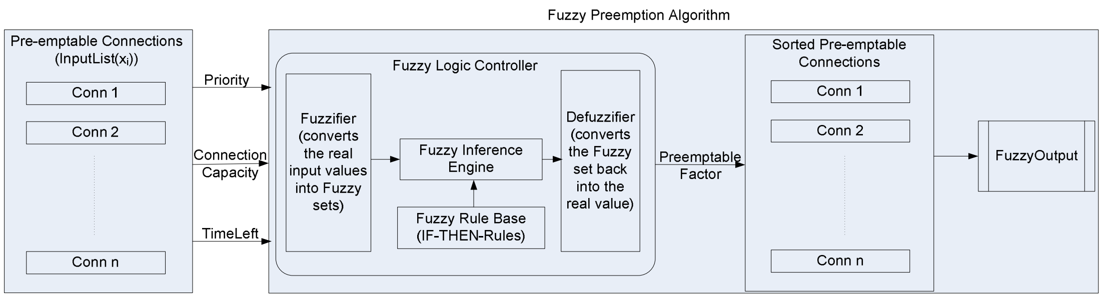

Figure 4-21 Block diagram of Fuzzy Pre-emption Algorithm

Figure 4-21 shows the block diagram describing the methodology and criteria used in the Fuzzy pre-emption algorithm. The main idea of the algorithm is to produce the output, *PreemptableFactor* for each pre-emptable connection using a given number of input criteria. *PreemptableFactor* indicates the odds of a connection to be pre-empted; the higher the value of *PreemptableFactor*, the greater the chance of the connection to be pre-empted and vice-versa. Once the *PreemptableFactor* is calculated for each pre-emptable connection, the list is sorted according to the value of *PreemptableFactor*. The sorted list is then sent to the *FuzzyOutput* procedure which selects the connections for pre-emption. The following three input criteria have been used to compute *PreemptableFactor*:

- *Priority* – indicates the priority of the connection which in turn depends on the type of traffic.
- *ConnectionCapacity* – indicates the resource utilized by the connection on the given forward subband.
- *TimeLeft* – indicates the remaining service time of the connection.

Table 4-9 shows the range used for the input and the output variables while designing the Fuzzy pre-emption algorithm. The range has been selected such that the design remains suitable for different traffic classes with varying QoS requirements.

Type of Variable	Variables	Range
Input	Priority	2,3,4 [Table 4-1]
Input	Connection Capacity	1 – 10 kbits
Input	TimeLeft	0 – 1200 seconds
Output	Pre-emptable Factor	0 – 1

Table 4-9 Range of Input and Output variables used in Fuzzy pre-emption algorithm

The core of the Fuzzy pre-emption algorithm is the Fuzzy Logic Controller (FLC) [96]. It collects the input variables for each pre-emptable connection and based on that information, it produces the *PreemptableFactor* as an output. The main functional entities of FLC are as follows:

- Fuzzifier – It maps the real values of the input criteria into fuzzy sets with a varying degree of membership using the membership functions defined for each input criteria.

- Fuzzy Rules – defines the appropriate fuzzy operation using IF-THEN rules.
- Inference – converts the fuzzy input into fuzzy output using IF-THEN fuzzy rules.
- Defuzzifier – converts the fuzzy output of the inference engine into real values using membership functions.

The operation of FLC can be divided into three stages according to its functional entities namely: Fuzzification, Inference and Defuzzification [97]. Each of the stages is described in detail in the following subsections.

4.2.4.4.1 Fuzzification

During fuzzification, the input values are fed into the fuzzifier which maps the real or crisp values to the corresponding linguistic values of the fuzzy sets. Fuzzy sets are the sets whose elements have varying degree of membership. For example, the capacity used by a connection in a crisp set can only be considered to be either ‘Low’ or ‘High’ and not both simultaneously. However, in a fuzzy set, connection capacity can be presented as ‘Low’, ‘Medium’, and ‘High’. The values of the fuzzy sets can be obtained by mapping the real value onto the membership function which is a curve or a line. The most common shapes used for membership functions are Gaussian, Trapezoidal and Triangular.

The given input and output linguistic variables are assumed to have either triangular or trapezoidal membership functions which are described as below:

- The triangular curve function, *trimf(x,[a,b,c])*, is a function of vector *x*, and depends on three scalable factors, *a*, *b*, and *c*. The parameters, *a* and *c* locate the "feet" of the triangle and the parameter *b* locates the peak.

- The trapezoidal curve function, *trapmf(x,[a,b,c,d])*, is a function of vector *x*, and depends on four scalable factors, *a*, *b*, *c* and *d*. The parameters, *a* and *d* locate the "feet" of the trapezoid and the parameter *b* and *c* locates the "shoulders".

The fuzzy variables assumed for the input linguistic variables, *Priority*, *TimeLeft*, and *ConnectionCapacity*; and for the output linguistic variable, *Pre-emptableFactor* are defined respectively as:

$$T(Priority) = \{High, Medium, Low\};$$
$$T(TimeLeft) = \{T1, T2, T3, T4\};$$
$$T(ConnectionCapacity) = \{Low, Medium, High\};$$
$$T(PreempatbleFactor) = \{PF1, PF2, PF3, PF4, PF5, PF6, PF7, PF8, PF9\}$$

Figure 4-22 shows the membership functions for the input variable, *Priority*, which can be presented as a set, $M(Priority) = \{\mu_{High}, \mu_{Medium}, \mu_{Low}\}$ where, μ_{High}, μ_{Medium}, μ_{Low} are the membership function for High, Medium and Low fuzzy variables respectively and are assumed to be triangular shaped.

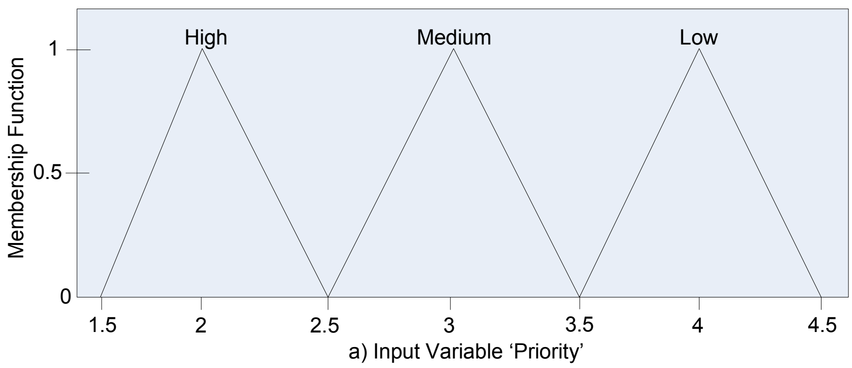

Figure 4-22 Membership function plot for Input variable, Priority

They are given by:

$$\mu_{High}(Priority) = trimf(Priority,[1.5\ \ 2\ \ 2.5]);$$
$$\mu_{Medium}(Priority) = trimf(Priority,[2.5\ \ 3\ \ 3.5]);$$
$$\mu_{Low}(Priority) = trimf(Priority,[3.5\ \ 4\ \ 4.5])$$

where, trimf is the triangular curve function.

Figure 4-23 shows the membership functions for the input variable, *TimeLeft*, which can be presented as a set, $M(TimeLeft) = \{\mu_{T1}, \mu_{T2}, \mu_{T3}, \mu_{T4}\}$. The membership functions μ_{T1}, μ_{T2}, μ_{T3}, μ_{T4} are assumed to be trapezoidal shaped.

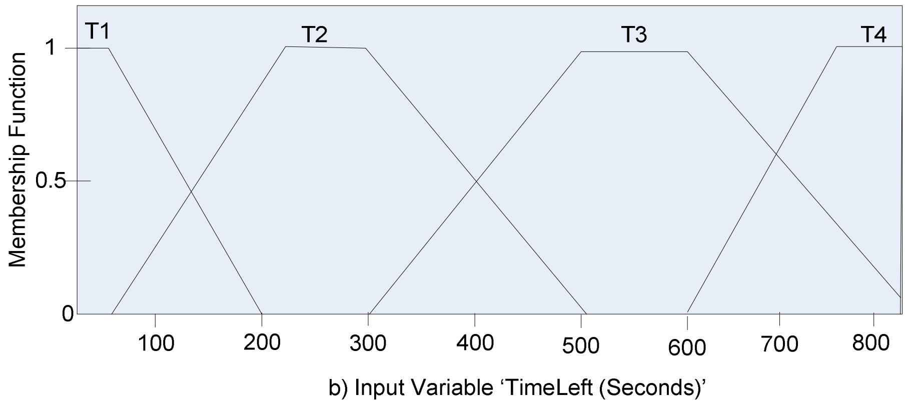

Figure 4-23 Membership function plot for Input variable, *TimeLeft*

They are given by:

$$\mu_{T1}(TimeLeft) = trapmf(Timeleft,[-200\ \ 0\ \ 20\ \ 200]);$$
$$\mu_{T2}(TimeLeft) = trapmf(TimeLeft,[20\ \ 200\ \ 300\ \ 500]);$$
$$\mu_{T3}(TimeLeft) = trapmf(TimeLeft,[300\ \ 500\ \ 600\ \ 800]);$$
$$\mu_{T4}(TimeLeft) = trapmf(TimeLeft,[600\ \ 750\ \ 850\ \ 1500])$$

where, trapmf is the trapezoidal curve function.

Figure 4-24 shows the membership functions for the input variable, *ConnectionCapacity*, which can be presented as a set,

$M(ConnectionCapacity) = \{\mu_{Low}, \mu_{Medium}, \mu_{High}\}$. The membership functions are again assumed to be trapezoidal shape.

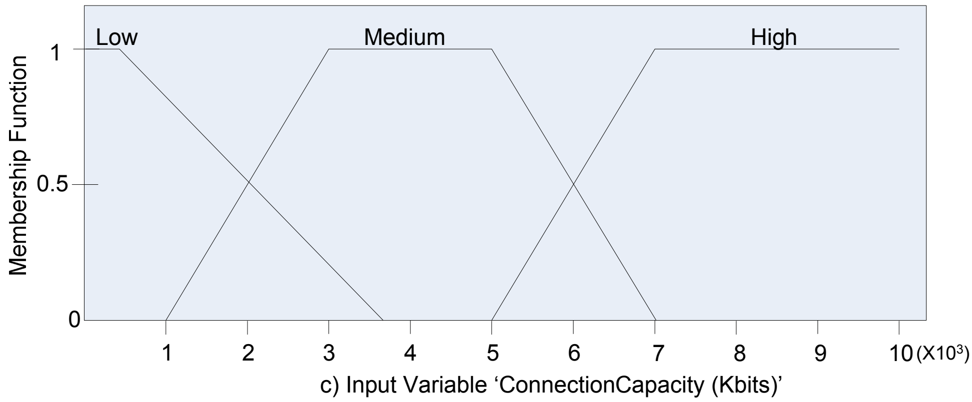

Figure 4-24 Membership function plot for Input variable, *ConnectionCapacity*

They are given by:

$$\mu_{Low}(ConnectionCapacity) = trapmf(ConnectionCapacity,[-3.6\ -0.4\ 0.4\ 3.6]);$$
$$\mu_{Medium}(ConnectionCapacity) = trapmf(ConnectionCapacity,[1\ \ 3\ \ 5\ \ 7]);$$
$$\mu_{High}(ConnectionCapacity) = trapmf(ConnectionCapacity,[5\ \ 7\ 10.4\ \ 14])$$

Figure 4-25 shows the membership functions for the output variable, *PreemptableFactor*, which can be presented as a set, $M(PreemptableFactor) = \{\mu_{PF1}, \mu_{PF2}, \mu_{PF3}, \mu_{PF4}, \mu_{PF5}, \mu_{PF6}, \mu_{PF7}, \mu_{PF8}, \mu_{PF9}\}$

The membership functions are assumed to be triangular shape.

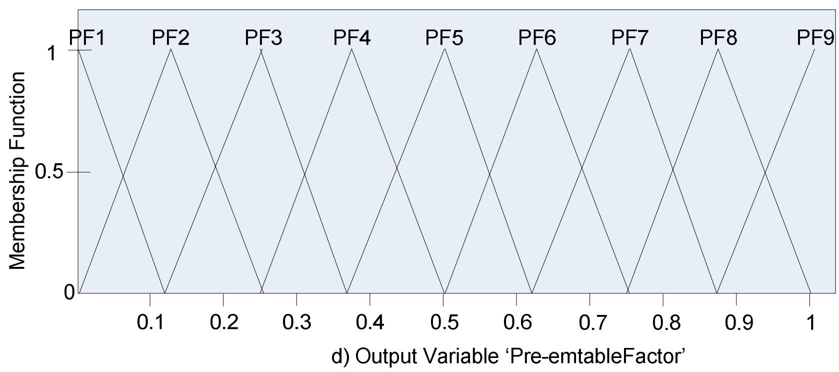

Figure 4-25 Membership function plot for Output variable, PreemptableFactor

They are given by:

$$\mu_{PF1}\left(PreemptableFactor\right) = trimf\left(PreemptableFactor,[-0.125\quad 0\quad 0.125]\right);$$
$$\mu_{PF2}\left(PreemptableFactor\right) = trimf\left(PreemptableFactor,[0\quad 0.125\quad 0.25]\right);$$
$$\mu_{PF3}\left(PreemptableFactor\right) = trimf\left(PreemptableFactor,[0.125\quad 0.25\quad 0.375]\right);$$
$$\mu_{PF4}\left(PreemptableFactor\right) = trimf\left(PreemptableFactor,[0.25\quad 0.375\quad 0.5]\right);$$
$$\mu_{PF5}\left(PreemptableFactor\right) = trimf\left(PreemptableFactor,[0.375\quad 0.5\quad 0.625]\right);$$
$$\mu_{PF6}\left(PreemptableFactor\right) = trimf\left(PreemptableFactor,[0.5\quad 0.625\quad 0.75]\right);$$
$$\mu_{PF7}\left(PreemptableFactor\right) = trimf\left(PreemptableFactor,[0.625\quad 0.75\quad 0.875]\right);$$
$$\mu_{PF8}\left(PreemptableFactor\right) = trimf\left(PreemptableFactor,[0.75\quad 0.875\quad 1]\right);$$
$$\mu_{PF9}\left(PreemptableFactor\right) = trimf\left(PreemptableFactor,[0.875\quad 1\quad 1.125]\right)$$

4.2.4.4.2 Inference

The second stage involves feeding the fuzzy sets into an inference engine by applying the fuzzy rules to obtain the fuzzy decisions. The fuzzy rule base is the control policy knowledge base, characterized by a set of linguistic statements in the form of IF-THEN rules that describe the fuzzy logic relationship between the input and output variables. The fuzzy rules can be implemented as a look up table.

Priority	***Time Left***	***Connection Capacity***	***Pre-emptable Factor***
Low	T1	Low	PF6
Low	T2	Low	PF7
Low	T3	Low	PF8
Low	T4	Low	PF9
Low	T1	Medium	PF5
Low	T2	Medium	PF6
Low	T3	Medium	PF7
Low	T4	Medium	PF8
Low	T1	High	PF4
Low	T2	High	PF5
Low	T3	High	PF6
Low	T4	High	PF7

Table 4-10 Fuzzy Rule base of Fuzzy pre-emption

Priority	***Time Left***	***Connection Capacity***	***Pre-emptable factor***
Medium	T1	Low	PF3
Medium	T2	Low	PF4
Medium	T3	Low	PF5
Medium	T4	Low	PF6
Medium	T1	Medium	PF3
Medium	T2	Medium	PF3
Medium	T3	Medium	PF4
Medium	T4	Medium	PF5
Medium	T1	High	PF2
Medium	T2	High	PF3
Medium	T3	High	PF4
Medium	T4	High	PF4
High	T1	Low	PF2
High	T2	Low	PF3
High	T3	Low	PF4
High	T4	Low	PF5
High	T1	Medium	PF1
High	T2	Medium	PF2
High	T3	Medium	PF3
High	T4	Medium	PF4
High	T1	High	PF1
High	T2	High	PF2
High	T3	High	PF3
High	T4	High	PF3

Table 4-11 Fuzzy Rule base of Fuzzy pre-emption (continued)

The fuzzy rule base for the fuzzy pre-emption consisting of 36 rules is shown in Table 4-10 and Table 4-11.

The most commonly used inference system is Mamdani's max-min fuzzy inference method [98]. Figure 4-26 illustrates the graphical analysis of the evaluation of the fuzzy rules and the aggregation of the outputs. The minimum membership value for the inputs propagates through to the output and truncates the membership function for the output of each rule [99]. This graphical inference is done for each rule. Then the truncated membership functions for each rule are aggregated using aggregation function max which results in an aggregated membership function comprised of the outer envelope of the individual truncated membership forms from each rule.

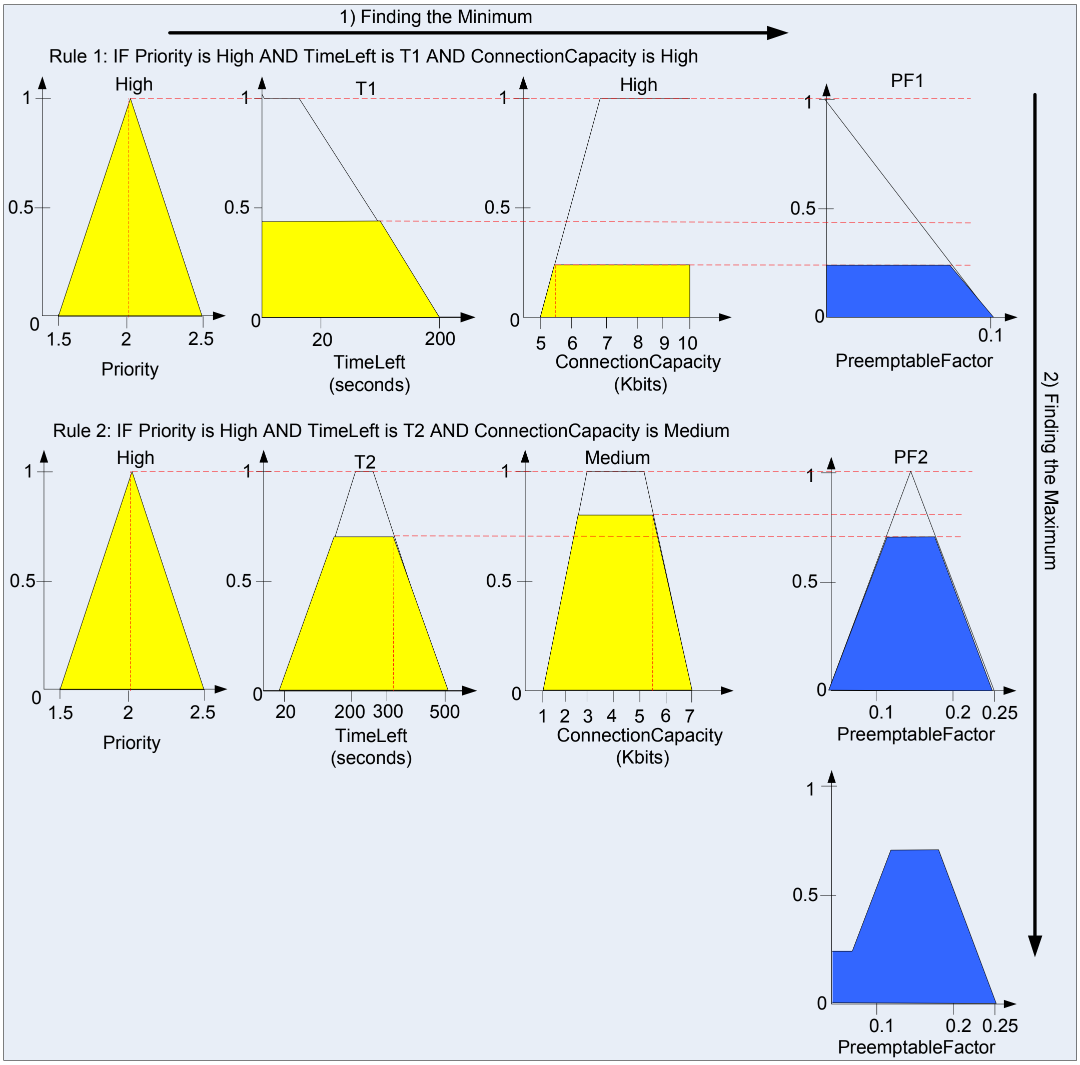

Figure 4-26 Fuzzy Inference Diagram for Fuzzy Preemption

4.2.4.4.3 Defuzzification

This is the last stage in finding the crisp value from the aggregated output from the Inference engine. Two most common techniques are the 'centroid' and 'maximum' methods. In the centroid method, the crisp value of the output variable is computed by finding the variable value of the center of gravity of the membership function for the fuzzy value. In the maximum method, one of the variable values at which the fuzzy subset has its maximum truth value is chosen as the crisp value for the output variable. The Fuzzy preemption uses the centroid defuzzification method to obtain the value of *PreemptableFactor*. The centroid defuzzification returns the center of area under the graph.

4.2.4.4.4 Fuzzy Output

Once the *PreemptableFactor* is calculated for each pre-emtable connection in *InputList*, the list is then sorted according to the *PreemptableFactor* in the descending order. The sorted list, *FuzzySortedInputList*, is send to the Fuzzy Output function to select the connections to be pre-empted.

Figure 4-27 shows the flow chart for fuzzy output. It selects the connection with the highest *PreemptableFactor*. The following sequence of steps illustrates the working of the Fuzzy output:

1. On receiving the *FuzzyOutput_Trigger* with *FuzzySortedInputList*(x_i) and *BWNeeded* as input parameters, the following condition is checked for the first connection in *FuzzySortedInputList*(x_i)

   $$BWNeeded \leq ConnCap \tag{4.16}$$

   where, *ConnCap* is the capacity of the first connection.

2. If the condition is true, the *PreemptConnList* consists of only one connection and depending on whether CIM is enabled, one of the following signals is sent back to the CAC Processor:

    a. *PreemptionPossible(PreemptConnList)* if no CIM is enabled

b. *NoPreemption(PreemptConnList)* if CIM is enabled

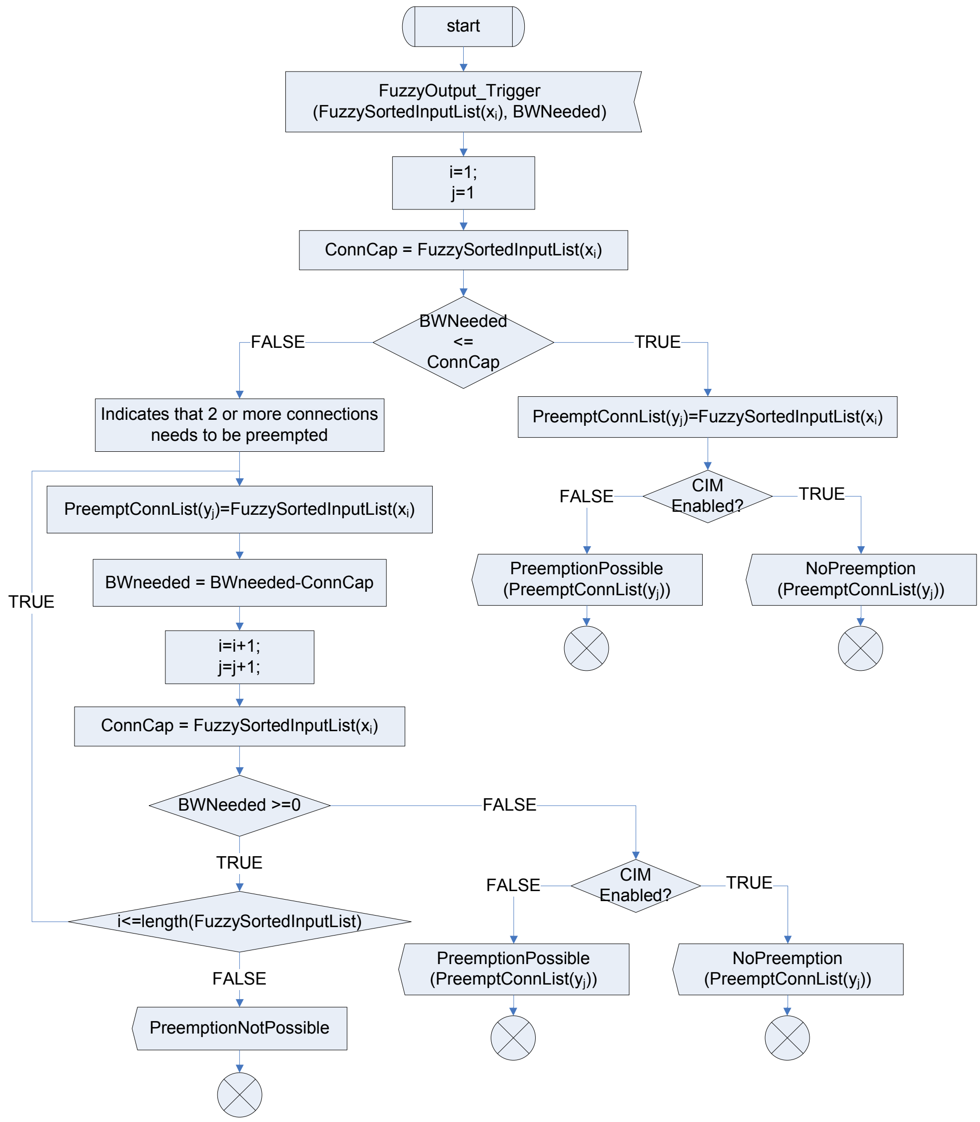

Figure 4-27 Flow Chart for Fuzzy Output

3. However, if the condition is false, more than one connection is required to be dropped in order to admit the new connection. Hence, the algorithm iterates in a loop forming the list, *PreemptConnList*, by selecting the connections from the sorted list until one of the following condition becomes true:

   a. $BWNeeded \leq \sum_{AllSelectedConnections} ConnCap$

   b. Sorted list is fully traversed

4. If condition 3a is satisfied, depending on whether CIM is enabled, one of the following signal is sent back to the CAC Processor:

    a. *PreemptionPossible(PreemptConnList)* if no CIM is enabled

    b. *NoPreemption(PreemptConnList)* if CIM is enabled

5. However, if condition 3b is satisfied indicating that the bandwidth required by the new connection is much higher than all the pre-emptable connections listed in *PreemptConnList,* therefore the pre-emption is not possible and hence, *PreemptionNotPossible* signal is sent back to CAC Processor.

4.3 Summary

This chapter described the CAC framework model proposed for the S-UMTS system satisfying the specified general RRM requirements. The proposed framework consists of three different functionalities; the CAC Processor performing the admission control algorithms, the Retune Controller performing the retune process, and the Pre-emption Controller performing the pre-emption procedure. Two admission control algorithms are considered, the non-adaptive and the adaptive admission control algorithms. The non-adaptive algorithm is presented as a baseline admission control algorithm based on the equivalent capacity approach which does not consider the variable link capacity of the system. The adaptive admission control algorithm is proposed to extend the non-adaptive algorithm by considering the changing link condition and the class of the MT to calculate the resources utilized by the connections. The system supports both unicast and the multicast mode of transmission. Separate admission decision processes are provided for each type of transmissions. For the multicast traffic, the data rate supported by a multicast group is revaluated by the adaptive admission control algorithm every time a new multicast connection joins an existing multicast group. This is required to ensure that the multicast data is received by the different class of MTs belonging to a particular multicast group supporting different data rate. The Retune Controller aims to avoid the blocking of the new connection request in case the given forward subband is congested. It checks the possibility of admitting the new connection request by possible retune of the new connection or any other existing connections from the given forward subband to another forward subband with enough resources. The Pre-emption Controller in the framework allows the system to support the prioritization and pre-emption capability. Such features allow the system to provide service to the higher priority connections such as emergency connections during the congestion state where it is acceptable to drop some lower priority connections to admit a higher priority connections Three pre-emption algorithms have been proposed; Greedy, SubSetSum, and Fuzzy. The Greedy pre-emption algorithm performs by pre-empting the connections with the lowest resource usage. The aim of the algorithm is to minimize the bandwidth released by the pre-empting connections. However in doing so, it greedily pre-empts more connections. The SubSetSum pre-emption algorithm provides an improvement over the Greedy pre-emption algorithm in terms of reducing the number of pre-emptable connections while minimizing the bandwidth released by the pre-empting connections. The drawback of the SubSetSum pre-emption algorithm lies in its mathematical complexity and a higher computation time. The Fuzzy pre-emption algorithm overcomes the drawbacks of the SubSetSum pre-emption algorithm by making use of the expert system knowledge to provide a better system performance. A CIM procedure has also been proposed in conjunction with the pre-emption procedure to further enhance the performance of the system. This mechanism allows the pre-emptable connections to be converted into the '*Non-Active*' state for a certain period of time as opposed to dropping the connections in the first place. Under the '*Non-Active*' state, the resources used by the selected connections are released allowing the admittance of a higher priority connection. At the end of the elapsed time, the '*Non-Active'* connections request for resources again. The connections are either converted into '*Active*' state in case the resources are granted, or the connections are dropped in case the resources are not available.

Chapter 5 : SIMULATION FRAMEWORK

5.1 Introduction

This chapter describes the simulation framework designed using MATLAB to study the performance of the proposed CAC framework for S-UMTS network architecture. MATLAB stands for Matrix Laboratory [100]. The other available simulation tools are Network Simulator (NS2) and OPNET. Although, NS2 provides a number of libraries that can be used for programming the simulation, it has only text based interface. Hence, it is not very user friendly and it is not easy to collect and analyse simulation results for statistical purposes. On the other hand, the OPNET simulation tool provides a graphical user interface for a complete network design which is more intuitive than the text based interface of NS2. However, for the simulation of the proposed CAC framework, it is not required to simulate a complete network design. The framework focuses on the number of algorithms that have been proposed. The MATLAB simulation tool is highly suitable for such purposes. It is a high-performance language which integrates computing, visualization, and programming in an easy-to-use environment. MATLAB has been developed by MathWorks who also develop Simulink, which along with MATLAB has been used to develop the CAC framework. Simulink is a graphical extension of MATLAB which can be used to model, simulate and analyse a system using a Graphical User Interface (GUI) environment. It includes a comprehensive range of blocksets including fuzzy logic, control, communication and power for all types of system requirements. Also, it allows the user to customize or create new blocks. Simulink is an integral part of MATLAB and hence, there is an easy transfer of data between the two programs allowing the user to take full advantage of the features provided in both the environments. Using Simulink, the system model can be organised into a hierarchy and hence, the model can be built using both top-down and bottom-up approaches.

The remainder of this chapter explains the MATLAB Simulink model of the proposed CAC framework in detail.

5.2 MATLAB Simulink System Modelling

Figure 5-1 shows the highest level view of the MATLAB Simulink model of the proposed CAC framework for S-UMTS network architecture. As mentioned earlier the Simulink modelling is hierarchical and hence, the system can be viewed at a high level and then double-click blocks to go down through the levels to see increasing levels of model detail [100].

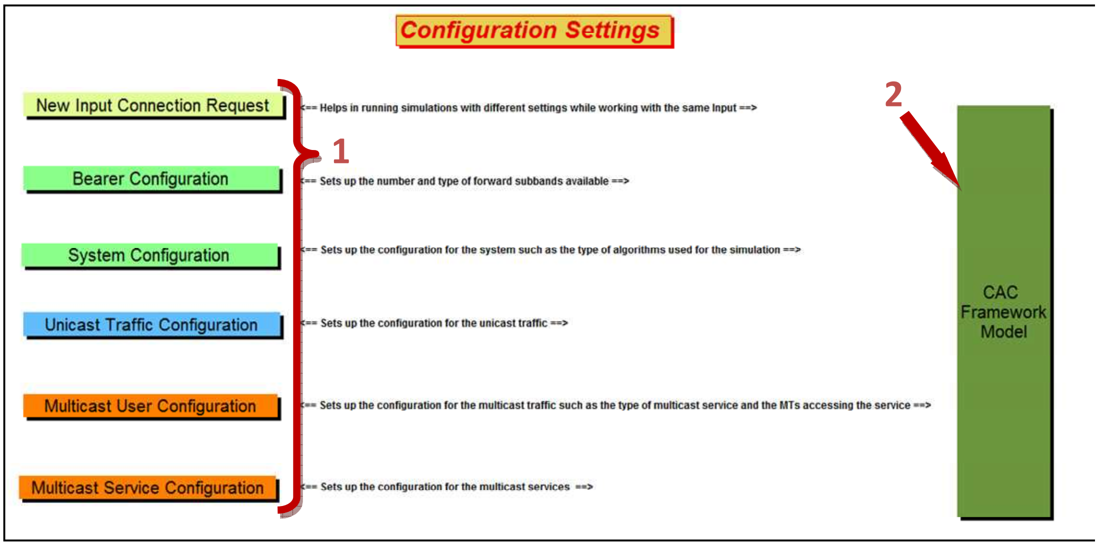

Figure 5-1 Highest level view of the MATLAB Simulink Model of the CAC framework for S-UMTS network architecture

As can be seen, the highest level contains the following blocks which are explained in detail in the following subsections:

- 1 – *Graphical User Interface*. This consists of the configuration settings for the various elements of the model.
- 2 – *CAC Framework Model*. This contains the lower levels of the model.

5.2.1 Graphical User Interface (GUI)

The GUI has been created to provide the user with the flexibility to configure the system settings before starting a simulation. In order to create a custom user interface, Simulink employs masking a subsystem using the *Simulink Mask Editor* which allows the user to define a set of user-configurable parameters. Each mask parameter is linked to a specific block parameter inside the masked subsystem, such that setting a mask parameter sets the associated block parameter. For the CAC framework proposed in this thesis, the configuration settings are divided into following six groups:

- *New Input Connection Request*
- *Bearer Configuration*
- *System Configuration*
- *Unicast Traffic Configuration*
- *Multicast User Configuration*
- *Multicast Service Configuration*

Each group is created using a masked subsystem.

a) New Input Connection Request

Figure 5-2 shows the *New Input Connection Request* subsystem. This configuration block provides the flexibility to run the simulation with different configuration settings; for example changing the number of the available forward subbands, the retune condition; while working with the same input connection queue.

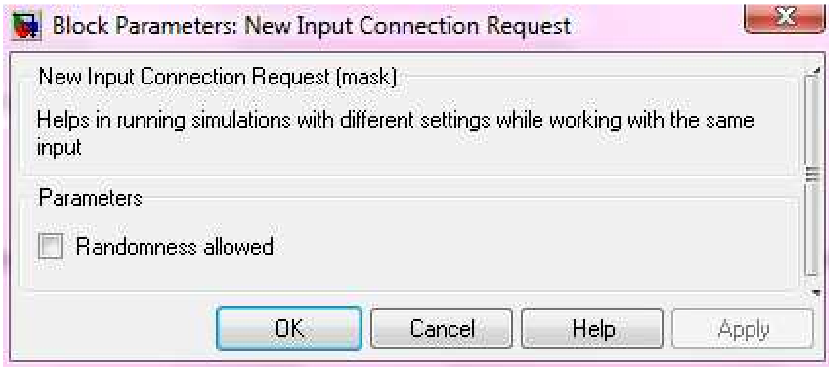

Figure 5-2 *New Input Connection Request* subsystem configuration block

Table 5-1 shows the masked parameter for the *New Input Connection Request* subsystem.

Masked Parameters	Values taken
Randomness Allowed	*Selected* - new input connection queue *Unselected* – existing input connection queue

Table 5-1 *New Input Connection Request* masked parameter

This subsystem uses only one parameter, *Randomness Allowed,* which takes two values either *Selected* or *Unselected.* In the *Selected* state, the simulation generates a new input connection queue such that a new set of inter-arrival times (time between the connection requests) and service times for the connections (length of time a connection is active) are created using the exponential distribution. Once the simulation has run once with this new input connection queue, the user can either create a further new input connection queue for the next simulation by keeping the parameter, *Randomness Allowed,* as *Selected,* or, in order to run another simulation with different configuration settings for the same input connection queue, the parameter, *Randomness Allowed,* is set as *Unselected.*

b) Bearer Configuration

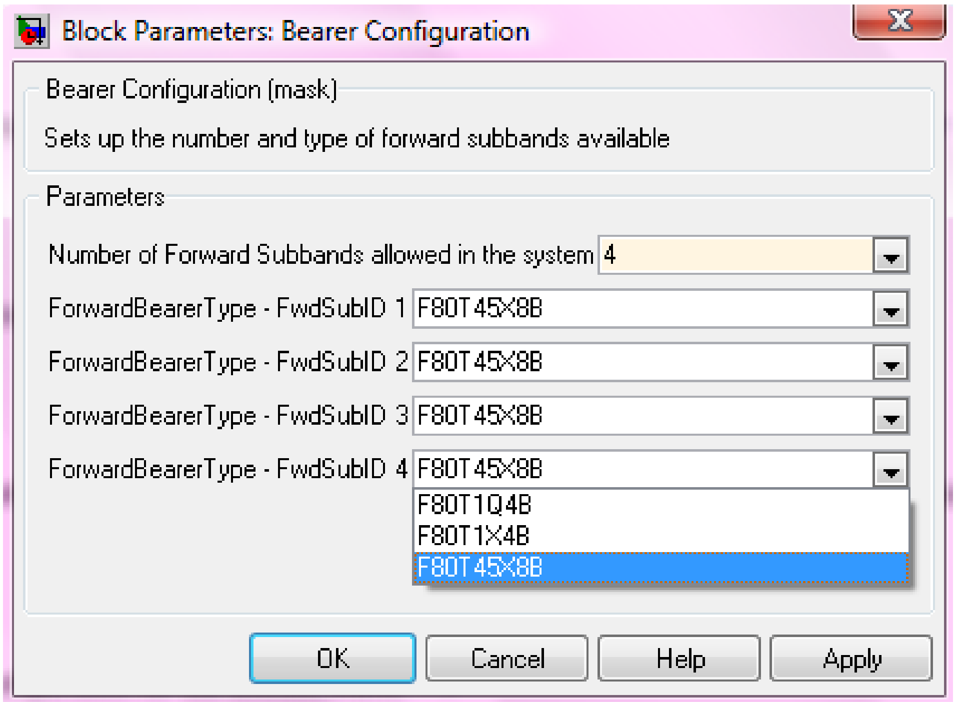

Figure 5-3 *Bearer Configuration* subsystem configuration block

Figure 5-3 shows the *Bearer Configuration* subsystem. This configuration block allows the configuration of the forward subbands or bearers for a given simulation. Table 5-2 shows the masked parameters for the *Bearer Configuration* subsystem.

Masked Parameters		Value taken
1	*Number of forward subbands allowed in the system*	1-8
2-9	*ForwardBearerType-FwdSubID1-8*	F80T1Q4B, F80T1X4B, F80T4.5X8B

Table 5-2 *Bearer Configuration* masked parameters

This subsystem has a maximum of nine masked parameters. The first parameter, *Number of forward subbands allowed in the system*, indicates the maximum number of available forward subbands, which lies between 1 and 8. Once this parameter is set, the option to set the value for the other parameter, *ForwardBearerType-FwdSubID,* for the chosen number of forward subbands appears. The *ForwardBearerType-FwdSubID* parameter indicates the type of the forward subband which takes 3 values: F80T1Q4B, F80T1X4B, and F80T4.5X8B.

c) System Configuration

Figure 5-4 shows the *System Configuration* subsystem. This configuration block allows the system to be defined in terms of parameters such as the link condition, supported MT classes etc.

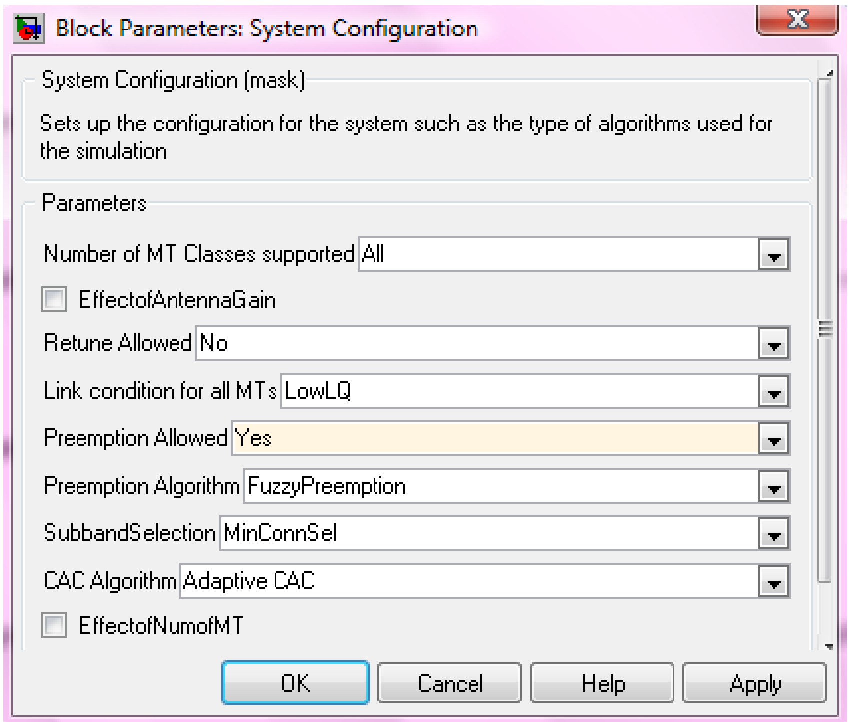

Figure 5-4 *System Configuration* subsystem configuration block

Table 5-3 shows the masked parameters for the *System Configuration* subsystem.

Masked Parameters		Value taken
1	*Number of MT Classes supported*	*All, 1, 2, 3*
2	*EffectofAntennaGain*	*Selected, Unselected*
3	*Retune Allowed*	*Yes, No, Random*
4	*Link condition for all MTs*	*Low, High, Random*
5	*Pre-emption Allowed*	*Yes, No, Random*
6	*Pre-emption Algorithm*	• *Greedy* • *GreedywithCIM* • *SubsetSum* • *SubsetSumwithCIM* • *Fuzzy* • *FuzzywithCIM*
7	*SubbandSelection*	*MinConnSubSel, Random*
8	*CAC Algorithm*	*Adaptive, Non-Adaptive*
9	*EffectofNumofMT*	*Selected, Unselected*

Table 5-3 *System Configuration* masked parameters

This subsystem has nine masked parameters. The first parameter, *Number of MT Classes supported*, allows the system to support one of the following: only Class1 MTs, only Class2 MTs, only Class3 MTs, or all types of MT classes. In the case of the last value being selected, the MT class is selected randomly for each input connection request. The second parameter, *EffectofAntennaGain,* allows the system to be analyzed for the effect of different classes of MTs whilst working with the same input connection queue. In the *Selected* state, the system can rerun with the same input connection queue whilst changing the *Number of MT Classes supported* for each simulation. In the *Unselected* state, the simulation uses the existing *Number of MT Classes supported* parameter. The third masked parameter, *Retune Allowed,* allows the system to be evaluated under different retune conditions. The parameter takes three values: all MTs are retunable, no MTs are retunable, and some MTs are selected on a random basis to be retunable. Similarly the fourth parameter, *Link Condition for all MTs,* allows the system to be evaluated under different link conditions. Again, the parameter takes three values: all MTs are in low link condition, all MTs are in high link condition, and the link condition for all the MTs lies randomly between low and high link conditions. The fifth parameter, *Pre-emption Allowed,* is used to analyze the system with pre-emption capability. The parameter takes three values: all connections are pre-emptable, no connections are pre-emptable, and some connections are selected to be pre-emptable on a random basis. Under the situation when some or all MTs are pre-emptable, the next parameter, *Pre-emption Algorithm*, is enabled otherwise it is disabled. *Pre-emption Algorithm* allows the user to select any of the three pre-emption algorithms proposed in section 4.2.4 with or without CIM. The seventh parameter, *SubbandSelection*, is used to analyze the system using the different forward subband selection methods described in section 4.2.2.1.1, which are either *MinConnSubSel* or *Random.* The next parameter, *CAC Algorithm,* allows the performance of the system to be compared using the *Adaptive* and the *Non-Adaptive* algorithm. Finally the last parameter, *EffectofNumofMT*, allows the simulation to be rerun using a different number of users while keeping the same input connection queue.

d) Unicast Traffic Configuration

Figure 5-5 shows the *Unicast Traffic Configuration* subsystem. This configuration block allows the configuration of the unicast traffic such as the number of unicast connections generated and the traffic characteristics.

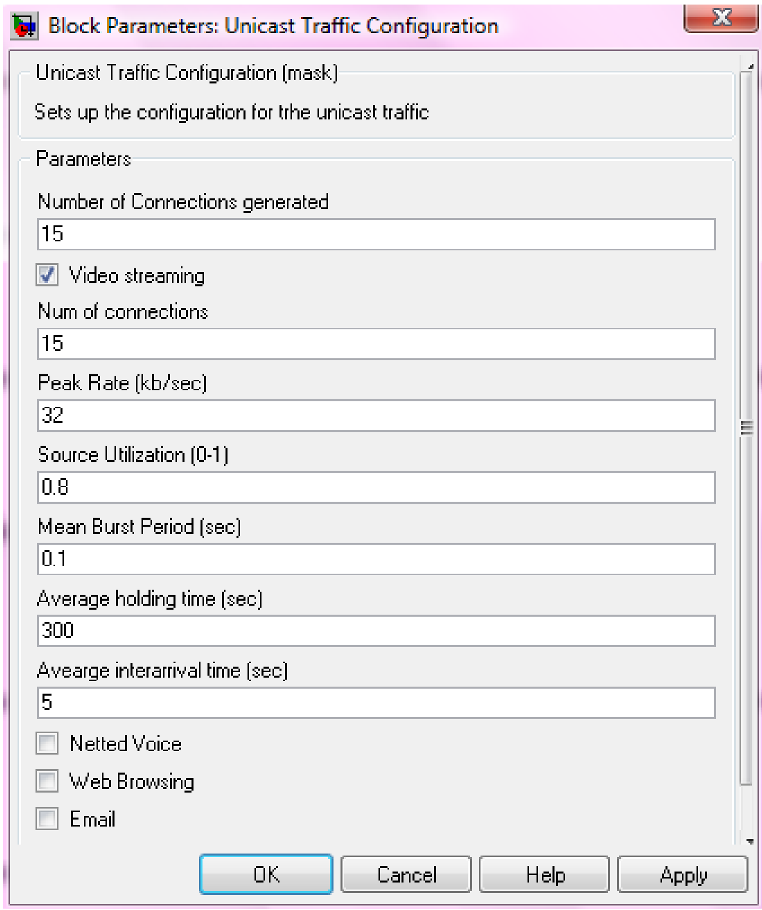

Figure 5-5 *Unicast Traffic Configuration* subsystem configuration block

Table 5-4 shows the masked parameters for the *Unicast Traffic Configuration* subsystem. The first parameter, *Number of Connections generated,* indicates the total number of unicast connections for all traffic types to be generated for the simulation and is chosen by the user. The parameters, *Video Streaming*, *Netted Voice*, *Web Browsing*, and *Email*, indicate the type of unicast traffic supported for the simulation. These parameters take two values, *Selected* or *Unselected.* In the *Selected* state, the remaining parameters defining the characteristics of the traffic are enabled for each type of traffic. These parameters are: *Number of connections* indicating the number of connections for the particular traffic type, *Peak Rate* in kb/sec, *Source Utilization* whose value lies between 0 and 1, *Mean Burst Period* in seconds, *Average holding time* in seconds, and *Average interarrival time* in seconds. All the traffic characteristics are set by the user and any of the characteristics can be varied to check the effect on the performance of the system if any. In the *Unselected* state, the traffic characteristics of the particular traffic type are not shown and hence, are disabled.

Masked Parameters		Value taken
1	*Number of Connections generated*	To be chosen by the user
2	*Video Streaming*	*Selected, Unselected*
3	*Netted Voice*	*Selected, Unselected*
4	*Web Browsing*	*Selected, Unselected*
5	*Email*	*Selected, Unselected*
6	*Number of connections*	To be chosen by the user
7	*PeakRate (kb/sec)*	To be chosen by the user
8	*Source Utilization (0-1)*	To be chosen by the user
9	*Mean Burst Period (sec)*	To be chosen by the user
10	*Average holding time (sec)*	To be chosen by the user
11	*Average interarrival time (sec)*	To be chosen by the user

Table 5-4 *Unicast Traffic Configuration* masked parameter

e) Multicast User Configuration

Figure 5-6 shows the *Multicast User Configuration* subsystem. This configuration block defines the user configuration for multicast enabled system such as indicating the number of users accessing the type of the multicast service.

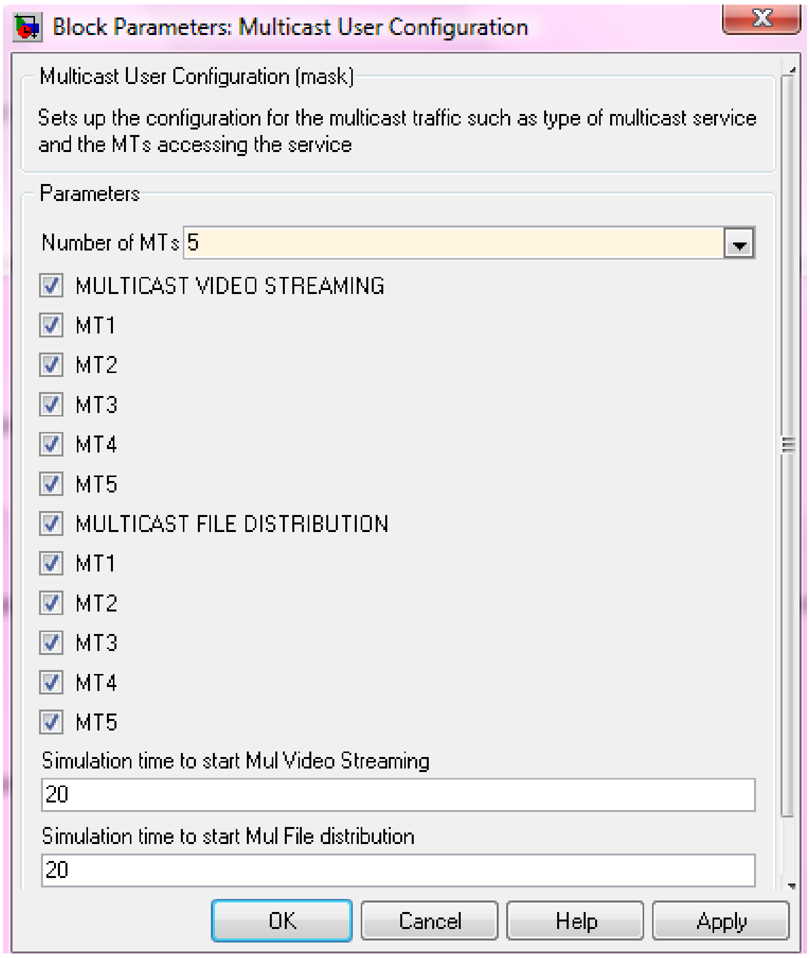

Figure 5-6 *Multicast User Configuration* subsystem configuration block

Table 5-5 shows the masked parameters for the *Multicast User Configuration* subsystem. The first parameter, *Number of MTs,* indicates the number of users present in the system. The parameter value appears as a drop down menu option containing four values: 5, 10, 15, and 20. It should be noted that for this thesis work, although the number of users are restricted to the pre-defined values, there is no restriction on the number of connections generated by each user. The restriction on the number of users is purely for simulation purposes. For unicast traffic, it is possible to see the effect of the number of users on the performance of the system, if any, by selecting the *EffectofNumofMT* parameter of the *System Configuration* subsystem. For multicast traffic, the configuration block allows the user to select individually for each MT, the multicast service(s) the MT wants to access. The parameters, *MULTICAST VIDEO STREAMING* and *MULTICAST FILE DISTRIBUTION* indicate the multicast services supported by the system. These parameters take two values, *Selected* or *Unselected.* In the *Selected* state, the MTs equivalent to the value of *Number of MTs* selected appears. The user can then configure the MTs accessing a particular multicast service. In addition, the user can also configure the simulation time at which to start the individual multicast service by specifying the values for the parameters, *Simulation time to start Mul Video streaming* and *Simulation time to start Mul File distribution.* These parameters allow the system to be evaluated by providing flexibility to start multicast services at different simulation times.

Masked Parameters		Value taken
1	*Number of MTs*	*5,10,15,20*
2	*MULTICAST VIDEO STREAMING*	*Selected, Unselected*
3	*MULTICAST FILE DISTRIBUTION*	*Selected, Unselected*
4	*MT1,MT2……MT20*	*Selected, Unselected*
5	*Simulation time to start Mul Video streaming*	To be chosen by the user
6	*Simulation time to start Mul File distribution*	To be chosen by the user

Table 5-5 *Multicast User Configuration* masked parameters

f) Multicast Service Configuration

Figure 5-7 shows the *Multicast Service Configuration* subsystem. This configuration block allows the configuration of the multicast traffic.

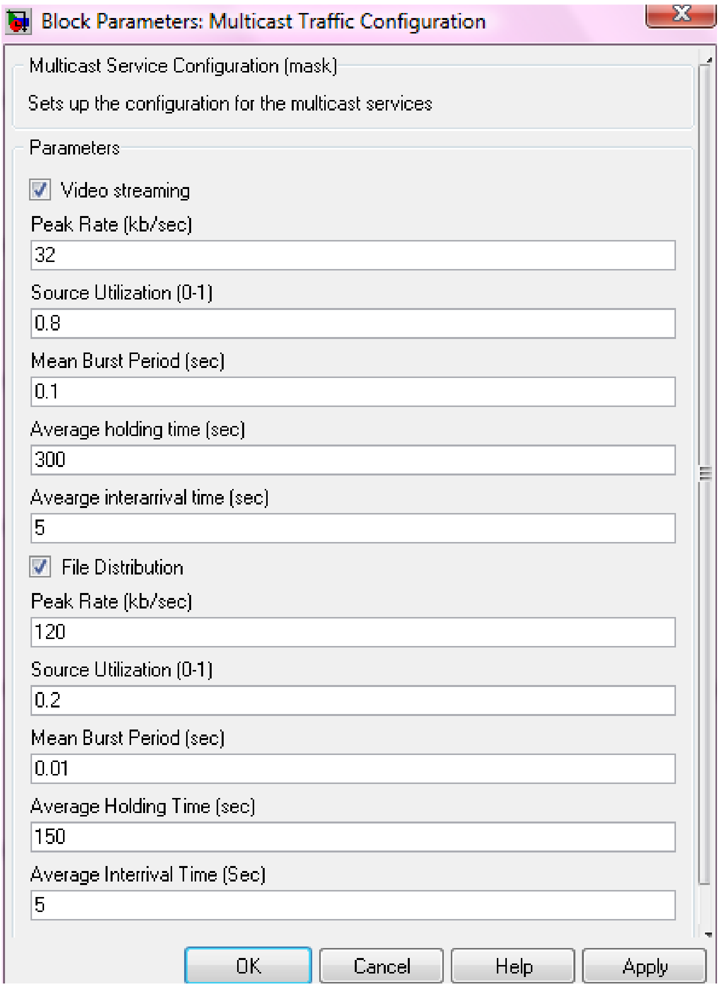

Figure 5-7 *Multicast Service Configuration* subsystem configuration block

Table 5-6 shows the masked parameters for the *Multicast Service Configuration* masked subsystem.

Masked Parameters		Value taken
1	*Video Streaming*	*Selected, Unselected*
2	*File Distribution*	*Selected, Unselected*
3	*PeakRate (kb/sec)*	To be chosen by the user
4	*Source Utilization (0-1)*	To be chosen by the user
5	*Mean Burst Period (sec)*	To be chosen by the user
6	*Average holding time (sec)*	To be chosen by the user
7	*Average interarrival time (sec)*	To be chosen by the user

Table 5-6 *Multicast Service Configuration* masked parameter

The parameters, *Video Streaming*, and *File Distribution*, indicates type of multicast traffic supported for the given simulation. These parameters take two values, *Selected* or *Unselected.* In the *Selected* state, the remaining parameters defining the characteristics of the traffic are enabled for each type of traffic which are: *Peak Rate*, *Source Utilization* whose value lies between 0 and 1, *Mean Burst Period*, *Average holding time,* and *Average interarrival time.* In the *Unselected* state, the traffic characteristics of the particular traffic type are not shown and hence, are disabled.

5.2.2 CAC Framework model

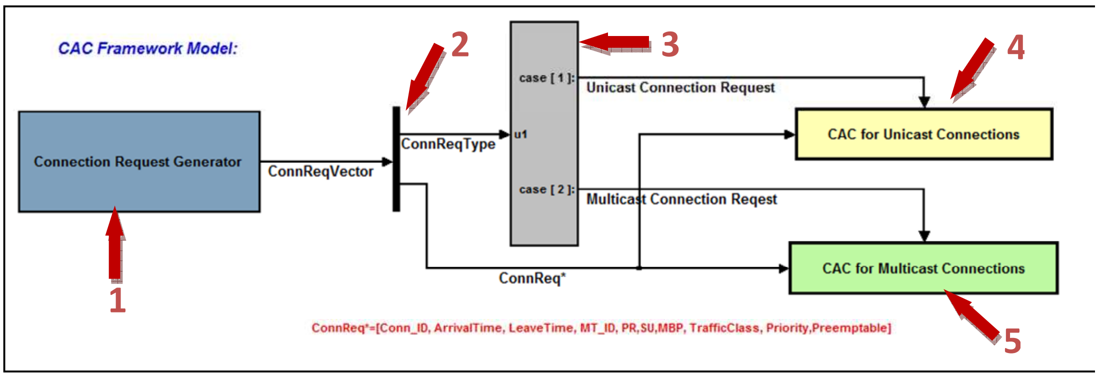

Figure 5-8 Second level of the MATLAB Simulink CAC framework model

Figure 5-8 shows the second level of the MATLAB Simulink CAC framework model. As can be seen, the framework model is built using the following Simulink blocks:

- 1 – *Connection Request Generator*: a user-defined block.
- 2 – *Demux*: a Simulink library block which extracts the components of an input signal and outputs the components as separate signals.
- 3 – *Switch Case*: a Simulink library block that receives a single input which is used to form the case conditions that determine which subsystem to execute.

- 4 – *CAC for Unicast Connections*: a user-defined switch case subsystem block triggered by the *Switch Case* block for a unicast connection request.
- 5 – *CAC for Multicast Connections*: a user-defined switch case subsystem block triggered by the *Switch Case* block for a multicast connection request.

The following subsections describe the user-defined blocks: *Connection Request Generator, CAC for Unicast Connections* and *CAC for Multicast Connections* in detail.

5.2.2.1 Connection Request Generator

The *Connection Request Generator* block is responsible for creating the input connection queue according to the arrival times of the connections set in the configuration using GUI. As mentioned earlier, the *Connection Request Generator* is a user-defined block which is created programmatically in the Simulink by writing an M-file containing the functions of the block, as described in the section 4.2.1, where the resulting file is called S-Function. Figure 5-9 shows the dialog box of the *Connection Request Generator S-function* block.

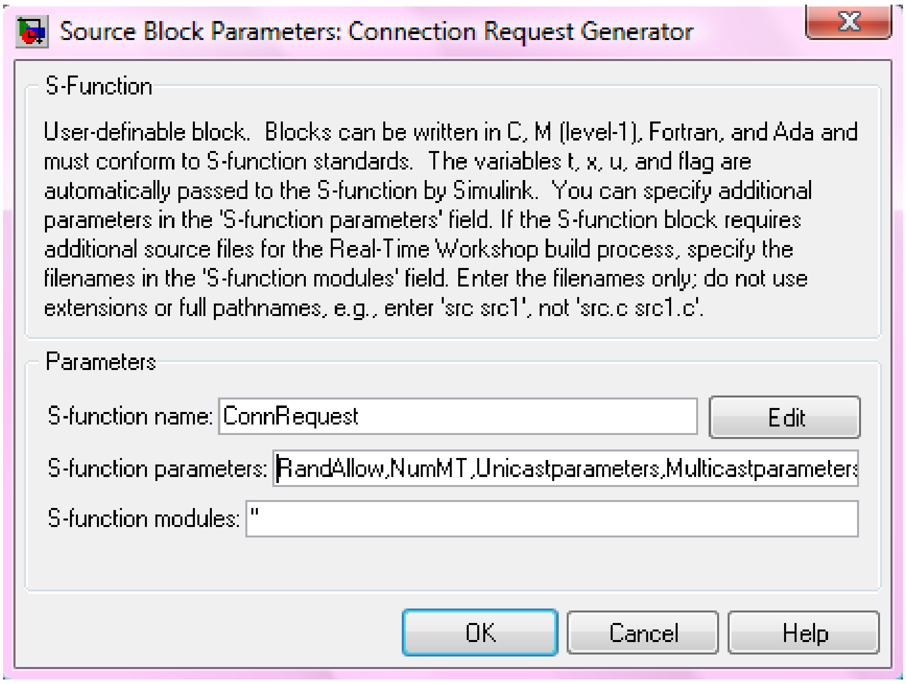

Figure 5-9 *Connection Request Generator S-function* block dialog box

As can be seen, it uses two parameters: *S-Function name* which contains the MATLAB M-file name, *ConnRequest*, and the *S-Function parameters* carrying the masked parameters from the configuration settings which are required in the corresponding S-function. For every connection, the block produces an output, *ConnReqVector,* containing the information shown in Table 5-7. The output, *ConnReqVector*, is sent to the *Demux* block which separates the signal into two parts; *ConnReqType* consisting of the first parameter and the *ConnReq* consisting of the remaining parameters of the *ConnReqVector.* The *ConnReqType* parameter which defines the type of the connection request (unicast or multicast) is sent to the *Switch Case* block as an input. The *Switch Case* block then triggers the appropriate switch case subsystem block; *CAC for Unicast Connections*, or *CAC for Multicast Connections.* The *ConnReq* is sent as an input signal to each of the subsystem blocks.

Parameters		Description
1	*ConnReqType*	*ConnReqType* =1: Unicast connection *ConnReqType* =2: Multicast connection
2	*Conn_ID*	Unique ID for a connection
3	*ArrivalTime*	Arrival time of the connection
4	*LeaveTime*	Service time of the connection
5	*MT_ID*	ID of the MT to which the connection belongs
6	*PR*	Peak Rate of the connection in bits/80ms for an 80ms physical frame
7	*SU*	Source Utilization of the connection
8	*MBP*	Mean Burst Period of the connection
9	*TrafficClass*	Traffic class of the connection
10	*Priority*	Priority of the connection
11	*Preemptable*	*Preemptable* = 0: Not a pre-emptable connection *Preemptable* = 1: A pre-emptable connection

Table 5-7 Parameters of the *ConnReqVector* output of the *Connection Request Generator* block

5.2.2.2 CAC for Unicast Connections

CAC for Unicast Connections is a user-defined Simulink block which is implemented using the *While Iterator Subsystem* Simulink library block. A *While Iterator Subsystem* executes repeatedly until the while condition is satisfied.

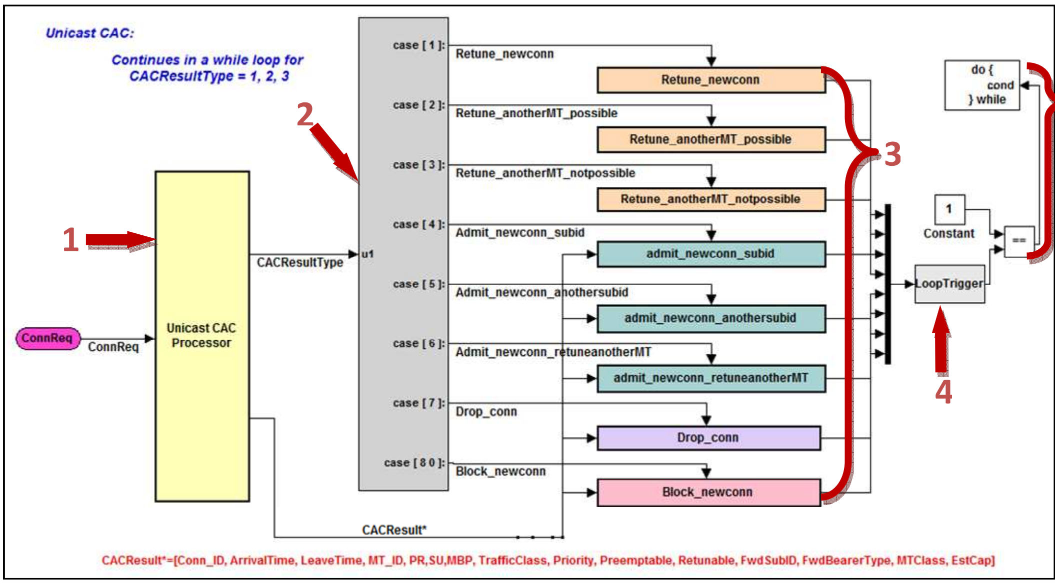

Figure 5-10 Third level of the MATLAB Simulink CAC framework model – *CAC for Unicast Connections*

Figure 5-10 shows the third level of the MATLAB Simulink CAC framework model for the *CAC for Unicast Connections* block. As can be seen, the model is built using the following Simulink blocks:

- 1 – *Unicast CAC Processor*: a user-defined subsystem block. It contains the main CAC algorithms.
- 2 – *Switch Case*.
- 3 – *Unicast Switch Case Subsystems*. This constitutes the group of eight subsystems which are triggered by the *Switch Case* block based on the value of the parameter, *CACResultType.*
- 4 – *Loop Trigger*: a user-defined S-function block, which takes the output value, which is either 0 or 1, from one of the 8 triggered subsystems and forwards the value to the *While Iterator* block.
- 5 – *While Iterator*: a Simulink library block which when placed in a subsystem, repeatedly executes the contents of the subsystem at the current time step until the specified condition is true [100]. The

condition is based on the output from the *Loop Trigger* block. If the value is 1, the while condition becomes true resulting in the subsystem to be executed again. However, if the value is 0, the while condition becomes false, and hence, the execution stops.

The following subsections describe the user-defined blocks: *Unicast CAC Processor,* and the eight *Switch Case Subsystems* in detail.

5.2.2.2.1 Unicast CAC Processor

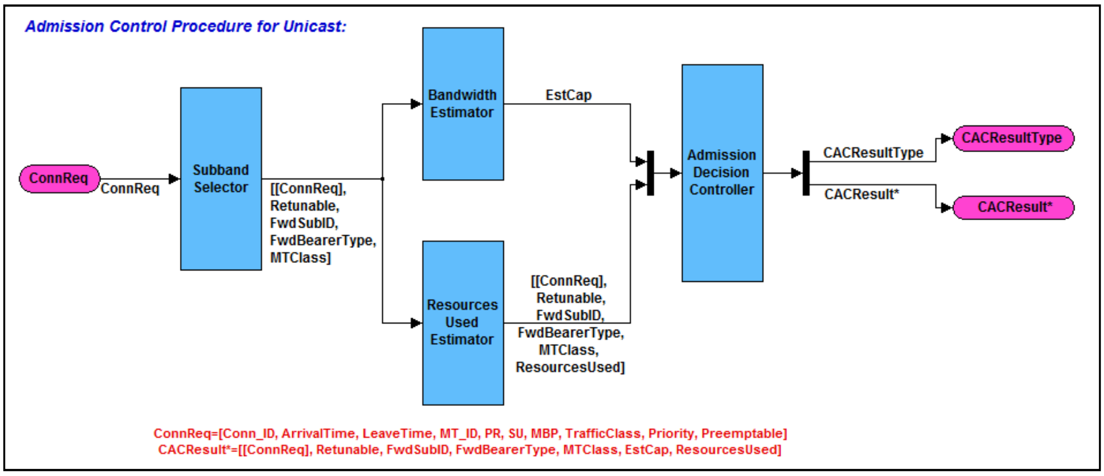

Figure 5-11 Fourth level of the MATLAB Simulink CAC framework model – *Unicast CAC Processor*

Figure 5-11 shows the fourth level of the MATLAB Simulink CAC framework model in terms of the *Unicast CAC Processor* block*.* The model is based on the functional block model proposed in the section 4.2.2.1. This model consists of the following S-function Simulink blocks:

- *Subband Selector* – takes the input, *ConnReq*, from the *Connection Request Generator* block and, based on the type of the subband selection method selected during the configuration, the new connection request is assigned a forward subband. Table 5-8 shows the output of the *Subband Selector* Simulink block.

Parameters		Description
1	*[ConnReq]*	Input signal from the *Connection Request Generator* block
2	*Retunable*	Takes the value 0 or 1, based on the value of the *Retune Allowed* configuration parameter. 0 indicates the connection cannot retune, and 1 indicates the connection is allowed to retune.
3	*FwdSubID*	Unique ID of the selected forward subband
4	*FwdBearerType*	Takes values 1-3 depending on the value of the *ForwardBearerType-FwdSubID* configuration parameter selected.
5	*MTClass*	Takes values 1-3 depending on the value of the *Number of MT Classes supported* configuration parameter selected

Table 5-8 Output of the *Subband Selector* Simulink block

- Bandwidth Estimator – estimates the capacity used by the connection, *EstCap*, as explained in the section 4.2.2.1.2.
- Resources Used Estimator – calculates the total resources used on the given forward subband, *ResourcesUsed*, depending on the adaptive or the non-adaptive admission control algorithm selected during the initial configuration as explained in the section 4.2.2.1.3.
- Admission Decision Controller – performs the admission decision and produces the output which is split into two signals, *CACResultType*, and *CACResult. CACResultType* decides the type of the *Switch Case Subsystem* to be triggered and the *CACResult* is sent as an input signal to the triggered subsystem block. Table 5-9 describe the values taken by the *CACResultType* parameter.

CAC Result Type	*Switch Case Subsystems*	Description
1	*Retune_newconn*	Checks for the possibility of retuning the new connection.
2	*Retune_anotherMT_possible*	Checks for the possibility of retuning another MT on the given forward subband to another forward subband.
3	*Retune_anotherMT_notpossible*	The above two conditions are not possible; hence, the possibility of pre-emption is checked.
4	*Admit_newconn_subid*	Admits the new connection on the given forward subband.
5	*Admit_newconn_anothersubid*	Admits the new connection on another forward subband by retuning.
6	*Admit_newconn_retuneanotherMT*	Admits the new connection on the given forward subband by retuning some MT on another forward subband.
7	*Drop_conn*	The selected connections for pre-emption are dropped.
8	*Block_newconn*	Blocks the new connection.

Table 5-9 Values taken by *CACResultType* parameter

The *CAC for Unicast Connections* subsystem is executed in a loop for the values of the *CACResultType:* 1, 2, and 3 and the execution is halted for the current time step for the values of the *CACResultType:* 4, 5, 6, 7, and 8.

5.2.2.2.2 Unicast Switch Case Subsystems

Referring to Table 5-9, the three switch case subsystems: *Retune_newconn, Retune_anotherMT_possible, and Retune_anotherMT_notpossible,* are built as shown in Figure 5-12.

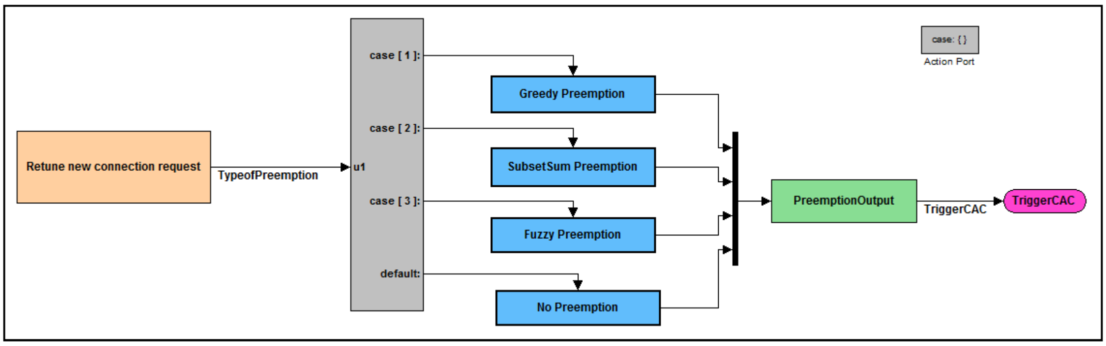

Figure 5-12 Fourth level of the MATLAB Simulink CAC framework model – *Retune_newconn*

Figure 5-12 shows an example for *Retune_newconn* subsystem. As can be seen, the model consists of an S-function Simulink user-defined block, *Retune new connection request,* which produces the output, *TypeofPreemption*. Under the condition when retuning is allowed and the connection request can be admitted into the system by retuning, the value of *TypeofPreemption* is set to a default value other than 1, 2 or 3, indicating there is no need for pre-emption procedure. However, in the case when pre-emption is enabled and the connection request can be admitted only by pre-empting one or more existing connections, depending on the configuration settings, the *TypeofPreemption* is assigned the value 1,2, or 3. This output is fed into a *Switch Case* block and the appropriate *Switch Case Subsystem* is triggered. The *Greedy Preemption* and the *SubsetSum Preemption* subsystems consist of a user defined S-function Simulink block containing the Greedy pre-emption algorithm and the SubSetSum pre-emption algorithm as described in the sections 4.2.4.2 and 4.2.4.3 respectively. The *Fuzzy Preemption* subsystem is implemented using a *While Iterator Subsystem* Simulink block which continues in a loop until the list of the connections to be preempted has been fully traversed as shown in Figure 5-13.

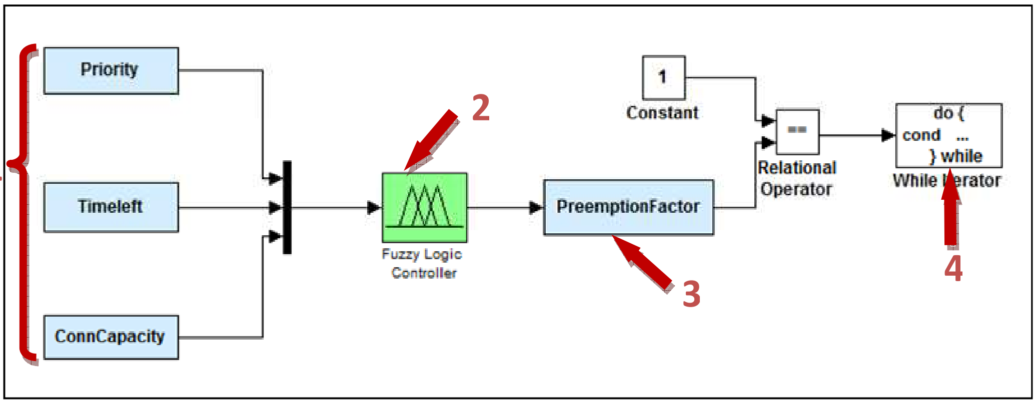

Figure 5-13 Fifth level of the MATLAB Simulink CAC framework model – *Fuzzy Preemption*

The *Fuzzy Preemption* is implemented based on the description provided in the section 4.2.4.4. As can be seen, the subsystem consists of the following simulink blocks:

- 1 – Input user-defined S-function blocks: *Priority*, *Timeleft*, and *ConnCapacity*. The function of these input blocks is to extract the appropriate information from the list of the pre-emptable connections and forward the information to the Fuzzy Logic Controller (FLC).

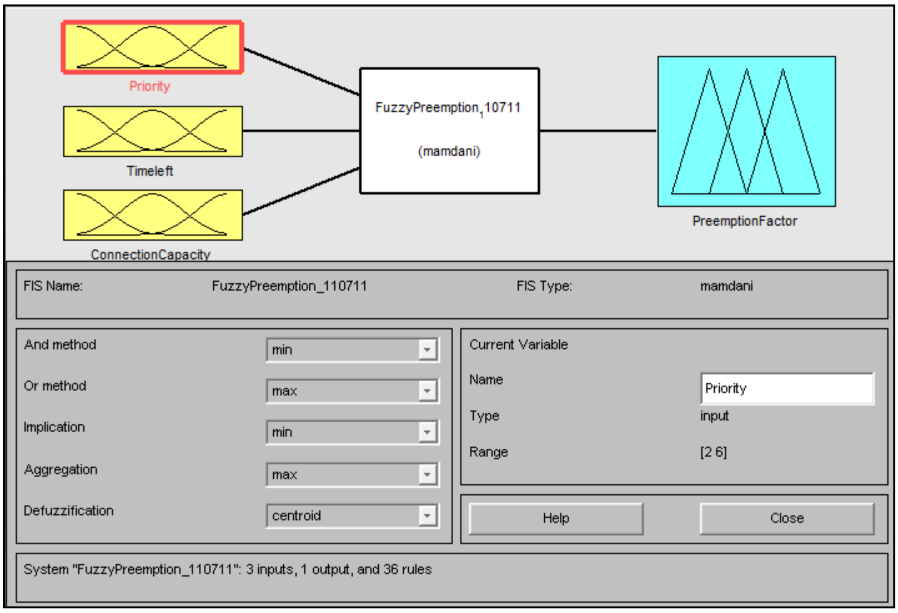

Figure 5-14 *FuzzyPreemption* Fuzzy Inference System Editor

- 2 – Fuzzy Logic Controller: a Simulink library block which implements the Fuzzy Inference System (FIS). The FIS for the *Fuzzy Preemption* is shown in Figure 5-14. It consists of the fuzzy inputs, a Mamdani based fuzzy operator and a fuzzy output.

- 3 – Output user-defined S-function block, *PreemptionFactor*. The function of this block is to obtain the output from the FLC and produce an output 0 or 1 by checking the list of pre-emptable connections. In case the list is fully traversed, it produces the output 0, and hence the execution stops, else the execution continues.
- 4 – *While Iterator*. Same as in section 5.2.2.2.

Referring to Table 5-9, the remaining five switch case subsystems are built using a user-defined S-function block such as that shown for *Admit_newconn_subid* subsystem in Figure 5-15.

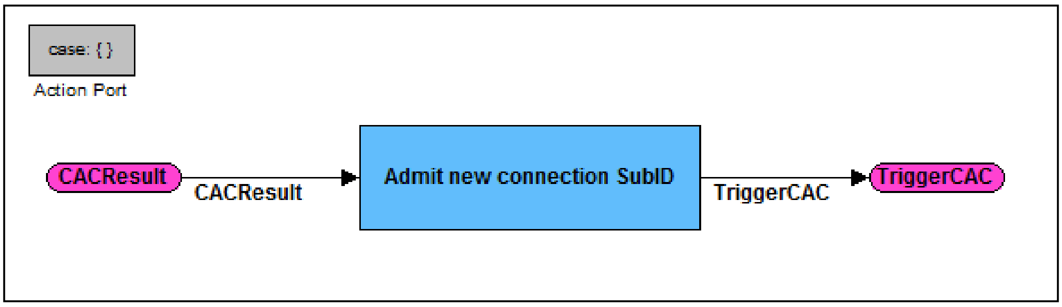

Figure 5-15 Fourth level of the MATLAB Simulink CAC framework model – *Admit_newconn_subid*

The output from these subsystems, *TriggerCAC* is set at value 0 which is received by the *Loop Trigger* simulink block in the upper hierarchical level. This triggers the *While Iterator* block to stop the execution of the *CAC for Unicast Connections* Simulink block.

5.2.2.3 CAC for Multicast Connections

CAC for Multicast Connections is a user-defined Simulink block similar to the *CAC for Unicast Connections* block. It is also implemented using a *While Iterator Subsystem* block as shown in Figure 5-16.

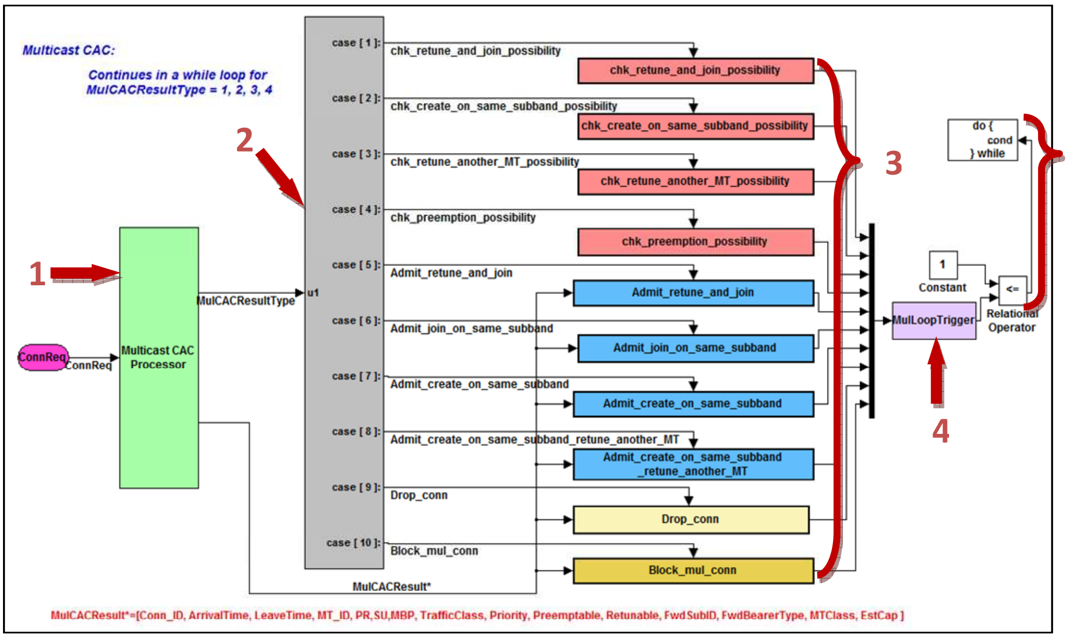

Figure 5-16 Third hierarchy level of the MATLAB Simulink CAC framework model – *CAC for Multicast Connections*

As can be seen, the subsystem consists of the following Simulink blocks:

- 1 – *Multicast CAC Processor*: a user-defined subsystem block.
- 2 – *Switch Case.*
- 3 – *Multicast Switch Case Subsystems*. This constitutes a group of ten subsystems which are triggered by the *Switch Case* block based on the value of the parameter, *MulCACResultType.*
- 4 – *MulLoopTrigger*: a user-defined S-function block. It takes the output value, which is either 0 or 1, from one of the 10 triggered subsystems and forwards the value to the *While Iterator* block.
- 5 – *While Iterator*. Same as in section 5.2.2.2.

The following subsections describe the user-defined blocks: *Multicast CAC Processor,* and the *Multicast Switch Case Subsystems* in detail.

5.2.2.3.1 Multicast CAC Processor

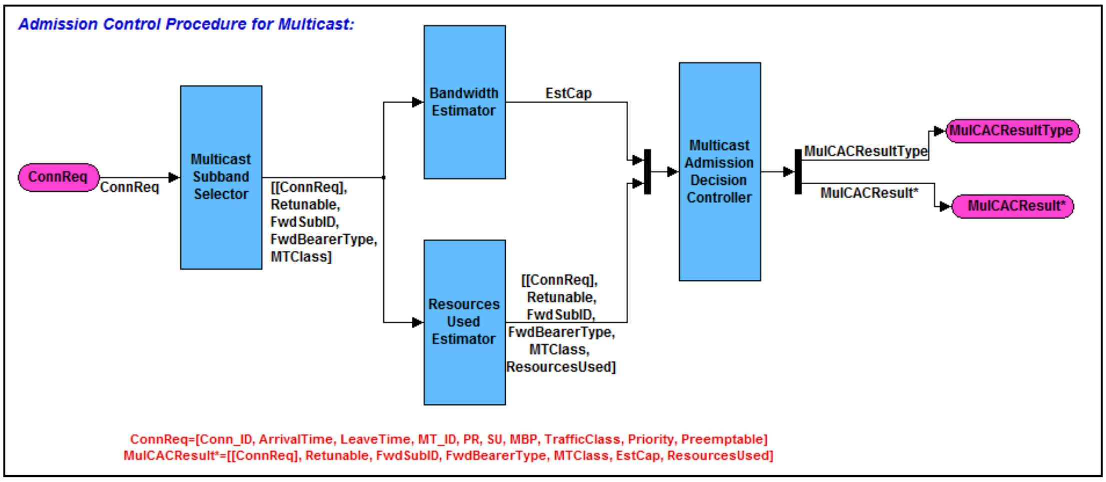

Figure 5-17 Fourth level of the MATLAB Simulink CAC framework model – *Multicast CAC Processor*

Figure 5-17 shows the fourth level of the MATLAB Simulink CAC framework model in terms of *Multicast CAC Processor.* It contains the CAC algorithm for the multicast connections. The model is similar to the *Unicast CAC Processor* except the Simulink blocks; *Multicast Subband Selector* and the *Multicast Admission Decision Controller*, which implement the functions specific to the multicast connections based on the admission decision process for multicast connections described in section 4.2.2.2.2.

5.2.2.3.2 Multicast Switch Case Subsystems

Table 5-10 describe the values taken by the *MulCACResultType* parameter. The *CAC for Multicast Connections* subsystem is executed in a loop for the values of the *MulCACResultType:* 1, 2, 3, and 4 and the execution is halted for the current time step for the values of the *MulCACResultType:* 5, 6, 7 and 8.

MulCAC Result Type	Switch Case Subsystems	Description
1	*chk_retune_ and_join_ possibility*	Checks for the possibility of retuning the new multicast connection and joining the existing multicast session on another forward subband
2	*chk_create_on_ same_subband_ possibility*	Checks for the possibility of creating the multicast session on the given subband
3	*chk_retune_ another_MT_ possibility*	Checks for the possibility of retuning another MT to another forward subband
4	*chk_preemption_ possibility*	Checks the possibility of pre-empting existing unicast connections
5	*Admit_retune_ and_join*	Admits the multicast conn by retuning and joining an existing multicast session
6	*Admit_join_on_ same_subband*	Admits the multicast conn by joining the existing multicast session on the given subband
7	*Admit_create_ on_same_ subband*	Admits the multicast conn by creating a new multicast session on the given subband
8	*Admit_create_ on_same_ subband_retune _another_MT*	Admits the multicast conn by creating a new multicast session on the given subband by retuning some other MT
9	*Drop_conn*	The selected connections for pre-emption are dropped

Table 5-10 Values taken by *MulCACResultType* parameter

The subsystem, *chk_preemption_possibility*, is implemented similar to the process shown in Figure 5-13 which takes into account the pre-emption process while all other subsystems are implemented as an S-function Simulink block, each performing the function as described in Table 5-10.

5.3 Summary

This chapter presented the simulation framework designed for the proposed CAC framework model for the S-UMTS based system. MATLAB and Simulink simulation tools have been used to design and evaluate the performance of the model. Simulink allows the modelling of the system to be more organized as the models are built hierarchically. Various levels of the CAC framework model has been presented and explained in detail. The GUI has been created to provide the user with the flexibility to configure the system settings before starting a simulation. The Simulink allows the user to define a set of user-configurable parameters using the *Simulink Mask Editor.* Each mask parameter is linked to a specific block parameter inside the masked subsystem, such that setting a mask parameter sets the associated block parameter. The system settings are grouped into six groups: *New Input Connection Request, Bearer Configuration, System Configuration, Unicast Traffic Configuration, Multicast User Configuration and Multicast Traffic Configuration.* The configuration settings allow the system to be analysed under different conditions. The actual algorithms are programmed using the S-Function user defined block provided as the in-built block in the Simulink.

Chapter 6 : SIMULATION SCENARIOS AND RESULTS

6.1 Introduction

This chapter describes the different scenarios which are simulated to verify the correct working of the CAC framework. The proposed CAC framework consists of three different functionalities as described in Section 4.2 namely; the CAC Processor, the Retune Controller and the Pre-emption Controller. In addition, the system also supports both unicast and multicast mode of traffic for which different admission decision processes are applied. Hence, in order to verify the correct working of the CAC framework as a whole, it is necessary to individually verify the working of the 3 major functionalities under different traffic scenarios. Table 6-1 summarizes the classification of the simulations with their objectives. Each of these simulations is described in detail in the sections to follow.

Simulation Classification		Objective
1	CAC Processor without retune and pre-emption capabilities	To verify the working of the CAC Processor
2	CAC Processor with only retune capability	To verify the working of the CAC Processor with the Retune Controller
3	CAC Processor with only pre-emption capability	To verify the working of the CAC Processor with the Pre-emption Controller
4	CAC Processor with both retune and pre-emption capabilities	To verify the working of the CAC Processor with the Retune Controller and the Pre-emption Controller

Table 6-1 Classification and Objectives of the Simulations

6.2 Simulation Parameters

The simulation parameters are defined to analyse the system performances under different scenarios. Table 6-2 lists the different simulation parameters used for simulations classified in Table 6-1.

<table>
<tr><th colspan="2">Simulation Classification</th><th>Simulation Parameters</th></tr>
<tr><td>1</td><td>CAC Processor without retune and pre-emption capabilities</td><td rowspan="2">• Blocking ratio
• Number of admitted connections
• User data throughput efficiency
• Channel utilization efficiency</td></tr>
<tr><td>2</td><td>CAC Processor with only retune capability</td></tr>
<tr><td>3</td><td>CAC Processor with only pre-emption capability</td><td rowspan="2">• Blocking ratio
• Dropping ratio
• Number of successful connections
• Pre-emptable bandwidth
• Computation time
• Revenue generated</td></tr>
<tr><td>4</td><td>CAC Processor with both retune and pre-emption capabilities</td></tr>
</table>

Table 6-2 List of simulation parameters for different simulations

The simulation parameters are defined as follows:

- Blocking ratio: This is calculated as the ratio of the number of connections rejected/blocked to the total number of connection request made.
- Number of admitted connections: It indicates the number of connections admitted into the system.
- User data throughput efficiency: The user data throughput efficiency in percentage is the number of data bits to the total available capacity. It

is calculated as the amount of data which is transferred from the link layer to the physical layer.

- Channel utilization efficiency: The channel utilization efficiency in percentage is the occupied capacity to the total available capacity of a channel or a forward subband. The occupied capacity is calculated as the amount of data bits and the parity bits of the FEC coding applied in the physical frame.
- Dropping ratio: This is calculated as the ratio of the number of connections dropped/pre-empted to the total number of connection request made.
- Number of successful connections: It is defined as the number of admitted connections which finish their service time such that the connections are not dropped in between their sessions.
- Pre-emptable data size: This indicates the total data size released by all the pre-empted connections over a period of simulation time.
- Computation time: This is used to measure the performance of the pre-emption algorithms. It indicates the time MATLAB takes to run an algorithm. For a given simulation, the amount of time to run the algorithm each time the algorithm is triggered in a given simulation time, is measured. The computation time for the given algorithm is the average of the measured times over a period of the simulation time.
- Revenue generation: This is also used to measure the performance of the pre-emption algorithms. A time-based charging mechanism has been applied to calculate the revenue. A tariff of *w* pence/sec is used for calculating the charge for the session the connection is active in

the system. The value of *w* depends on the type of the traffic and is proportional to the priority of the application. Hence, a higher priority connection generates higher revenues and a lower priority connection generates lower revenues.

6.3 CAC Processor without retune and pre-emption capability

This section describes the simulation scenarios to verify the working of the CAC Processor without the retune and the pre-emption capabilities. As the CAC Processor encloses the adaptive and the non-adaptive admission control algorithms along with discrete admission decision processes for the unicast and the multicast traffic, it is necessary to verify and compare all the important elements of the CAC Processor. In order to obtain the required objective, the following five major scenarios are considered based on the different traffic sources:

- Scenario 1: Single class of unicast traffic
- Scenario 2: Mixed classes of unicast traffic
- Scenario 3: Single class of multicast traffic
- Scenario 4: Mixed classes of multicast traffic
- Scenario 5: Mixed unicast and multicast traffic

Each scenario is further divided into different test cases. Table 6-3 summarizes the test scenarios and test cases, together with the test objectives.

Scenario		Test Cases	Test Objectives
1	Single class of unicast traffic	a) Effect of link condition b) Effect of class of MT c) Effect of traffic intensity	Compare the performance of the system using the adaptive and the non-adaptive admission control algorithms for: • different radio link conditions • different classes of MT • different traffic load conditions
2	Mixed classes of unicast traffic	a) Effect of Link Condition b) Effect of class of MT	Compare the performances of the system using the adaptive and the non-adaptive admission control algorithms for: • different radio link conditions • different classes of MT
3	Single class of multicast traffic	a) Effect of number of MTs b) Effect of class of MT	To verify the working of the admission decision process for multicast using single class of multicast traffic by varying the following: • number of MTs accessing multicast traffic • classes of MT
4	Mixed classes of multicast traffic	a) Effect of number of MTs b) Effect of class of MT	To verify the working of the admission decision process for multicast using mixed classes of multicast traffic by varying the following: • number of MTs accessing multicast traffic • classes of MT
5	Mixed unicast and multicast traffic	a) Effect of number of MTs b) Effect of link condition	To compare the working of the admission decision process for unicast and multicast using a mix of unicast and multicast traffic by varying the following: • number of MTs accessing unicast and multicast traffic • radio link condition

Table 6-3 Simulation scenarios and test cases for *CAC Processor without retune and pre-emption capabilities*

6.3.1 Results for Scenario 1: Single Class of Unicast Traffic

The objective of this scenario is to compare the performance of the adaptive and the non-adaptive admission control algorithm using a single class of unicast traffic which is assumed to be video streaming application. The two algorithms differs in the way the resources used on a given forward subband is calculated as explained in section 4.2.2.1.3. The performances are measured under different test cases and the results are compared.

Table 6-4 summarizes the MATLAB scenario configuration for the given scenario. With the set configuration, the video application generates 100 connection requests with traffic parameters; peak rate, source utilization and mean burst period set at 32 kbps, 0.8 and 0.1sec respectively. The average holding time is taken as 300 seconds and it is assumed that only one forward subband of the type, F80T4.5X8B is available for the given scenario.

Common Simulation Parameters	Values
Total no. of connections	100
Number of MTs	5
No. of forward subbands	1
Forward bearer type	F80T4.5X8B
Peak Rate (kbps)	32
Source Utilization	0.8
Mean Burst Period (sec)	0.1
Avg. Holding time (sec)[3]	300

Table 6-4 Scenario configuration for unicast video streaming application

[3] The average holding time mentioned throughout this chapter for different traffic types are taken from the reference [95].

6.3.1.1 Test Case 1: Effect of Link Condition

This test case compares the performance of the adaptive and the non-adaptive admission control algorithms under different link conditions. The system is run with different link conditions for MTs while keeping the same input connection queue. The performances are measured and the results are compared. The following three link conditions are considered:

- Low LQ – All MTs are assumed to be in low link condition.
- High LQ – All MTs are assumed to be high link condition.
- Random LQ – All MTs are assumed to have a random link condition between low and high.

The link quality of a given MT is indicated by C/No ratio. For the given scenario with F80T4.5X8B type of forward bearer, Table 6-5 shows the range of C/No ratios supported by the different type of the MT classes; Class1, Class2, and Class3.

MT Class / Link Condition	Carrier to Noise ratio (C/No) (dB)		
	1	2	3
Low LQ	62.5	61.5	57
High LQ	65.5	65	62
Random LQ	Random value between 62.5 and 65.5	Random value between 61.5 and 65	Random value between 57 and 62

Table 6-5 Range of C/No ratios supported by three MT classes for the F80T4.5X8B forward bearer type

For the given test case, it is assumed that all MTs are of the type Class1 and average inter-arrival time for the incoming connection requests is 5 seconds.

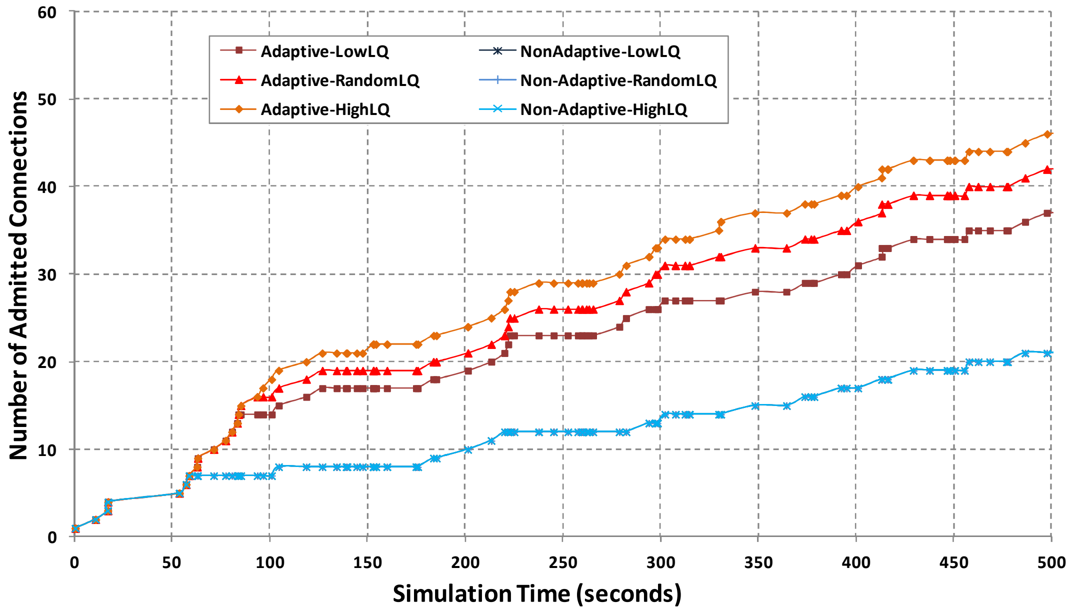

Figure 6-1 Effect of link condition on the number of admitted connections for single class of unicast traffic

Figure 6-1 shows the effect of the link quality on the number of admitted connections. For the adaptive algorithm, the number of connections admitted increases under high link quality condition as compared to the low link quality condition, whereas the number lies in between the high and the low for the random link quality condition. This is an expected behaviour since the FEC code rate and hence, the amount of user data that can be sent in a physical frame, increases as the link condition improves from low to high. This results in an increased number of connections that can be admitted into the system. On the other hand, for the non-adaptive algorithm, irrespective of the change of the link conditions, the number of admitted connections remains unchanged. This is due to the fact that the non-adaptive algorithm does not consider the change in the FEC code rate with the change in the link quality while calculating the resource used on a forward subband. It considers the lowest code rate supported for a given forward bearer type as mentioned in section 4.2.2.1.3. This results in a reduced amount of data that can be sent in a frame in the physical layer and hence reduces the number of connections that can be admitted.

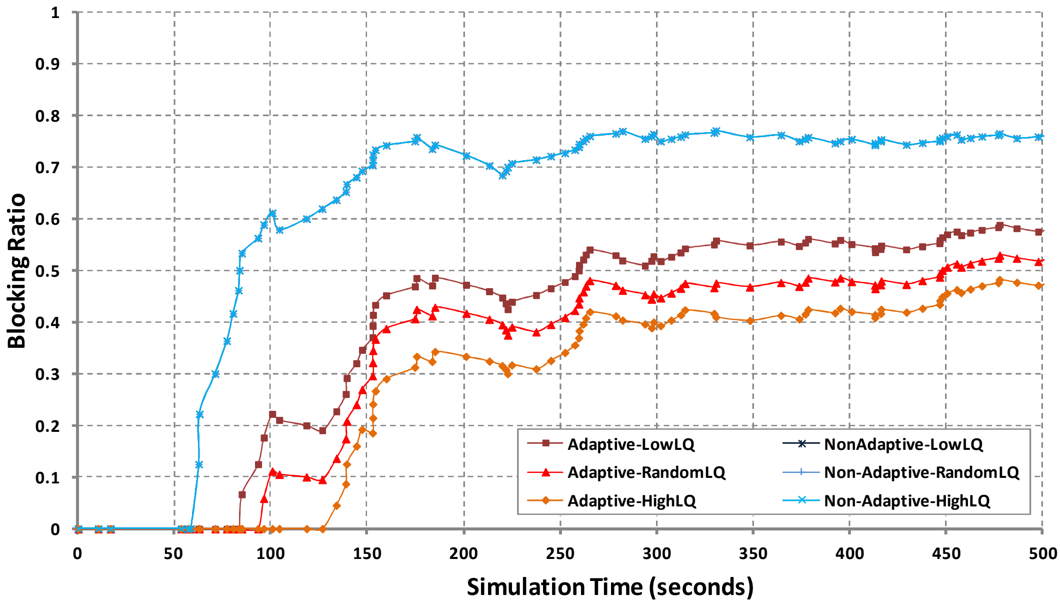

Figure 6-2 Effect of link condition on blocking ratio for single class of unicast traffic

Figure 6-2 shows the effect of the link quality on the blocking ratio. As can be seen, the blocking ratio increases as the link condition depreciate from high to low for the adaptive algorithm. This is directly related to the number of admitted connections shown in Figure 6-1. Under low link condition, fewer connections are admitted, more connections are blocked and hence, the blocking ratio is high. Again, for the non-adaptive algorithm the blocking ratio remains same irrespective of the change in the link conditions.

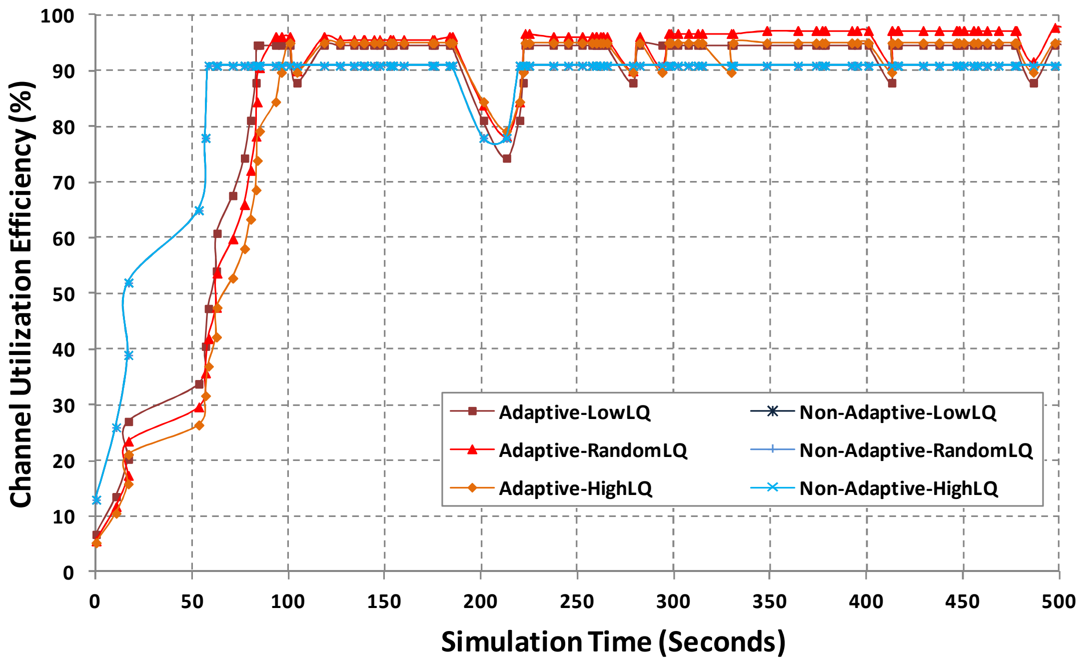

Figure 6-3 Effect of the link condition on the channel utilization efficiency for single class of unicast traffic

Figure 6-3 shows the effect of the link condition on the channel utilization efficiency. For the adaptive algorithm, before a steady state is reached around 100 sec of the simulation time, the channel utilization is slightly higher for the low link condition as compared to the high link condition and lies in between the high and low for the random link condition. This is due to the fact that the FEC code rate applied in the low link condition is lower than in the high link condition. Hence, an increased number of parity bits are required to send the same amount of data which results in an increase in the channel utilization. However, when the steady state is reached, the channel is fully occupied and hence, there is not much difference in the channel utilization with different link quality condition at the steady state. For the non-adaptive algorithm, there is no change in the channel utilization with the changing link quality condition. This is expected as the algorithm does not adapt with the change in the link condition as it works with a fixed value of the code rate. Also, before the steady state is reached, the channel utilization is higher than the adaptive algorithm. This is again due to the fact that the algorithm sends the data with a lower code rate which increases the number of parity bits to send the same amount of data and hence uses more resources. However, during the steady state, the channel utilization efficiency becomes constant at 91%. This is because each connection requires more than 10% of the channel capacity due to the lower code rate. This implies that after the channel utilization efficiency reaches 91%, no more connection can be admitted into the system and hence the utilization remains constant at 91%.

The frequent dips in the graph at the steady state are due to the connection departing from the system after finishing their service time. As the connection departs, the resources are released and the channel utilization drops. For example, at 100.4 seconds of the simulation time a new connection request arrives. This request is accepted into the highly congested channel as a result of some existing connections departing from the system after finishing their service time. The resources released by the departing connections are then assigned to the new connection. For the adaptive algorithm, two existing connections finish their service time while for the non-adaptive algorithm only one existing connection departs from the system. This is because the other connection was blocked earlier by the non-adaptive algorithm due to the lack of the resources. Hence, the channel utilization drops for the adaptive algorithm while it remains constant for the non-adaptive algorithm. The same explanation is provided for all other dips in this graph and all the similar graphs to follow in this scenario and the next scenario.

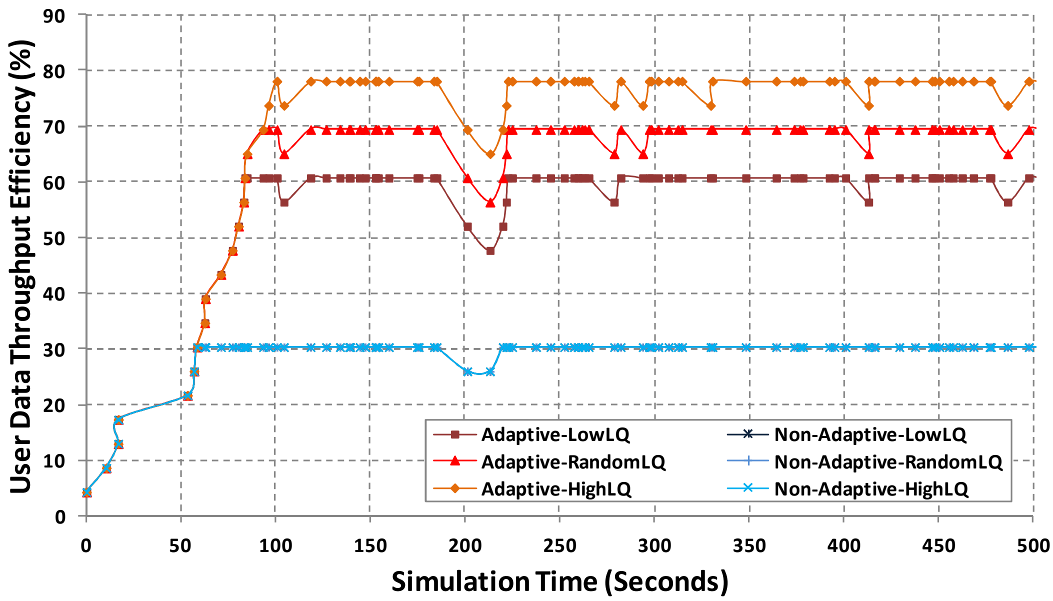

Figure 6-4 Effect of link condition on user data throughput efficiency for a single class of unicast traffic

Figure 6-4 shows the effect of the link condition on the user data throughput efficiency. Before the steady state is reached, all the graphs overlap each other. This is an expected result since the user data throughput, indicated by the amount of user data transferred from the link layer to the physical layer depends on the QoS parameter set for the video streaming application which remains the same for both the algorithms irrespective of the link conditions. However, when the steady state is reached, indicating that the channel is congested, it can be seen with the non-adaptive algorithm only 30% of the total channel capacity consists of the user data bits. This is due to a lower code rate used by the non-adaptive algorithm. Similarly, for the adaptive algorithm, the user data throughput efficiency increases with the increase in the link condition. Since a higher code rate is supported in high link condition, more data bits can be sent in the physical frame and fewer parity bits are required for error detection and correction during the transmission.

6.3.1.2 Test Case 2: Effect of class of MT

This test case shows the effect on the performance of the adaptive and the non-adaptive admission control algorithm using different classes of MT. The system is run with different class of MT while keeping the same input connection queue. The performances are measured and the results are compared. The following four conditions are considered:

- All MTs are of type Class1
- All MTs are of type Class2
- All MTs are of type Class3
- All MTs are randomly of type Class1, Class2, and Class3.

The range of code rates supported by each MT class varies for a given type of forward bearer. In the given simulation for the bearer type, F80T4.5X8B, the code rates supported by each type of MT class are shown in Table 6-6.

MT Class	Code Rates Supported	
	Min (Low Link Quality)	Max (High Link Quality)
Class1	0.642	0.822
Class2	0.575	0.775
Class3	0.334	0.642

Table 6-6 Code rates supported for each class of MT for F80T4.5X8B type of bearer

For the test case, it is assumed that all MTs are under low link condition and the average inter-arrival time for the incoming connection requests is 5 seconds.

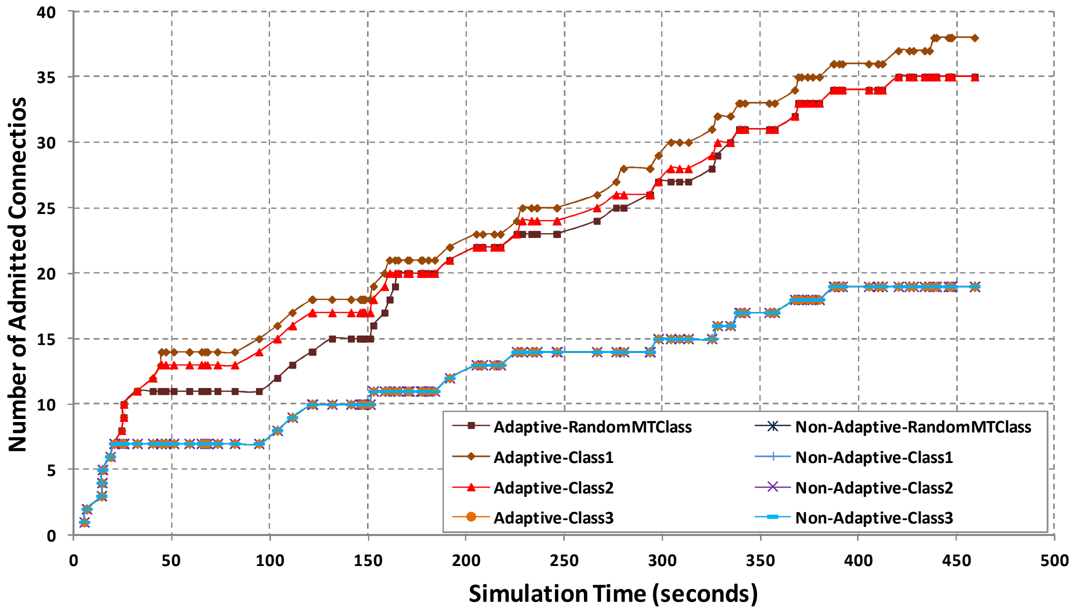

Figure 6-5 Effect on the number of admitted connections using different class of MT for single class of unicast traffic

Figure 6-5 shows the effect on the number of the admitted connections using different class of MT. For the adaptive algorithm, the number of connections admitted is highest for Class1 MTs which is then closely followed by Class2 MTs and then Class3 MTs. This is evident from the figures in Table 6-6 which shows that Class1 MTs supports the highest code rate followed by Class2 and Class3 MTs under the low link condition. Also, the code rate value supported by Class1 and Class2 are much higher and closer than Class3. As explained before, the higher values of code rate allows more data to be sent in the physical frame, and hence, increasing the number of connections admitted. On the other hand, for the non-adaptive algorithm, the number of admitted connections does not vary with the change in the class of the MT as it uses the lowest code rate supported by a given type of bearer. For the given simulation with F80T4.5X8B bearer type, the Class3 MTs also uses the lowest code rate supported by the bearer in low link condition and hence, the number of admitted connections for both the adaptive and the non-adaptive algorithms using only Class3 MTs overlap.

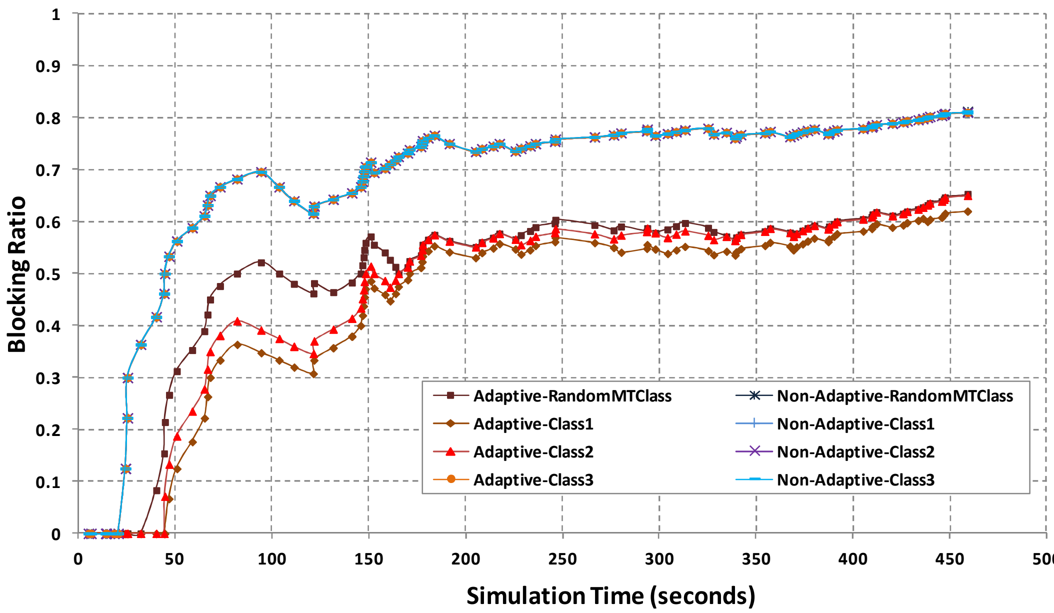

Figure 6-6 Effect on the blocking ratio using different class of MT for single class of unicast traffic

Figure 6-6 shows the effect on the blocking ratio using different class of MT. For the adaptive algorithm, the blocking ratio is the lowest for Class1 MTs which is then closely followed by Class2 MTs whereas; the blocking ratio for Class3 MTs is the highest. This is directly related to the number of admitted connections shown in Figure 6-5. Class1 MTs allows highest number of connection admissions such that few connections are blocked and hence the blocking ratio is low. Again, for the non-adaptive algorithm the blocking ratio remains same irrespective of the change in the class of MT.

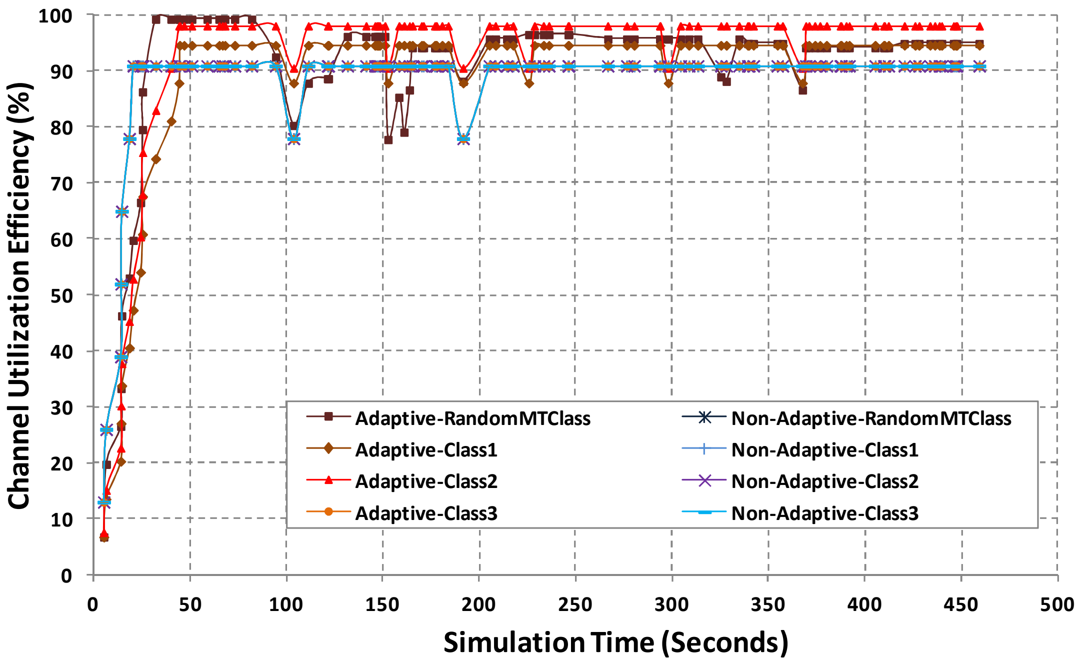

Figure 6-7 Effect on the channel utilization efficiency using different classes of MT for single class of unicast traffic

Figure 6-7 shows the effect on the channel utilization efficiency using different class of MT. For the adaptive algorithm, comparing the Class1 and Class2 MTs, the graphs closely follow each other. This is due to the fact that Class2 supports a slightly lower code rate compared to Class1. In the set configuration, each connection requires 7.5% of the channel capacity for Class2 MTs and 6.8% for Class1 MTs. Hence, in the steady state, the channel utilization reaches up to 98% for Class2 MTs and 94% for Class1 MTs. Although, the number of connections admitted for Class1 MTs is slightly higher than Class2 MTs at the steady state. Since the non-adaptive algorithm and the Class3 MTs in the adaptive algorithm support the lowest code rate, the same graph is followed in both conditions. Before the steady state is reached the channel utilization is the highest since the lowest code rate is considered and hence more parity bits are required to send the same data. However, when the steady state is reached, the channel utilization becomes constant at 91% of the total channel capacity which is explained earlier.

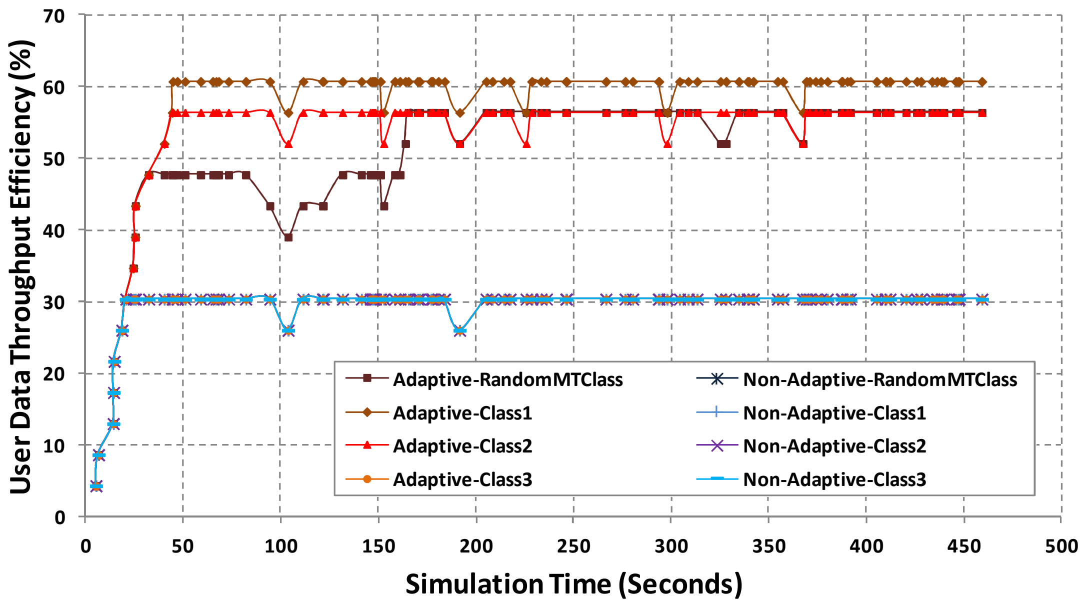

Figure 6-8 Effect on the user data throughput efficiency using different classes of MT for single class of unicast traffic

Figure 6-8 shows the effect on the user data throughput efficiency using different classes of MT. Before the steady state is reached, all the graphs overlap each other. This is an expected result since the amount of data transferred from the link layer to the physical layer depends on the QoS parameter set for the video streaming application which remains same for both the algorithms irrespective of the class of MT.

However, when the steady state is reached, indicating that the channel is congested, it can be seen with the non-adaptive algorithm only 30% of the total channel capacity consists of the user data bits. This is due to a lower code rate used by the non-adaptive algorithm. For the adaptive algorithm, the user data throughput efficiency is highest for Class1 MTs followed by Class2 MTs and Class3 MTs. Since a higher code rate is supported by Class1 MT, more data bits can be sent in the physical frame and fewer parity bits are required for error detection and correction during the transmission.

6.3.1.3 Test Case 3: Effect of Traffic load

This test case shows the effect of the traffic load on the performances of the adaptive and the non-adaptive admission control algorithm. The traffic load (ρ) is defined as the ratio of the average arrival rate ('λ') to the average service rate ('μ'). Therefore, $\rho=\lambda/\mu$. The system runs with different traffic loads by changing the average inter-arrival times for the video streaming traffic. Table 6-7 summarizes the different values of the traffic load values used for each simulation.

Simulations	Average HoldingTime,1/μ (Seconds)	Average InterarrivalTime,1/λ (Seconds)	Traffic Load, ρ = λ/μ
Sim1	300	100	3
Sim2	300	20	15
Sim3	300	10	30
Sim4	300	6	50
Sim5	300	5	60
Sim6	300	3	100

Table 6-7 Traffic load used for different simulation for single class of unicast traffic

For the test case, it is assumed that all MTs are under low link condition and of Class1 type of MT Class. The simulation is run for 300 seconds.

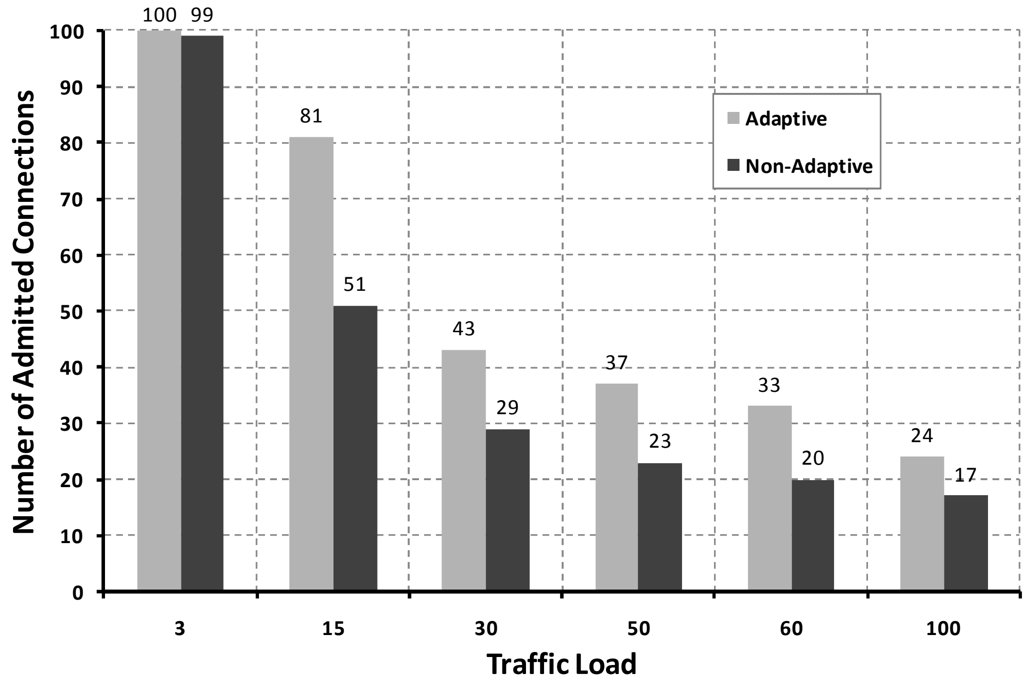

Figure 6-9 Effect of traffic load on the number of admitted connections for single class of unicast traffic

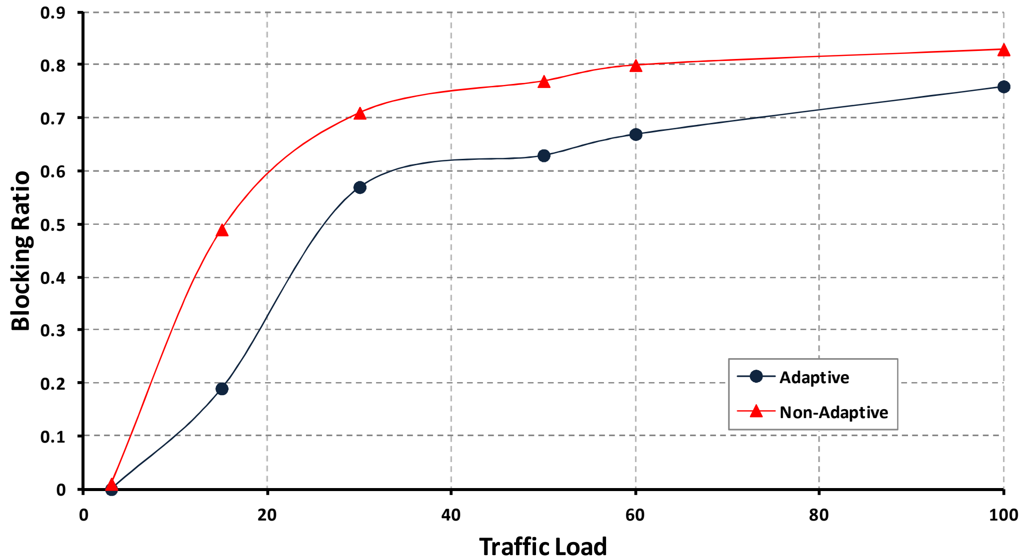

Figure 6-10 Effect of traffic load on blocking ratio for single class of unicast traffic

Figure 6-9 and Figure 6-10 compares the number of connections admissions and the blocking ratio using the adaptive and the non-adaptive algorithms under different traffic loads. As can be seen the number of admitted connections reduces for both the algorithms as the traffic load increases which is expected. With the increase in the traffic load, the system becomes congested since the frequency of the connection requests increases. However, the service time of the connections remains same in different traffic load, therefore the incoming connection requests are blocked as there is not enough channel capacity to be allocated to the new connection. In low traffic load condition, the existing connections continue to depart after finishing their service time which allows the new connection to be admitted. Comparing the adaptive algorithm with the non-adaptive algorithm, the former admits more connections under any traffic load as compared to the latter which has been explained in previous test cases.

6.3.2 Results for Scenario 2: Mixed classes of Unicast traffic

The objective of this scenario is to compare the performance of the adaptive and the non-adaptive admission control algorithm using mixed classes of unicast traffic. Although the scenario is similar to the above scenario with single class of unicast traffic, it is necessary to see the behaviour of the two algorithms, which differ in the way the resource is calculated on a given forward subband, under different traffic classes. Four types of unicast traffic are considered for this scenario; video streaming, netted voice, web browsing and email. The performances are measured under different test cases and the results are compared.

Table 6-8 summarizes the MATLAB scenario configuration for the given scenario. It is assumed that only one forward subband of the type F80T4.5X8B is available for the given scenario. Equal number of connections of each type of unicast traffic has been considered and the total number of connections is taken as 100.

Common Simulation Parameters		Values
Total No. of connections		100
Number of MTs		5
No. of forward subbands		1
Forward bearer type		F80T4.5X8B
Video Streaming	Number of connections	25
	QoS (kbps)	32
	Source Utilization	0.8
	Mean Burst Period	0.1
	Avg. Holding time (sec)	300
Netted Voice	Number of connections	25
	QoS (kbps)	60
	Source Utilization	0.6
	Mean Burst Period	0.01
	Avg. Holding time (sec)	240
Web Browsing	Number of connections	25
	QoS (kbps)	32
	Source Utilization	0.4
	Mean Burst Period	0.01
	Avg. Holding time (sec)	200
Email	Number of connections	25
	QoS (kbps)	120
	Source Utilization	0.2
	Mean Burst Period	0.01
	Avg. Holding time (sec)	150

Table 6-8 Scenario configuration for mixed classes of unicast traffic

6.3.2.1 Test Case 1: Effect of Link Condition

This test case shows the effect of the link condition on the performance of the adaptive and the non-adaptive admission control algorithm with a mix of unicast traffic. The conditions are similar to the conditions considered for single class of unicast traffic in section 6.3.1.1. Similar results are obtained for mixed classes of unicast traffic and compared against the results for the single class of unicast traffic.

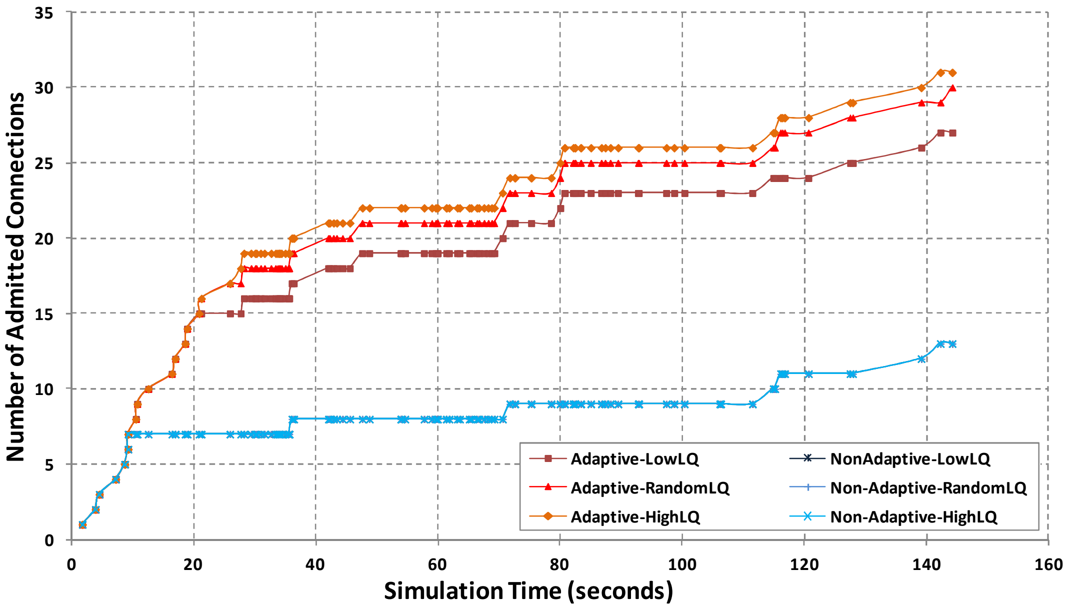

Figure 6-11 Effect of link condition on the number of admitted connections for mixed classes of unicast traffic

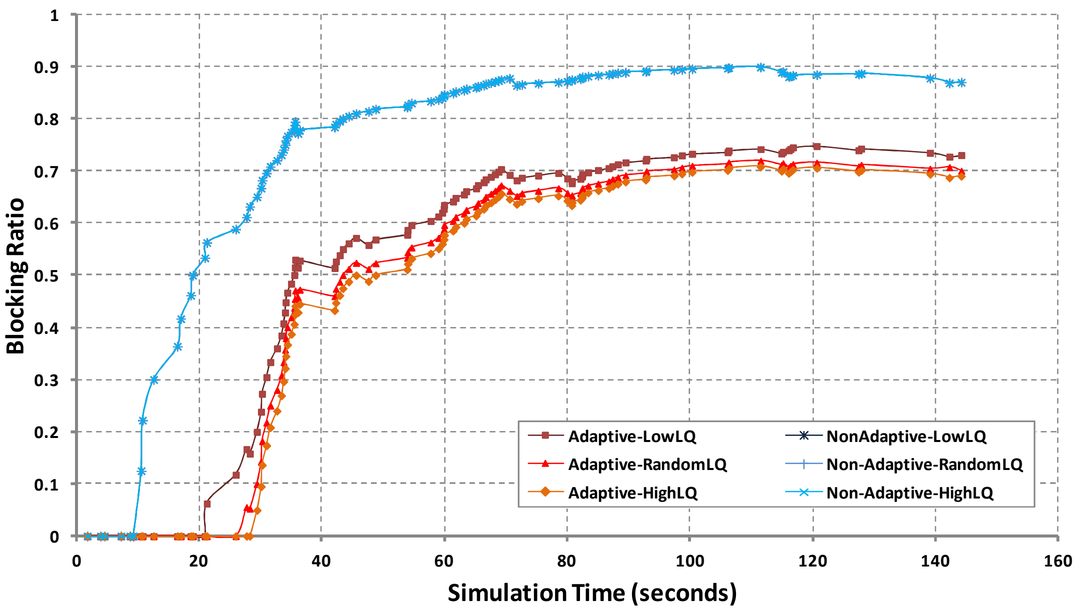

Figure 6-12 Effect of link condition on blocking ratio for mixed classes of unicast traffic

Figure 6-11 and Figure 6-12 shows the effect of the link condition on the number of admitted connections and the blocking ratio respectively. Comparing these results with single class of unicast traffic shown in Figure

6-1 and Figure 6-2 respectively, similarities can be seen. For the adaptive algorithm, the blocking ratio reduces under high link condition as compared to low link condition. Since higher code rates are supported in high link condition hence, more number of connections can be admitted in turn reducing the blocking ratio. However, for the non-adaptive algorithm, there is no effect on the blocking ratio of the change in link condition as expected.

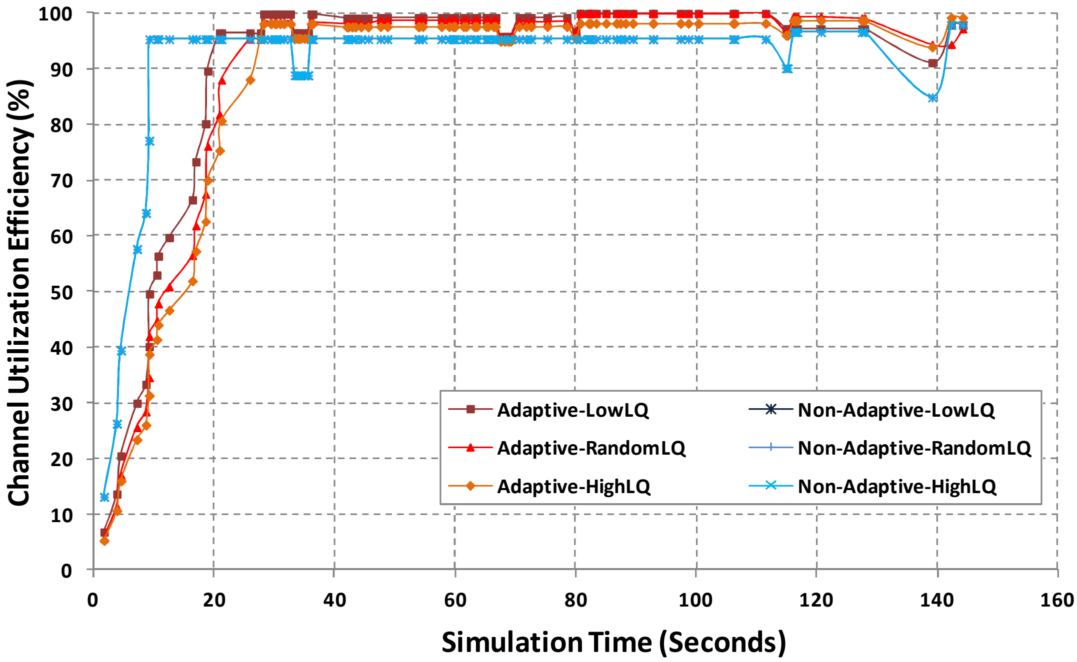

Figure 6-13 Effect of link condition on the channel utilization efficiency for mixed classes of unicast traffic

Figure 6-13 shows the effect of the link condition on the channel utilization efficiency. Comparing the result with the single class of unicast traffic shown in Figure 6-3, similarities can be seen. However, a slight increase in the channel utilization efficiency at the steady state with the mixed classes of traffic can be seen as compared to the single class traffic. This is due to the fact that in the presence of connections with different QoS requirements, some connections may require more resources than the others. For example, if the channel utilization efficiency is 91%, with single class traffic, if the QoS requirements are such that each connection utilizes more than 10% of the total channel capacity, no more connections can be admitted and the channel utilization become constant at 91%. However, with the mixed classes of unicast traffic, there may be the connections for a traffic type whose QoS requirement are such that it requires less than 9% of the total channel capacity. In such a case, the connection can be admitted and the channel utilization efficiency increases.

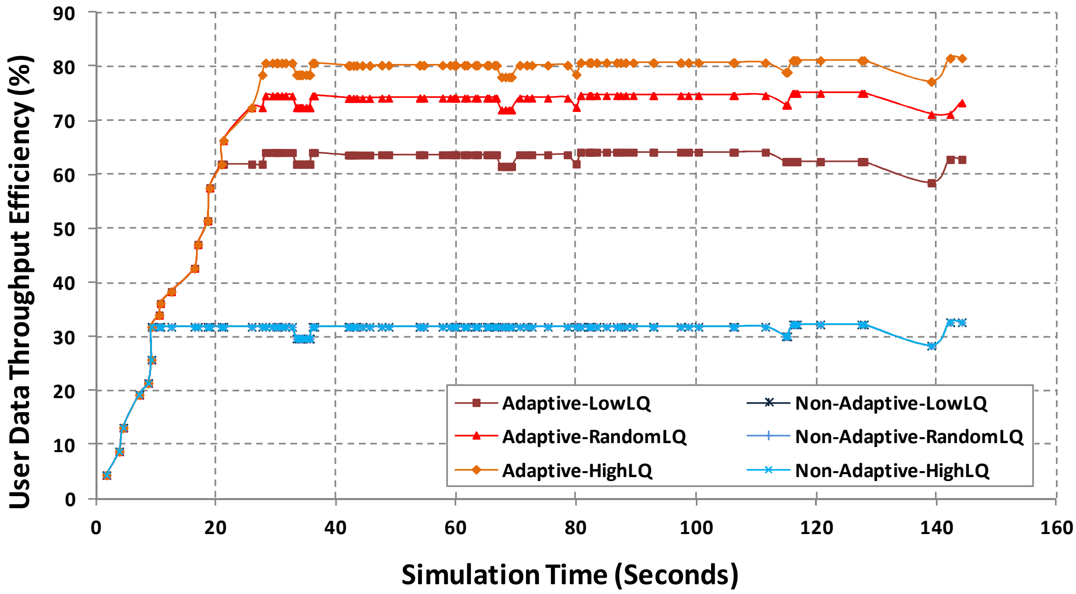

Figure 6-14 Effect of link condition on the user data throughput efficiency for mixed classes of unicast traffic

Figure 6-14 shows the effect of the link condition on the user data throughput efficiency. Comparing the result with single class of unicast traffic shown in Figure 6-4, similarities can be seen. However, there is a slight increase in the user data throughput efficiency at the steady state with the mixed classes of traffic. This is again due to the fact that in the presence of connections with different QoS requirements, some connections may require more resources than the others. Hence, the amount of user data transferred from the link layer to the physical layer is slightly higher for the mixed classes of traffic than the single class of traffic.

6.3.2.2 Test Case 2: Effect of class of MT

This test case shows the effect on the performance of the adaptive and the non-adaptive admission control algorithm with a mix of unicast traffic by using different classes of MT. The conditions are similar to the conditions considered for single class of unicast traffic in section 6.3.1.2. Similar results are obtained for mixed class of unicast traffic and compared against the results for the single class of unicast traffic.

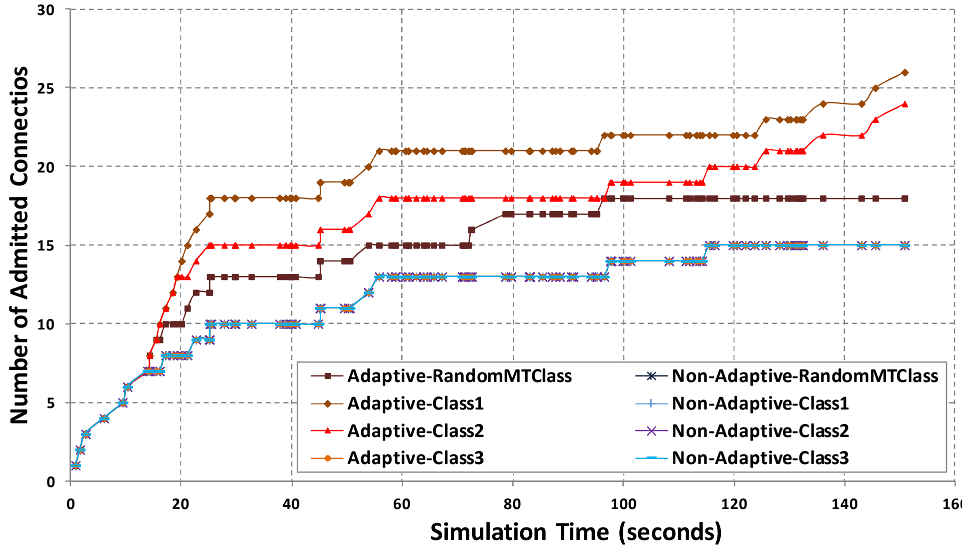

Figure 6-15 Effect on the number of admitted connections using different classes of MT for mixed classes of unicast traffic

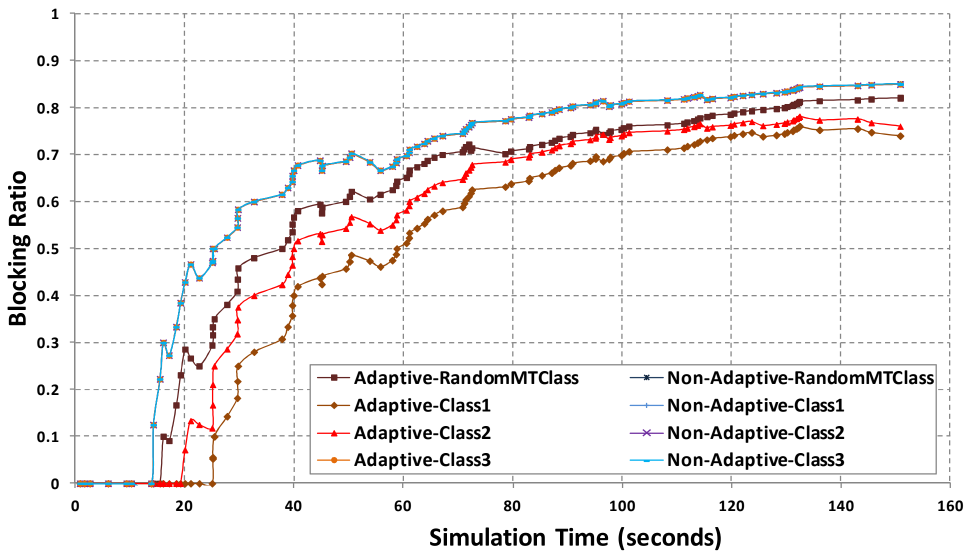

Figure 6-16 Effect on the blocking ratio using different classes of MT for mixed classes of unicast traffic

Figure 6-15 and Figure 6-16 shows the effect on the number of admitted connections and the blocking ratio respectively using different classes of MT. Comparing these results with single class of unicast traffic shown in Figure 6-5 and Figure 6-6 respectively, it can be seen that with mixed classes of traffic for both the algorithms, there is a slight increase in the number of connections admitted and a proportional decline in the blocking ratio before the steady state is reached. For example, for adaptive algorithm with Class1 MTs, with mix class of traffic, the number of admitted connections are about 18 whereas with single class of traffic, the value falls down to approximately 14. This can be explained from the fact that with a mix of unicast traffic with different QoS requirements, it is possible to admit more connections with varying resource consumption as compared to the condition where only a single class of traffic with fixed QoS is considered.

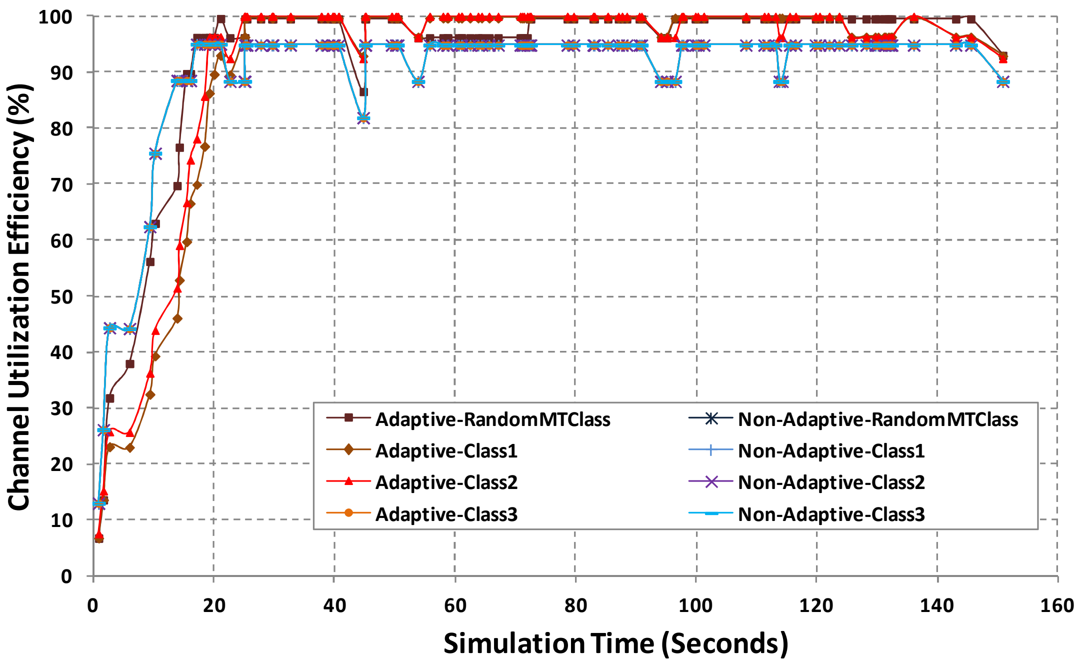

Figure 6-17 Effect on the channel utilization efficiency using different classes of MT for mixed classes of unicast traffic

Figure 6-17 shows the effect on the channel utilization efficiency using different classes of MT. As can be seen, with a mix of unicast traffic, the channel utilization is slightly higher than that for the single class of unicast traffic as shown in Figure 6-7 for both adaptive and non-adaptive algorithm. This is due to the fact that in the presence of connections with different QoS requirement the channel is utilized more effectively with some connections requiring more resources than the others.

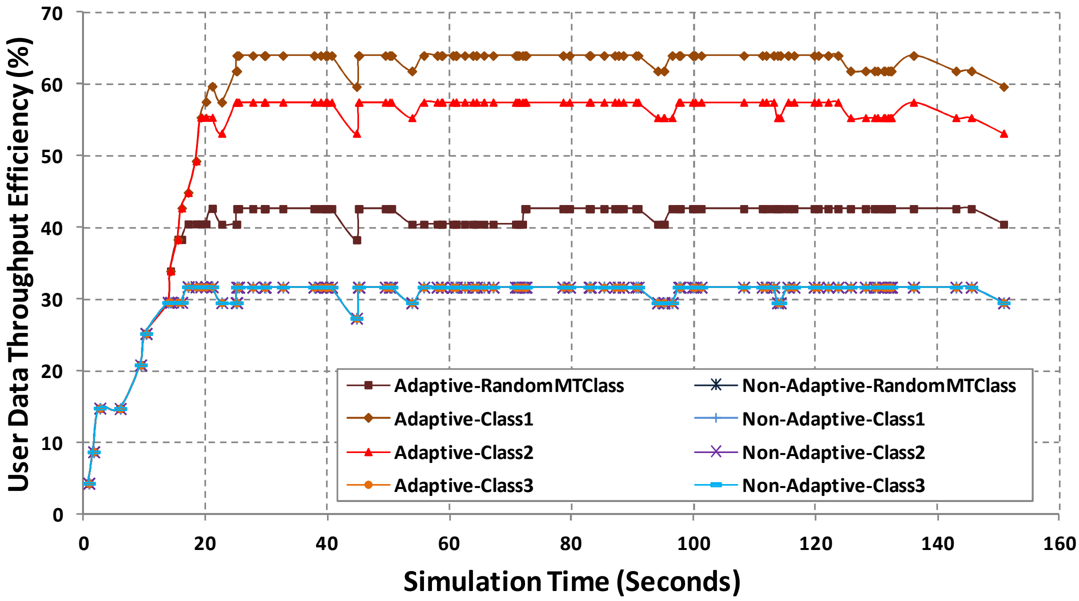

Figure 6-18 Effect on the user data throughput efficiency using different classes of MT for mixed classes of unicast traffic

Figure 6-18 shows the effect on the user data throughput efficiency using different classes of MT. As can be seen, with a mix of unicast traffic, the user data throughput efficiency is slightly higher than that for the single class of unicast traffic as shown in Figure 6-8. This is expected since with a mix of traffic with different QoS requirements, the channel is utilized more effectively with some connections requiring more resources than the others hence, more data bits can be sent in the physical frame.

6.3.3 Results for Scenario 3: Single class of Multicast traffic

The objective of this scenario is to analyse the working of the admission decision process using single class of multicast traffic which is assumed to be a video streaming application. The performances are measured under different test cases and the results are compared. Table 6-9 summarizes the MATLAB scenario configuration for the given scenario. With the set configuration, the multicast video application generates connection requests with traffic parameters; peak rate, source utilization and mean burst period set at 32 kbps, 0.8 and 0.1sec respectively. The average holding time is taken as 300 seconds and it is assumed that only one forward subband of the type, F80T4.5X8B is available for the given scenario. Also, the link condition for all the MTs is assumed to be low and the average inter-arrival time is taken as 5 seconds.

Common Simulation Parameters	Values
No. of forward subbands	1
Forward bearer type	F80T45X8B
Link condition	Low
Peak Rate (kbps)	32
Source Utilization	0.8
Mean Burst Period (sec)	0.1
Avg. Holding time (sec)	300
Avg. Inter-arrival time (sec)	5

Table 6-9 Scenario configuration for multicast video streaming application

6.3.3.1 Test Case 1: Effect of Number of MTs

In this test case, the effect of the number of MTs on the system performance is tested under the condition when all MTs access the video streaming multicast service. The system is run with different number of MTs and the following four conditions are considered:

- 5 MTs
- 10 MTs
- 15 MTs
- 20 MTs

For the test case, all MTs are assumed to be in low link condition and the class of MT is assumed to be Class1.

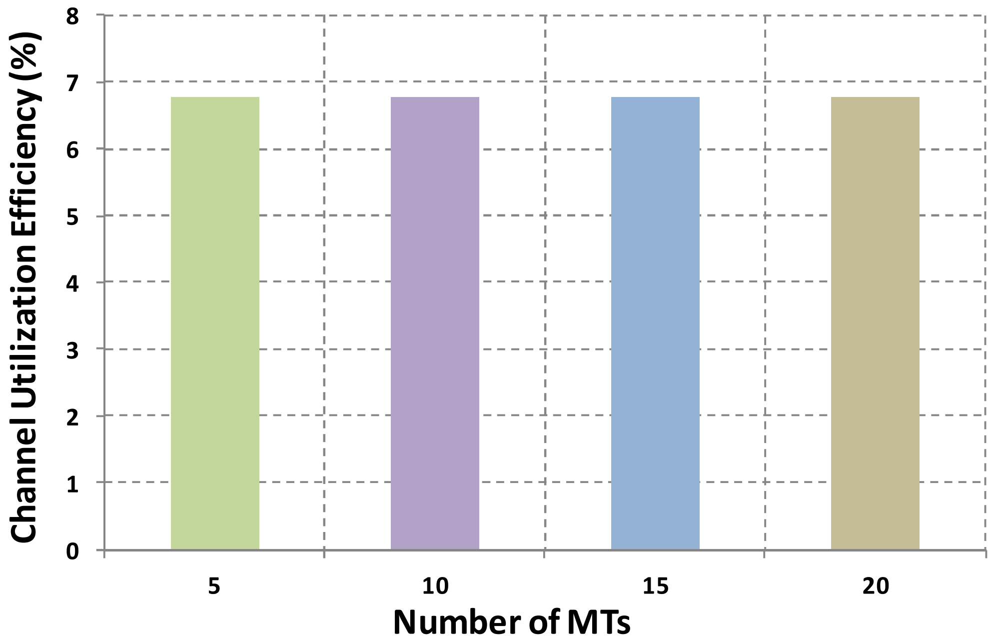

Figure 6-19 Effect of number of MTs on the channel utilization efficiency for single class of multicast traffic

Figure 6-19 shows the effect of the number of MTs on the channel utilization efficiency under the condition when all MTs access the video streaming multicast connection. As it can be seen, only 6.8% of the total channel is utilized irrespective of the number of MTs accessing the multicast service. This is evident from the fact that in the multicast mode of transmission, the resources are assigned to a multicast group on a given channel. Hence, the video streaming multicast connection is set up by the first MT and any subsequent MTs requesting the service, joins the existing group without having to assign separate resources. Hence, any number of MTs can join a group without affecting the channel utilization.

6.3.3.2 Test Case 2: Effect of Class of MT

In this test case, the effect of the class of MTs on the system performance is tested under the condition when all MTs access the video streaming multicast service. The system is run with different class of MT while keeping the same input connection queue. The performances are measured and the results are compared. The following four conditions are considered:

- All MTs are of type Class1
- All MTs are of type Class2
- All MTs are of type Class3
- All MTs are randomly of type Class1, Class2, and Class3.

For the test case, the system is run with 20 MTs

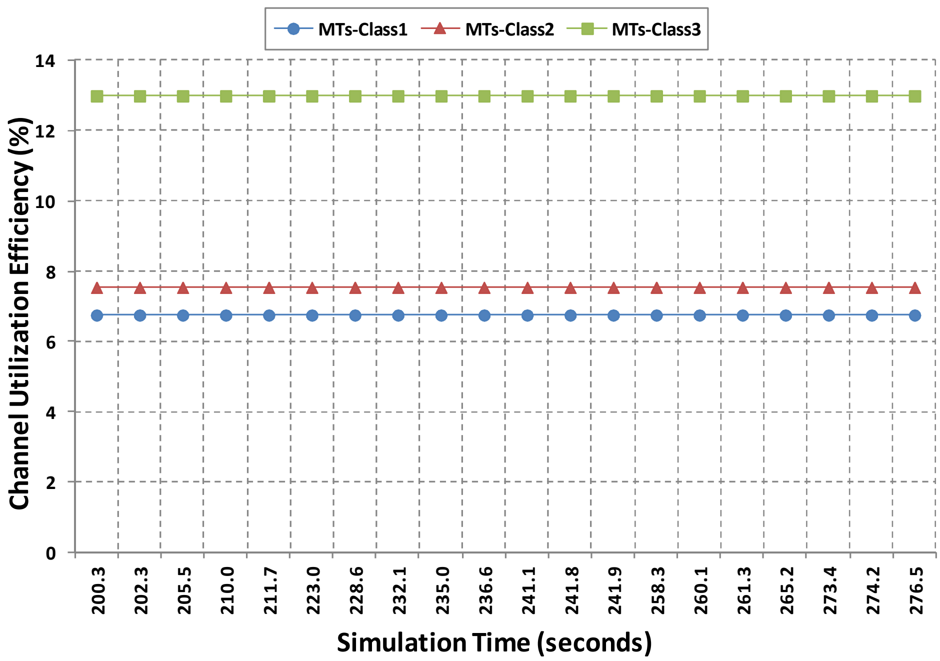

Figure 6-20 Comparison of the channel utilization efficiency using MTs of type Class1, Class2, and Class3 for single class of multicast traffic

Figure 6-20 compares the channel utilization efficiency under the conditions when the MTs are of type Class1, Class2 and Class3 accessing the video streaming multicast service. Different class of MT support different code rates for a given type of forward bearer. In the given simulation with F80T4.5XB forward bearer type, Class3 MTs supports very low code rate as compared to Class1 and Class2 MTs as shown in Table 6-6 and hence Class3 MTs require more resources to send the same amount of data. As it can be seen, Class3 MTs uses the maximum resources followed by Class2 and Class1.

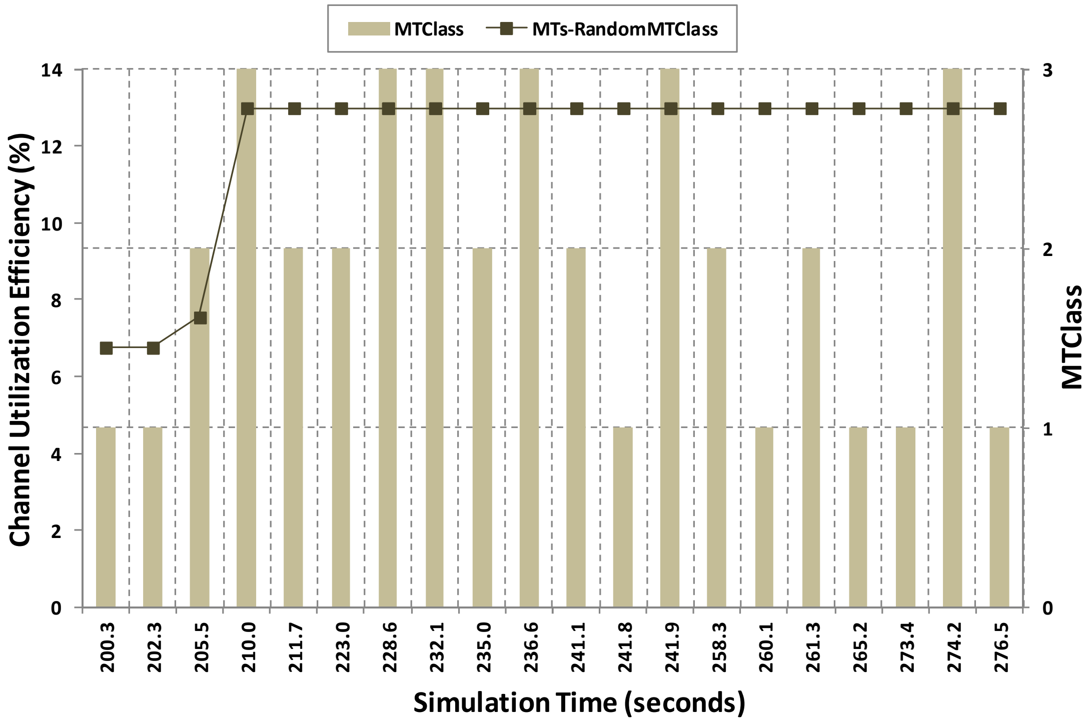

Figure 6-21 Comparison of the channel utilization efficiency using MTs randomly of type Class1, Class2, and Class3 for single class of multicast traffic

Figure 6-21 compares the channel utilization efficiency under the condition when MTs are randomly assigned one of three types of MT class. The secondary axis indicates the class of the MT. In a multicast group, the code rate selected for the data transmission is such that the lowest capable MT belonging to the group can also receive the data. As shown in Figure 6-20, Class3 MTs requires more resources to send the same amount of data. Hence, Class3 MTs are least capable MTs followed by Class2 and Class1 MTs. As seen in the graph, the channel utilization increases from 6.8% to 7.5% when a Class2 MT joins the existing multicast group consisting of only Class1 MTs. A further increase in channel utilization occurs when a Class3 MT joins the group. Once the lowest code rate supported by Class3 MT is set for the group, irrespective of the type of MTs joining the group, the channel utilization remains fixed at 13.2% as long as at least one MT of Class3 remains in the given multicast group.

6.3.4 Results for Scenario 4: Mixed classes of Multicast traffic

The objective of this scenario is to analyse the working of the admission decision process for mixed classes of multicast traffic; video streaming and file distribution with different QoS requirements. These are the two available multicast services that the users can access. The performances are measured under different test cases and the results are compared. Table 6-10 summarizes the MATLAB scenario configuration for the given scenario.

Common Simulation Parameters		Values
No. of subbands		1
Forward bearer type		F80T4.5X8B
Link condition		Low
Video Streaming	QoS (kbps)	32
	Source Utilization	0.8
	Mean Burst Period	0.1
	Avg. Holding time (sec)	300
	Avg. Inter-arrival time (sec)	5
	Traffic Class	5
File Distribution	QoS (kbps)	120
	Source Utilization	0.2
	Mean Burst Period	0.01
	Avg. Holding time (sec)	150
	Avg. Inter-arrival time (sec)	5
	Traffic Class	6

Table 6-10 Scenario configuration for mixed classes of multicast traffic

6.3.4.1 Test Case 1: Effect of Number of MTs

In this test case, the effect of the number of MTs on the system performance is tested under the condition when all MTs access two multicast services, video streaming and file distribution. The system is run with different number of MTs and the following four conditions are considered:

- 5 MTs
- 10 MTs
- 15 MTs
- 20 MTs

For the test case, all MTs are assumed to be in low link condition and the class of MT is assumed to be Class1.

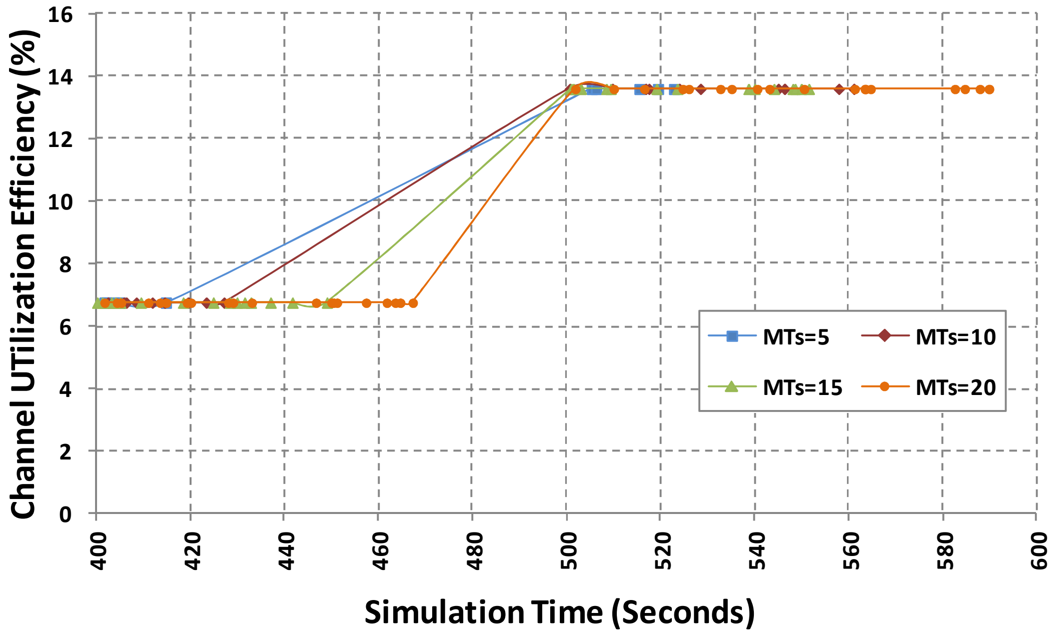

Figure 6-22 Effect of number of MTs on the channel utilization efficiency for mixed class of multicast traffic

Figure 6-22 shows the effect of number of MTs on the channel utilization efficiency when all MTs access two multicast services. At 400 seconds of the simulation time, the MTs start requesting the video streaming multicast service and at 500 seconds the file distribution multicast service is requested. As can be seen, the video streaming multicast connection utilizes 6.8% of the channel. The channel utilization rises to 14% when MTs start to access the file distribution multicast service. In the multicast mode of transmission, the resources are assigned to a multicast group on a given forward subband for a given multicast service. Since each MT is accessing both the multicast services, the channel utilization rises to 6.8% when an MT accesses the first multicast service and then to 14% when the MT accesses the second multicast service. Any subsequent MTs requesting to access any of the two services, just joins the existing groups without having to assign separate resources. Hence, any number of MTs can join a group without increasing the channel utilization efficiency.

6.3.4.2 Test Case 2: Effect of Class of MT

In this test case, the effect of the class of MTs on the system performance is tested under the condition when a mix of multicast service is accessed by the MTs. The system is run with different classes of MT while keeping the same input connection queue. The performances are measured and the results are compared. The following four conditions are considered:

- All MTs are of type Class1
- All MTs are of type Class2
- All MTs are of type Class3
- All MTs are randomly of type Class1, Class2, and Class3.

For the test case, the system is run with 20 MTs where first 10 MTs request the video streaming multicast service at 200 seconds of the simulation time and the remaining 10 MTs request the file distribution multicast service at 300 seconds of the simulation time.

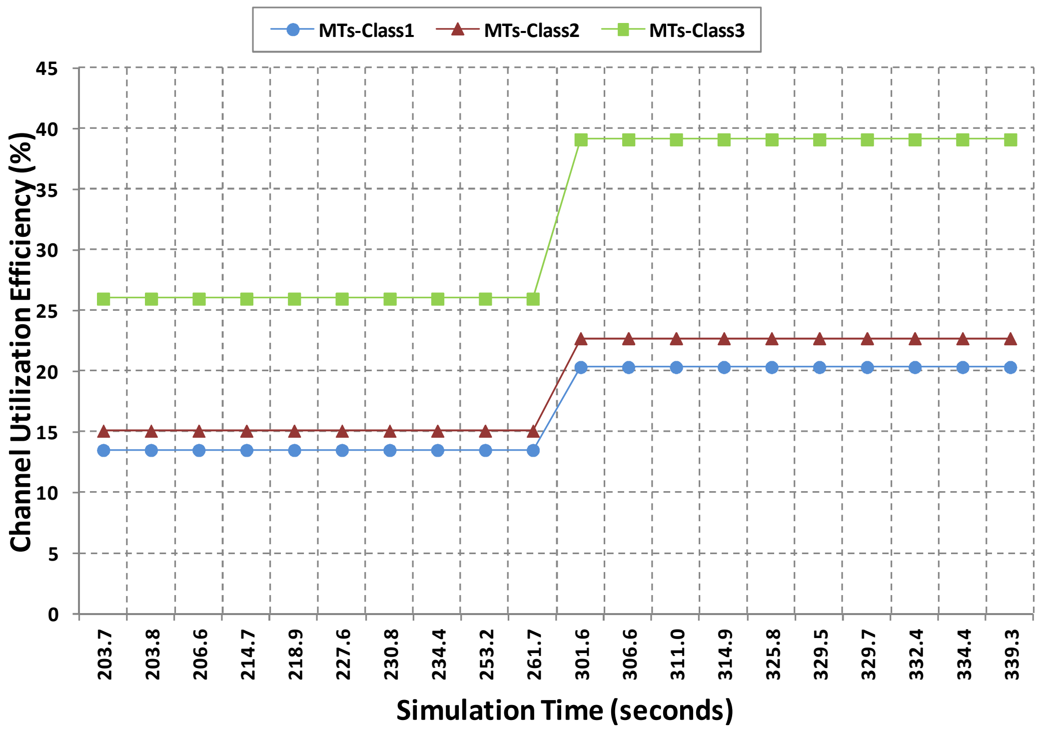

Figure 6-23 Comparison of the channel utilization efficiency using MTs of type Class1, Class2, and Class3 for mixed multicast traffic

Figure 6-23 compares the channel utilization efficiency under the conditions when the MTs are of type Class1, Class2 and Class3 accessing the video streaming and file distribution multicast service. Different classes of MT support different code rates for a given type of forward bearer. In the given simulation with F80T4.5XB forward bearer type, Class3 MTs supports very low code rate as compared to Class1 and Class2 MTs as shown in Table 6-6 and hence require more resources to send the same amount of data. As it can be seen, Class3 MTs uses the maximum resources followed by Class2 and Class1.

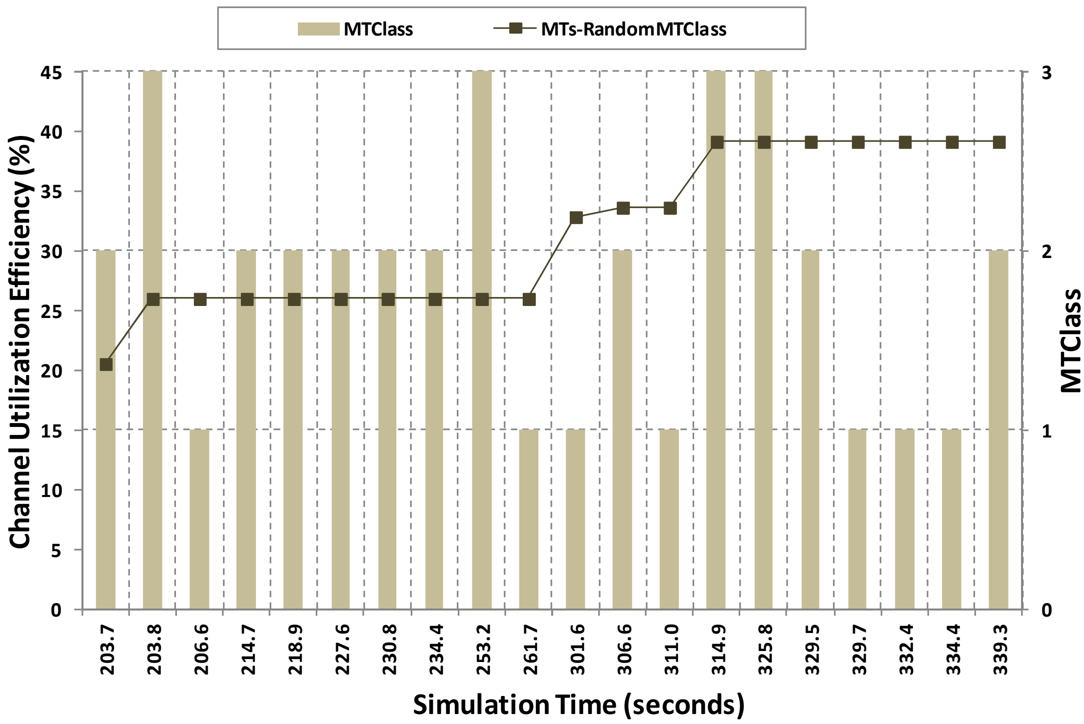

Figure 6-24 Comparison of the channel utilization efficiency using MTs randomly of type Class1, Class2, and Class3 for mixed multicast traffic

Figure 6-24 compares the channel utilization efficiency under the condition when MTs are randomly assigned one of three types of MT Class. The secondary axis indicates the class of the MT. The first 10 MTs accessing the video streaming multicast connection belongs to one multicast group and the remaining 10 MTs accessing the file distribution multicast connection belongs to another multicast group. In a multicast group, the code rate selected for the data transmission is such that the lowest capable MT belonging to the group can also receive the data. As seen in the graph, for the first 10 users accessing the video streaming multicast service the resource utilization increases from 21% to 27% when a Class3 MT joins the existing multicast group initiated by Class2 MT. Since Class3 MT supports the lowest code rate, the resource utilization remains constant at 27% irrespective of the different MT Classes present in the video streaming multicast group. After 300 seconds of the simulation time, the remaining 10 MTs request the file distribution multicast service. The channel utilization efficiency rises from 27% to 32% as the new multicast connection is established by a Class1 MT. The channel utilization efficiency increases when a Class2 MT joins the multicast group and further increases when a Class3 MT joins the group. Once the lowest code rate supported by Class3 MT is set for the group, irrespective of the class of MTs joining the group, the resource utilization remains fixed at 40% as long as there is at least one MT of Class3 in the given multicast group.

6.3.5 Results for Scenario 5: Mixed Unicast and Multicast traffic

The objective of this scenario is to compare the working of the admission decision process for a mix of unicast and multicast traffic. The video streaming application is used for both unicast and multicast.

Common Simulation Parameters		Values
No. of available forward subbands		1
Forward bearer type		F80T4.5X8B
Unicast Video Streaming	QoS (kbps)	32
	Source Utilization	0.8
	Mean Burst Period	0.1
	Avg. Holding time (sec)	300
	Avg. Inter-arrival time (sec)	5
Multicast Video Streaming	QoS (kbps)	32
	Source Utilization	0.8
	Mean Burst Period	0.1
	Avg. Holding time (sec)	300
	Avg. Inter-arrival time (sec)	5

Table 6-11 Scenario configuration for mixed unicast and multicast traffic

Table 6-11 summarizes the MATLAB scenario configuration for the given scenario. The performances are measured under different test cases and the results are compared.

6.3.5.1 Test Case 1: Effect of number of MTs

In this test case, the effect of the number of MTs on the system performance using unicast and multicast mode of transmission is compared. The system is run with different number of MTs where each MT has one unicast connection and one multicast connection. Hence, the total numbers of connections are double the number of MTs present in the system. The following two conditions are considered:

- 5 MTs – This condition represents the system in a non-congested state as the total number of connections is 10
- 20 MTs – This condition represents the system in a congested state as the total numbers of connections is 40

In both the conditions, all MTs first access the unicast video streaming application and after 100 seconds of the simulation time, all MTs access the multicast video streaming application. For the test case, all MTs are assumed to be in low link condition.

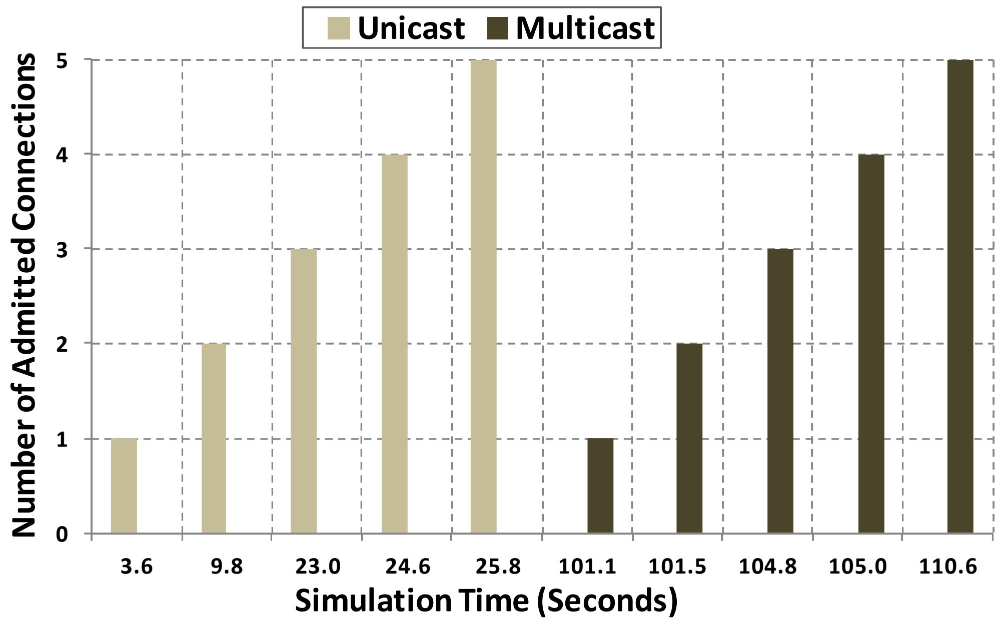

Figure 6-25 Comparison of the number of admitted connections using unicast and multicast traffic for 5 MTs

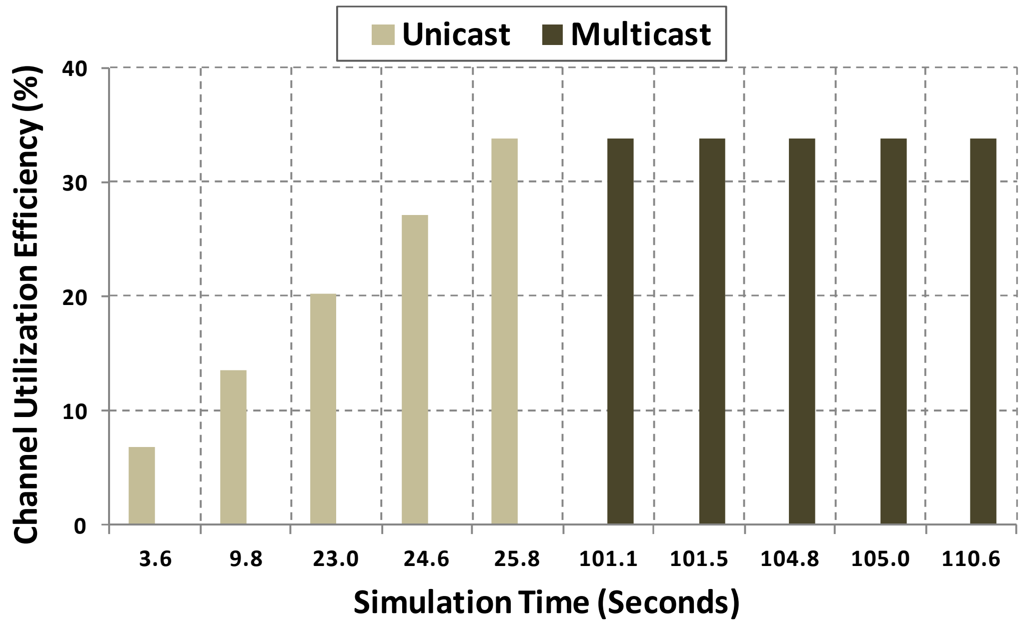

Figure 6-26 Comparison of the channel utilization efficiency using unicast and multicast traffic for 5MTs

Figure 6-25 and Figure 6-26 considers the system with 5MTs. As can be seen in Figure 6-25, all the unicast and multicast connections are accepted as the system is lightly loaded. Figure 6-26 shows a constant increase in the channel utilization with the arrival of each unicast connection. In the simulation, before the first multicast connection arrives at 101.1 sec, one unicast connection departs from the system and hence, the channel utilization efficiency remains same even though a new multicast connection has been created. All the subsequent multicast connections requests are admitted into the system by joining the existing multicast connection and hence there is no further increase in the channel utilization efficiency with the increase in the multicast connections.

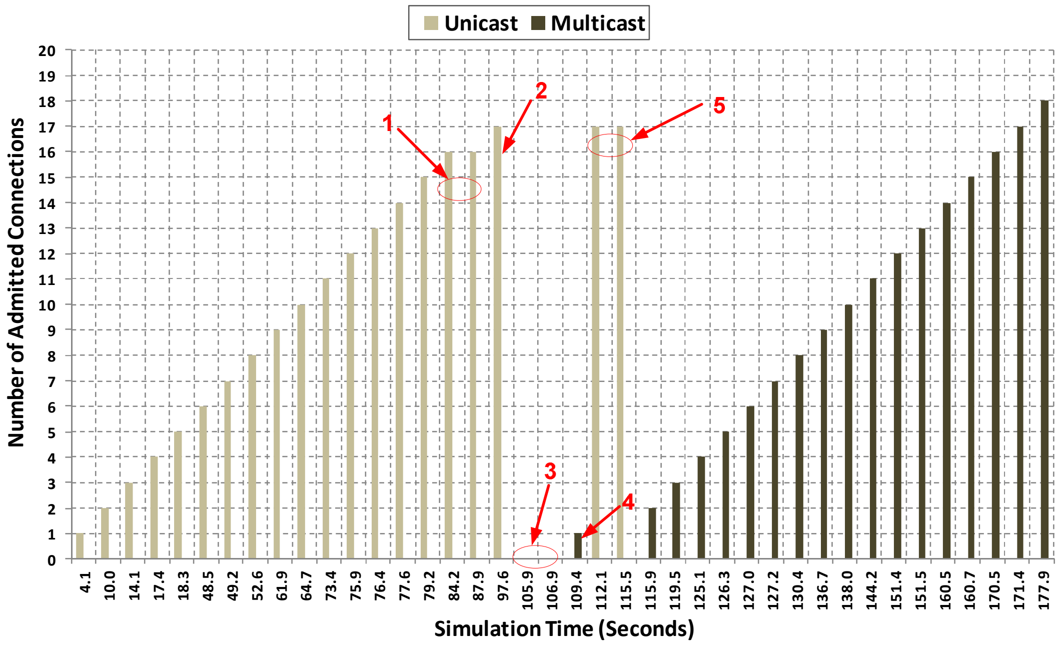

Figure 6-27 Comparison of the number of admitted connections using unicast and multicast traffic for 20 MTs

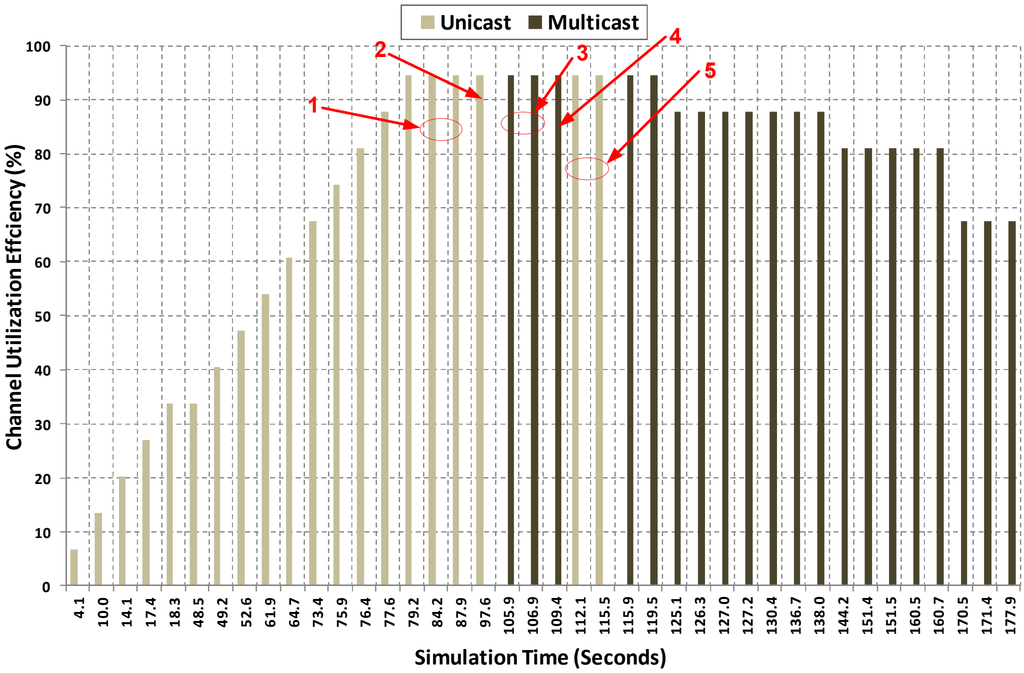

Figure 6-28 Comparison of the channel utilization efficiency using unicast and multicast traffic for 20 MTs

Figure 6-27 and Figure 6-28 considers the system with 20MTs. As can be seen 5 points have been marked on the figures which are explained as follows:

1 As the unicast connections are admitted into the system, the channel utilization efficiency increases until it reaches 94% at 79.2 seconds. The system is heavily loaded at this time and no more unicast connections can be admitted. Hence, the next two unicast connection requests at 84.2 and 87.9 seconds are blocked.

2 At 97.6 seconds, an existing unicast connection departs from the system and a new unicast connection request is accepted. Hence, there is an increase in the number of admitted connections; however, the channel utilization efficiency remains same.

3 At 105.9 seconds, the first multicast connection arrives. However, as there are not enough resources to establish the new multicast connection, the connection is blocked. Similarly, at 106.9 seconds,

another multicast connection arrives which is also blocked. Therefore, the number of multicast connection admitted is 0 and the channel utilization efficiency remains at 94%.

4 At 109.4 seconds, an existing unicast connection departs from the system and a new multicast connection request is accepted. However, the channel utilization efficiency remains same.

5 At 112.1 and 115.5 seconds, the last two unicast connections requests are blocked as there are not enough resources on the subband. Hence the total number of admitted unicast connections is 17. For the multicast traffic, since the multicast connection has been established, any subsequent multicast connection request joins the established multicast connection without using the additional resources. Therefore, the total number of admitted multicast connections is 18. As can be seen, the channel utilization efficiency falls as the simulation time increases. This is due to the existing unicast connections departing from the system.

6.3.5.2 Test Case 2: Effect of link condition

In this test case, the effect of the link condition on the system performance using unicast and multicast mode of transmission is compared. The system is run with 20 MTs where each MT has one unicast connection and one multicast connection. The MTs first access the unicast video streaming application and after 500 seconds of the simulation time, the MTs access the multicast video streaming application.

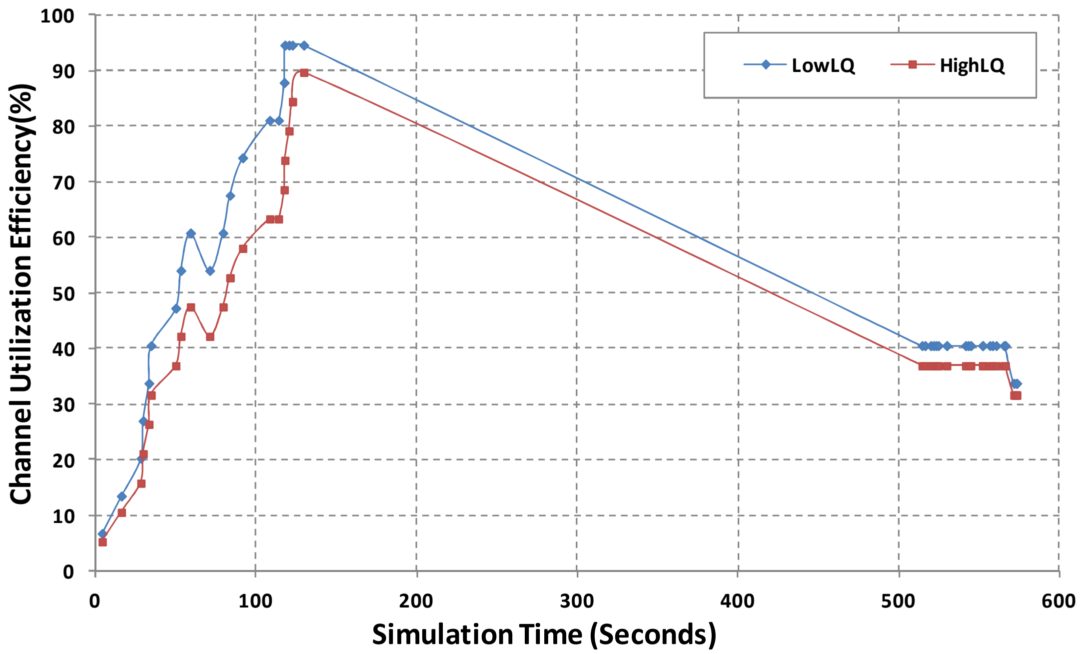

Figure 6-29 Comparison of the channel utilization efficiency using unicast and multicast traffic under different link condition

Figure 6-29 compares the channel utilization efficiency using the unicast and the multicast traffic under different link condition. As can be seen, in the first 150 sec of the simulation time, the unicast connections are set up. The channel utilized in the low link condition is higher than that in the high link condition. This is due to the fact that the low link condition employs a lower FEC code rate. Hence, an increased number of parity bits are required to send the same amount of data which results in an increase in the channel utilization. After 500 secs of the simulation time, the channel utilization drops as some of the unicast connections departs from the system. The multicast connection is then established by one MT which is joined by the subsequent MTs and hence, the channel utilization remains constant.

6.4 CAC Processor with only retune capability

The retune process is triggered by the CAC Processor when it is required to retune an MT from one forward subband to another subband as described in Section 4.2.3. This section describes the simulation scenarios to verify the working of the Retune Controller. The retune process varies slightly for the unicast and the multicast traffic. Hence, in order to verify the correct working of the Retune Controller, the following two scenarios are considered based on the traffic sources:

- Scenario 1: Mixed classes of unicast traffic
- Scenario 2: Mixed classes of multicast traffic

For each scenario, it is necessary to test different elements affecting the retune process. For example, for the unicast traffic the following elements are tested:

- The Subband Selection method: The effect of the selection of the forward subbands using *MinConnSubSel* and *Random* as described in section 4.2.2.1.1 on the retune process.
- The number of MTs: The effect of the number of MTs on the retune process while keeping the number of connections same.
- The number of available forward subbands: The effect of the number of available subbands on the retune process.

Based on the above list, each scenario is further divided into different test cases.

<table>
<tr><th colspan="2">Scenario</th><th>Test Cases</th><th>Test Objectives</th></tr>
<tr><td rowspan="3">1</td><td rowspan="3">Mixed classes of unicast traffic</td><td>a) Effect of subband selectors</td><td>• To compare the two types of subband selectors.
• To compare the system with retune and without retune procedure using two types of subband selectors.</td></tr>
<tr><td>b) Effect of number of MTs</td><td>• To compare the effect on the retune process by varying the number of MTs in the system.</td></tr>
<tr><td>c) Effect of number of available forward subbands</td><td>• To compare the effect on the retune process by varying the number of available forward subbands in the system.
• To compare the system with retune and without retune procedure by varying the number of available forward subbands.</td></tr>
<tr><td>2</td><td>Mixed classes of multicast traffic</td><td colspan="2">• To verify the retune procedure for multicast traffic.</td></tr>
</table>

Table 6-12 Simulation scenarios and test cases for *CAC Processor with only retune capability*

Table 6-12 summarizes the test scenarios and test cases, together with test objectives.

6.4.1 Results for Scenario 1: Mixed Classes of Unicast Traffic

The objective of this scenario is to verify the working of the retune process for mixed classes of unicast traffic. Table 6-13 summarizes the MATLAB scenario configuration for the given scenario. Four types of unicast traffic are considered; the video streaming, the netted voice, the web browsing and the email, where each traffic type generates 30 connection requests. The MTs are randomly assigned the MT class of type Class1, Class2, and Class3. The link condition for all MTs is assumed to be low.

Common Simulation Parameters		Values
MT Class		1,2,3
Link condition		Low
Forward bearer type		F80T4.5X8B
Video Streaming	Number of connections	30
	QoS (kbps)	32
	Source Utilization	0.8
	Mean Burst Period	0.1
	Avg. Holding time (sec)	300
	Avg.Inter-arrival time (sec)	5
Netted Voice	Number of connections	30
	QoS (kbps)	60
	Source Utilization	0.6
	Mean Burst Period	0.01
	Avg. Holding time (sec)	240
	Avg. Inter-arrival time (sec)	5
Web Browsing	Number of connections	30
	QoS (kbps)	32
	Source Utilization	0.4
	Mean Burst Period	0.01
	Avg. Holding time (sec)	200
	Avg. Inter-arrival time (sec)	5
Email	Number of connections	30
	QoS (kbps)	120
	Source Utilization	0.2
	Mean Burst Period	0.01
	Avg. Holding time (sec)	150
	Avg. Inter-arrival time (sec)	5

Table 6-13 Scenario configuration for mixed unicast traffic – *CAC Processor with only retune capability*

6.4.1.1 Test Case 1: Effect of subband selectors

The following two types of subband selection methods have been proposed in the section 4.2.2.1.1:

- *MinConnSubSel* – This method selects the forward subband with the minimum number of connections running. Such selection aims to evenly spread the traffic on the available subbands and hence, performing a sort of load balancing between the subbands.
- *Random* – This method randomly selects a forward subband from the list of available subbands.

In this test case, the effect of the two types of the subband selection methods on the system performance and on the retune process is tested. The system is run with proposed subband selection methods while keeping the same input connection queue. The performances are measured and the results are compared. For the test case, two forward subbands of the type, F80T4.5X8B, are considered and the number of MTs is taken as 20.

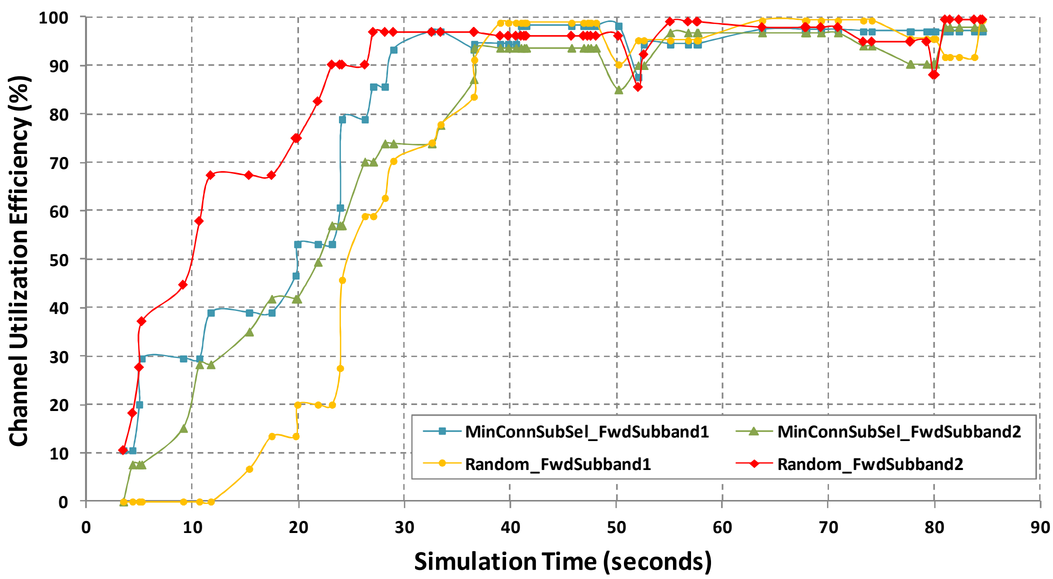

Figure 6-30 Effect of the type of the subband selection method on the channel utilization efficiency

Figure 6-30 shows the effect of the type of the subband selection method on the channel utilization efficiency of the available two forward subbands. As can be seen before the steady state is reached around 36 seconds of the simulation time, there is an even distribution of the resources on the two subbands using *MinConnSubSel* method as compared to *Random* subband selection method. This is expected since *MinConnSubSel* method performs load balancing on the available subbands by selecting the subband with minimum number of connections.

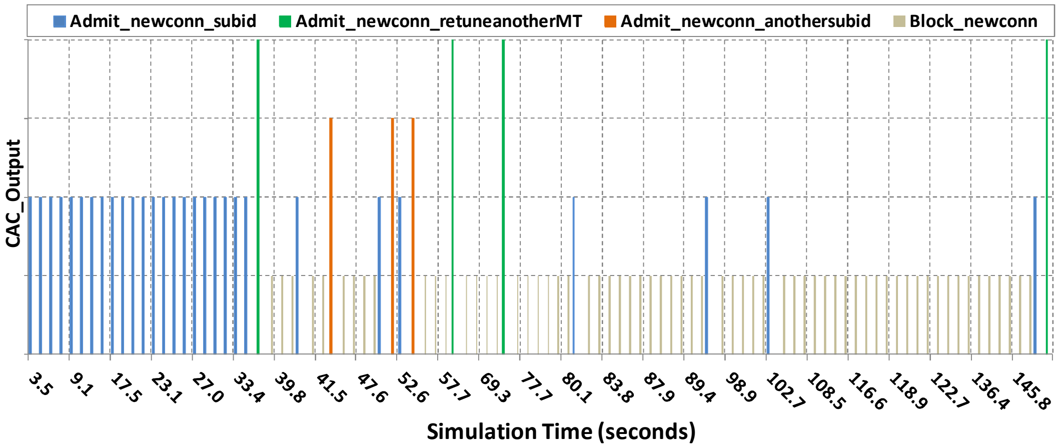

Figure 6-31 Effect of *MinConnSubSel* subband selection method on the number of retunes

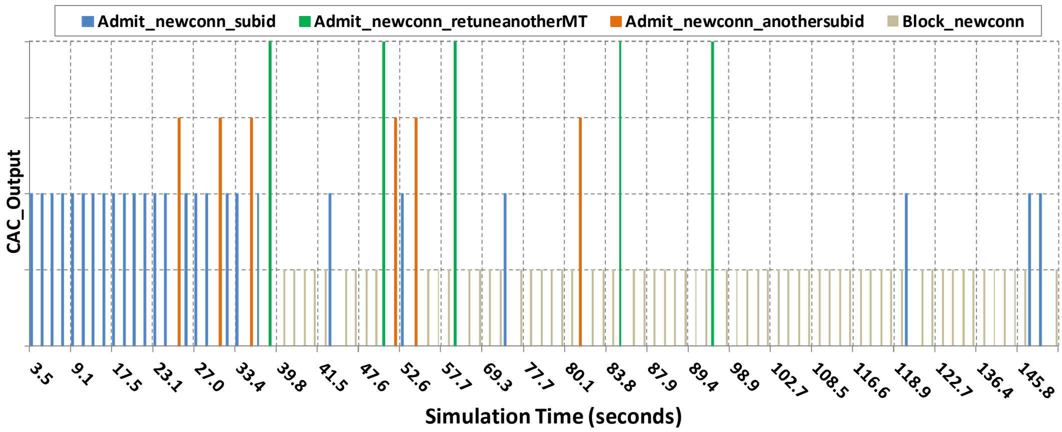

Figure 6-32 Effect of *Random* subband selection method on the number of retunes

The effect of the *MinConnSubSel* and the *Random* selection method on the number of retunes is shown in Figure 6-31 and Figure 6-32 respectively. The vertical axis is represented by the *CAC_Output* which indicates the action taken by the Admission Decision Controller in the CAC Processor as described in section 4.2.2.1.4 when a new connection request arrives. In the given test case where the pre-emption capability is disabled and only unicast traffic is considered, the possible actions indicated by *CAC_Output* are as follows:

- *Admit_newconn_subid* – Indicates that there are enough resources on the given forward subband to admit the new connection.
- *Admit_newconn_anothersubid* – Indicates that the resources are not enough on the given forward subband and hence, the MT of the new connection is retuned to another subband.
- *Admit_newconn_retuneanotherMT* – Indicates that the MT of the new connection cannot be retuned to another subband and hence, another MT on the given forward subband is retuned to another subband to admit the new connection on the given subband.
- *Block_newconn* – Indicates all above possibilities cannot be achieved and therefore, the new connection is blocked.

As can be seen, with the *Random* subband selection method there are in total eleven retunes required as compared to *MinConnSubSel* method which requires only seven retunes. This is expected as the forward subbands are randomly selected in *Random* method which may cause one subband to be densely populated than the other causing unnecessary retunes. In comparison, the *MinConnSubSel* selection method provides a basic form of load balancing amongst the available subbands and hence does not result in any unnecessary retunes.

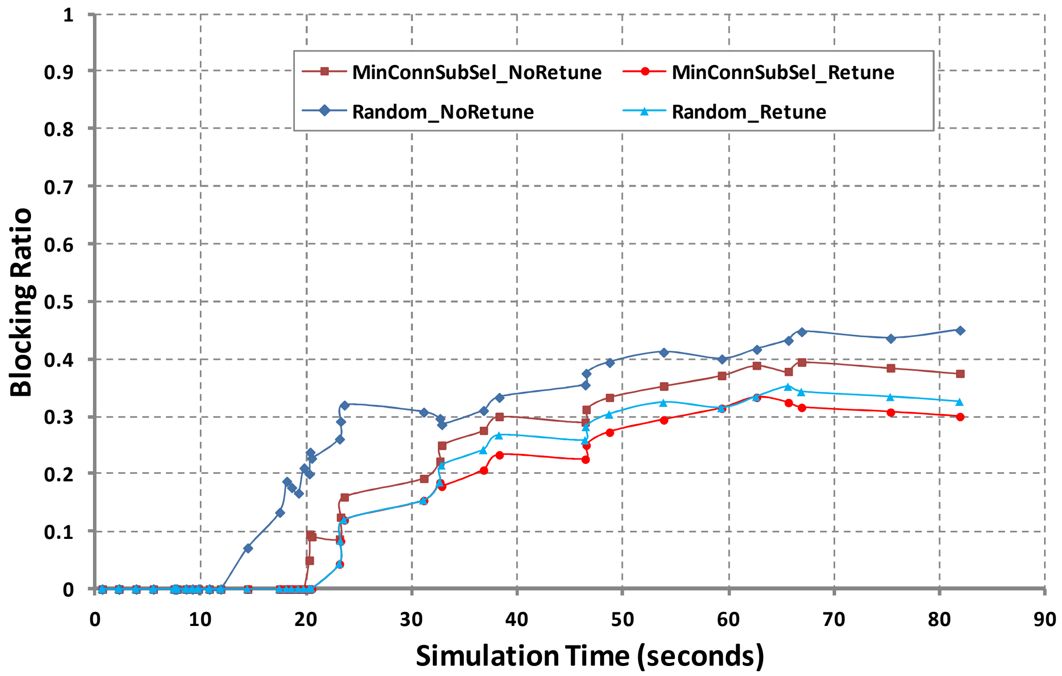

Figure 6-33 Effect on the blocking ratio with retuning enabled and disabled

Figure 6-33 compares the blocking ratio for the *MinConnSubSel* and the *Random* subband selection method when the retuning is enabled and disabled. As can be seen, under the condition when retune is not allowed, the blocking ratio for the *Random* selection method is greater than that for the *MinConnSubSel* method. This is expected since the random selection of subbands can lead to a highly uneven distribution of load amongst the subbands. The connections requesting the admission to the densely populated subband will lead to the blocking of the new connection and hence, increasing the blocking ratio. In contrast, *MinConnSubSel* method selects the subbands for the new connection such that load are approximately symmetric on all the subbands. Hence, there is less blocking of the new connections. For the condition when retune is allowed, the blocking ratio for the *MinConnSubSel* is just slightly better than that for the *Random* selection but lower than that with no retune condition. This is expected since the connections can retune from one forward subband to another to admit the new connection instead of blocking the new connection as in the case with no retune.

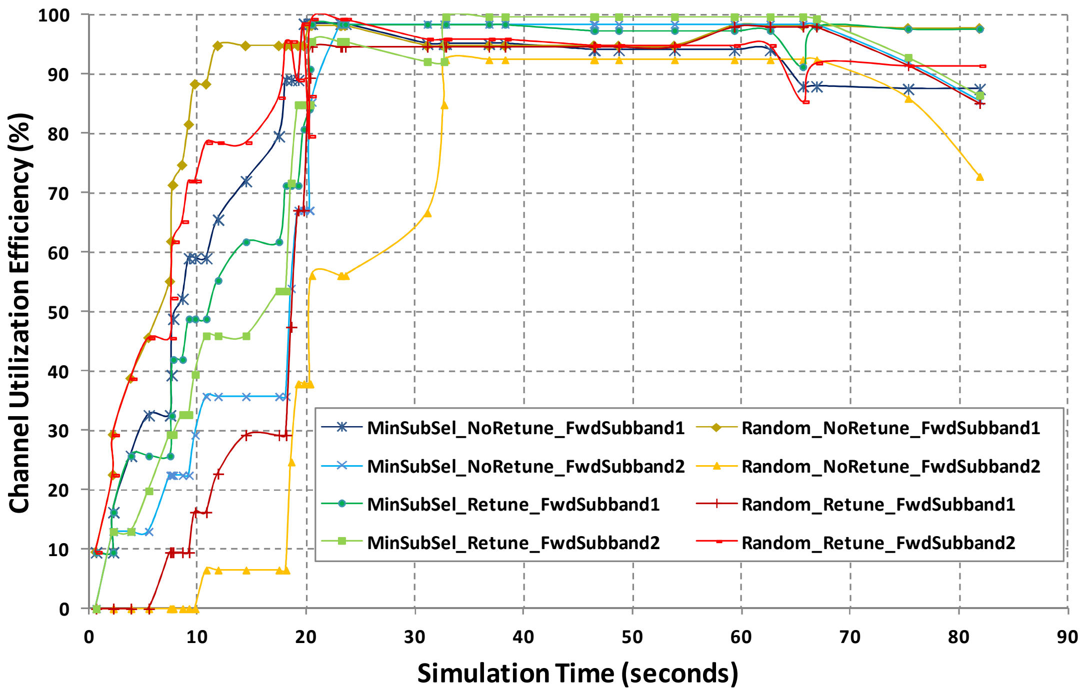

Figure 6-34 Effect on the channel utilization efficiency with retuning enabled and disabled

Figure 6-34 compares the channel utilization efficiency of the two available subbands using the *MinConnSubSel* and the *Random* subband selection method when the retuning is enabled and disabled. As can be seen, for the *Random* selection under the condition when retune is not allowed, the resource is highly unevenly distributed on the two forward subbands which is expected. Although, there is a slight improvement when retune is allowed; however, the subbands continue to have uneven distribution. In contrast, for the *MinConnSubSel* selection method, the two subbands are approximately equally balanced under the condition when retune is allowed. Even with no retune condition, the two subbands appear more balanced than in the *Random* selection method.

6.4.1.2 Test Case 2: Effect of the number of MTs

In this test case, the effect of the number of MTs on the system performance is tested under the condition when the retuning is allowed. The system is run with different number of MTs while keeping the same input connection queue. The following four conditions are considered:

- 5 MTs
- 10 MTs
- 15 MTs
- 20 MTs

For the test case, two forward subbands of the type, F80T4.5X8B, are considered and the *MinConnSubSel* subband selection method is used.

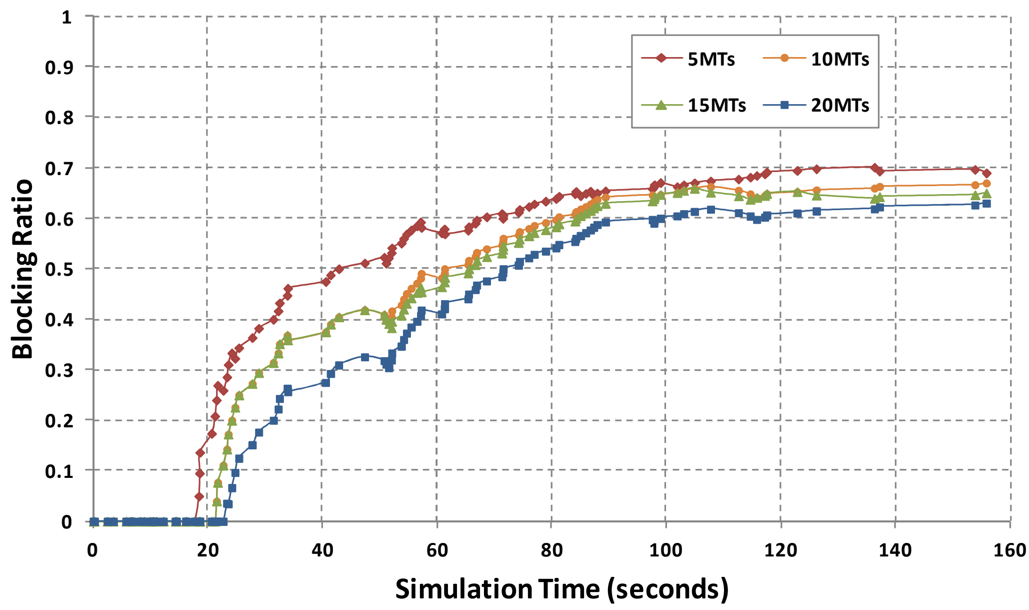

Figure 6-35 Effect of number of MTs on the blocking ratio

Figure 6-35 shows the effect of the number of MTs on the blocking ratio. As it can be seen, with the increased number of MTs there is a decline in the blocking ratio. This is an expected behaviour since the retuning between the forward subbands requires the MT to retune with all its connections. As mentioned earlier, the total number of connections for the scenario is fixed at

120 which imply that with fewer MTs such as 5, the average number of connections per MT is 24 and with higher number of MTs such as 20, the average number of connections per MT is 6. These statistics indicate that in order to retune an MT between the forward subbands with an average of 6 connections per MT is far successful than that with 24 connections. Hence, a large number of MTs result in a lesser number of connections being blocked and in turn a reduced blocking ratio.

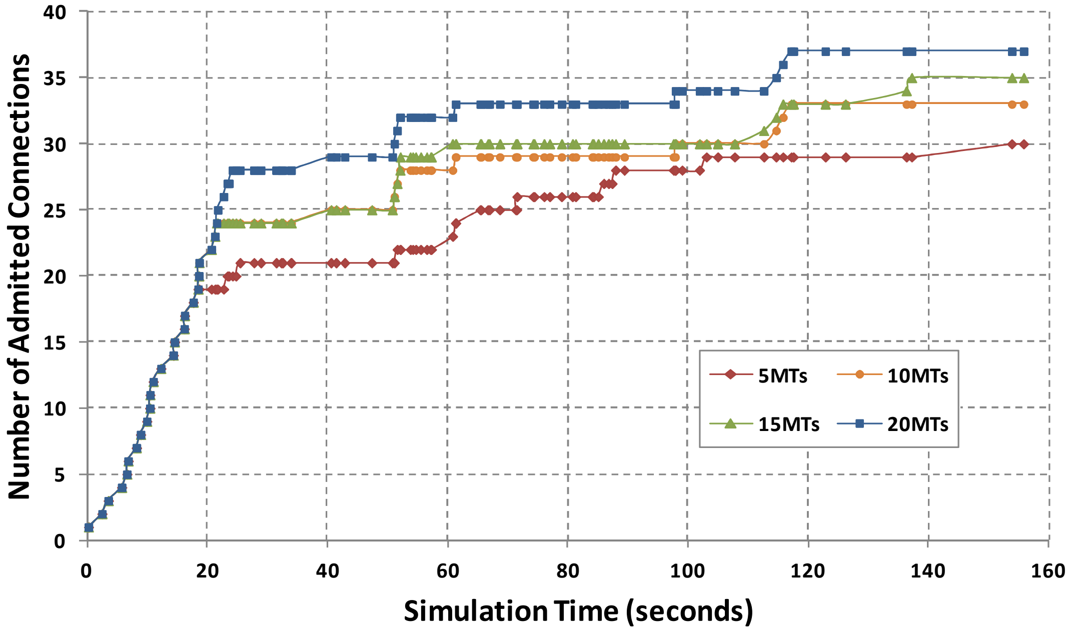

Figure 6-36 Effect of number of MTs on the number of admitted connections

Figure 6-36 shows the effect of the number of MTs on the number of admitted connections. As can be seen, the number of admitted connections increases as the number of MTs increases due to the decline in the blocking ratio as explained above.

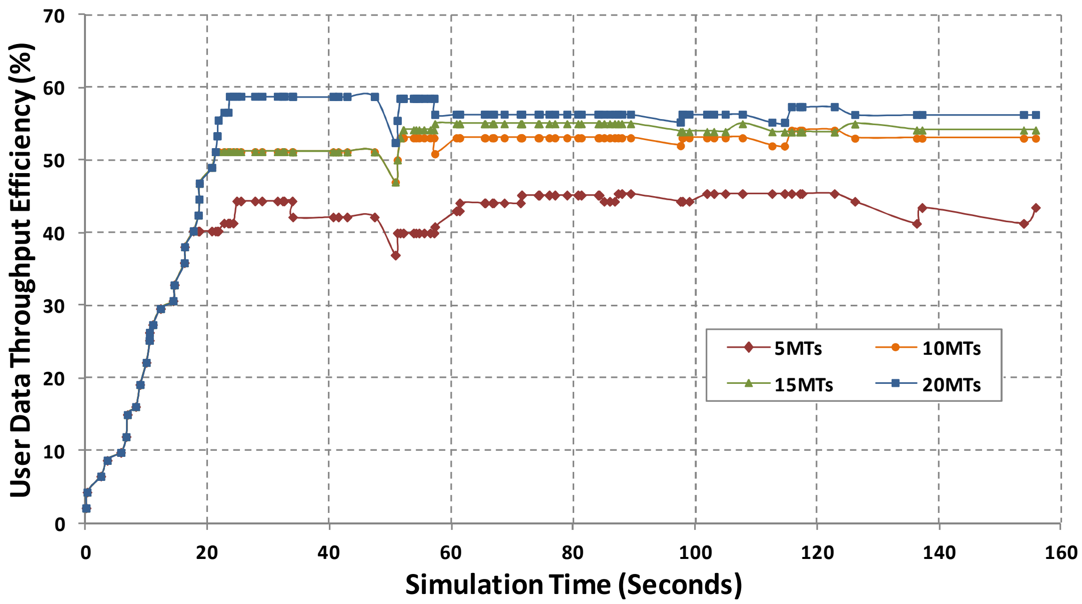

Figure 6-37 Effect of number of MTs on the user data throughput efficiency

Figure 6-37 shows the effect of the number of MTs on the average of the user data throughput efficiency of the two forward subbands. As can be seen, before the steady state is reached, all the graphs overlap each other. This is due to the fact that the amount of user data transferred from the link layer to the physical layer depends on the QoS parameter set for the four types of applications which remains same irrespective of the number of users used in the system. The steady state is reached faster for 5 MTs than for 20 MTs. This is explained by the fact that as the subband becomes congested, the possibility of admitting the connection by retuning with 5MTs is less than 20MTs as explained earlier. Hence, the connections get blocked earlier in the simulation time for 5MTs than 20MTs. Also, fewer connections are admitted into the system, therefore the user data throughput efficiency is less for 5MTs as compared to 20MTs in the steady state.

6.4.1.3 Test Case 3: Effect of number of available forward subbands

In this test case, the effect of the number of available forward subbands on the system performance is tested under the condition when the retuning is allowed and when the retuning is not allowed. The system is run with a different number of available forward subbands while keeping the same input connection queue. The following four available forward subbands are considered; 2, 4, 6, and 8. For the test case, the number of MTs is taken as 20 and the subband selection method is chosen as *MinConnSubSel.* The simulation is run for 180 seconds.

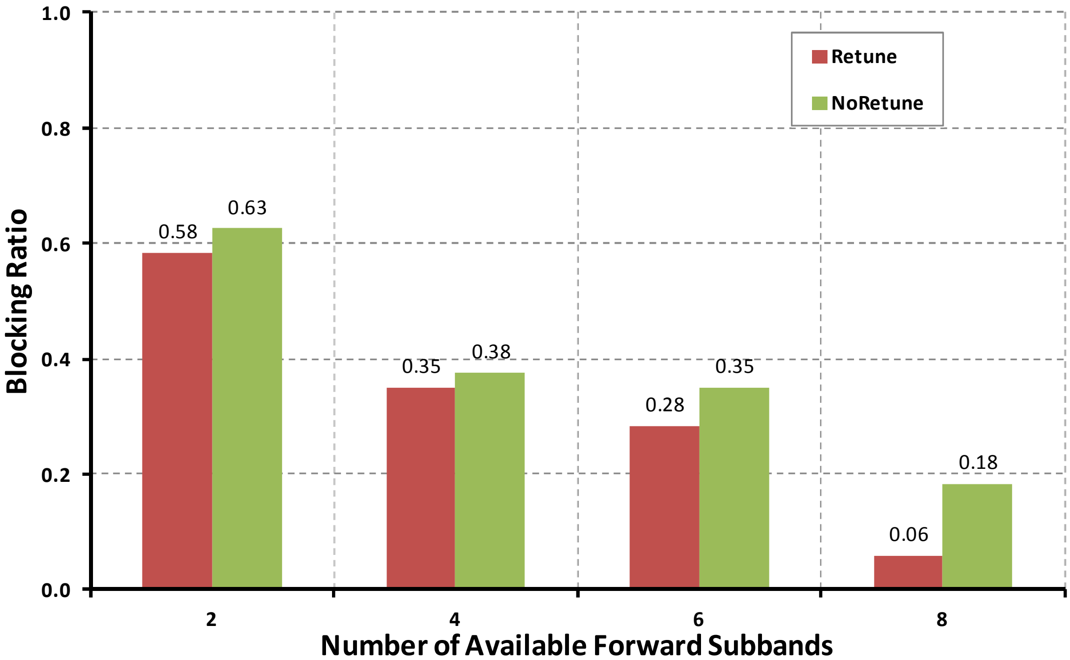

Figure 6-38 Effect of number of available forward subbands on the blocking ratio

Figure 6-38 shows the effect of the number of available forward subbands on the blocking ratio with and without retune. As can be seen, as the number of forward subband increases, the blocking ratio reduces. This is expected since the increased number of available forward subbands allows more available resource for the incoming connections. Also, the blocking ratio with retuning is lower than without retuning. This is due to the fact that when the retuning is allowed, the connections can retune to another forward subband which has enough resource to accommodate the connection. However, with retuning disabled, there is no possibility of retuning from one forward subband to another if there are not enough resources on the given subband to accommodate the new connection, hence leading to a higher blocking ratio.

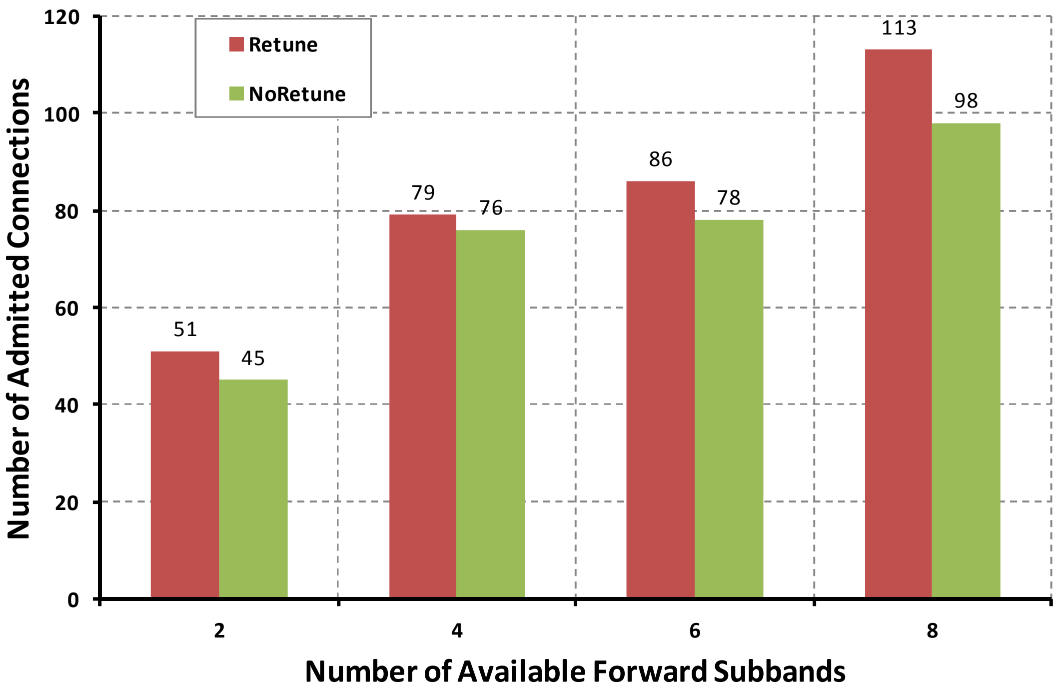

Figure 6-39 Effect of number of available forward subbands on the number of admitted connections

Figure 6-39 shows the effect of the number of available forward subbands on the number of admitted connections with and without retune. As it can be seen, the number of admitted connections increases as the number of forward subband increases due to the decline in the blocking ratio as explained above.

6.4.2 Results for Scenario 2: Mixed classes of Multicast Traffic

The objective of this scenario is to verify the working of the retune process for a mixed classes of multicast traffic.

Common Simulation Parameters		Values
No. of available forward subbands		2
Forward bearer type		F80T4.5X8B
Link condition		Low
Total number of MTs		5
Video Streaming	QoS (kbps)	32
	Source Utilization	0.8
	Mean Burst Period	0.1
	Avg. Holding time (sec)	300
	Avg. InterarrivalTime (sec)	5
	Traffic Class	5
	Users accessing (MT_ID)	1,3,4,5
File Distribution	QoS (kbps)	120
	Source Utilization	0.2
	Mean Burst Period	0.01
	Avg. Holding time (sec)	150
	Avg. Interarrival Time (sec)	5
	Traffic Class	6
	Users accessing (MT_ID)	2,3,4,5

Table 6-14 Scenario configuration for Mixed multicast traffic

Table 6-14 summarizes the MATLAB scenario configuration for the given scenario. Two types of multicast traffic are considered; video streaming and file distribution. The number of MTs is assumed to be 5 with each MT identified by a unique identifier, MT_ID for example MT_ID 1, MT_ID 2 and so on. For the simulation, the MTs with MT_ID 3, 4, and 5 accesses both the multicast services. User with MT_ID 1 accesses only video streaming service and user with MT_ID 2 accesses only file distribution service.

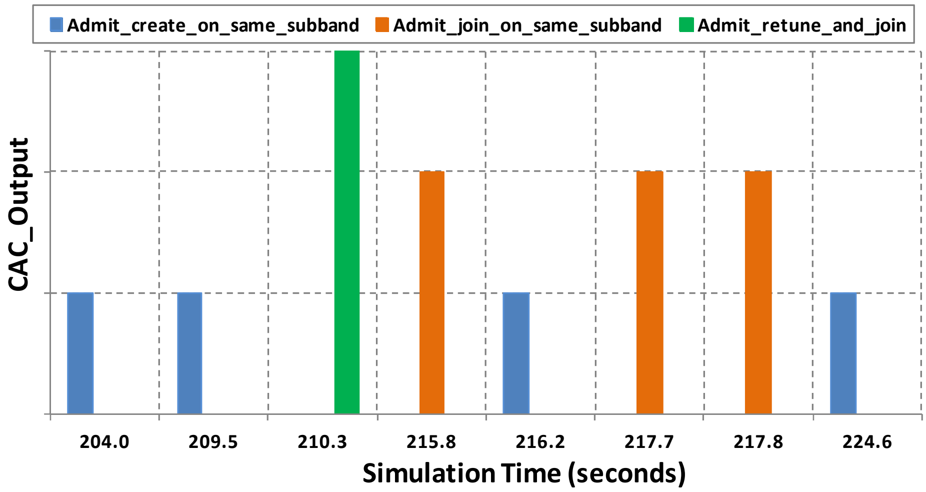

Figure 6-40 Retune behaviour for mixed classes of multicast traffic

Figure 6-40 shows the retune procedure for mixed classes of multicast traffic. The vertical axis is represented by the *CAC_Output* which indicates the action taken by the Admission Decision Controller in the CAC Processor as described in Section 4.2.2.1.4 when a new multicast connection request arrives. In the given test case where the pre-emption capability is disabled and only multicast traffic is considered, the possible actions indicated by *CAC_Output* are as follows:

- *Admit_create_on_same_subband* – Indicates the establishment of the new multicast connection on the given forward subband.
- *Admit_join_on_same_subband* – Indicates user joins the existing multicast connection on the given subband.
- *Admit_retune_and_join* – Indicates users retunes and joins an existing multicast connection on some other forward subband.

In order to fully understand the sequence of events leading to different outputs shown in Figure 6-40, Table 6-15 shows the sequence of events.

Since only two forward subbands have been assumed for the scenario, the FwdSubID takes two values 1 or 2.

Simulation Time (sec)	MT_ID	Fwd SubID	Application	*CAC_Output*
204.04	2	2	File Distribution	*Admit_Create_on_same_subband*
209.46	1	1	Video Streaming	*Admit_Create_on_same_subband*
210.58	3	1	Video Streaming	*Admit_retune_and_join*
215.76	4	1	Video Streaming	*Admit_join_on_same_subband*
216.19	3	1	File Distribution	*Admit_Create_on_same_subband*
217.68	4	1	File Distribution	*Admit_join_on_same_subband*
217.76	5	2	File Distribution	*Admit_join_on_same_subband*
224.59	5	2	Video Streaming	*Admit_Create_on_same_subband*

Table 6-15 Retune behaviour for mixed class of multicast traffic

The following are the sequence of events explaining the *CAC_Output* at the arrival of new multicast connection requests.

1. *Admit_Create_on_same_subband*: The first MT, MT_ID 2, requests the multicast service, file distribution, on FwdSubID 2. The connection is established as there are enough resources.
2. *Admit_Create_on_same_subband:* The first MT, MT_ID 1, requests the second multicast service, video streaming, on FwdSubID 1. The connection is established as there are enough resources.
3. *Admit_retune_and_join:* MT_ID 3 requests the video streaming multicast service on FwdSubID 2. However, there is an existing video

streaming multicast connection on FwdSubID 1. Hence, MT_ID 3 retunes to FwdSubID 1 and joins the existing multicast connection.

4. *Admit_join_on_same_subband* : MT_ID 4 initiates the video streaming connection on FwdSubID 1, and hence just joins the existing multicast connection.
5. *Admit_Create_on_same_subband* : MT_ID 3 on FwdSubID 1 initiates another multicast connection, file distribution connection. Although, there is an existing requested multicast connection running on FwdSubID 2, MT_ID 3 cannot be retuned to FwdSubID 2 as it is already running the video streaming multicast connection. Hence, MT_ID 3 establishes a new file distribution multicast connection on FwdSubID 1.
6. *Admit_join_on_same_subband:* MT_ID 4 on FwdSubID 1 initiates another multicast connection, file distribution connection. As MT_ID 3 created a new file distribution multicast connection on FwdSubID 1, MT_ID 4 just joins the existing multicast connection.
7. *Admit_join_on_same_subband:* MT_ID 5 initiates the file distribution connection on FwdSubID 2. As there is an existing file distribution connection created by MT_ID 2, MT_ID 5 just joins the existing multicast connection.
8. *Admit_Create_on_same_subband* : MT_ID 5 on FwdSubID 2 initiates another multicast connection, video streaming connection. Although, there is an existing requested multicast connection running on FwdSubID 1, MT_ID 5 cannot be retuned to FwdSubID 1 as it is already running the file distribution multicast connection. Hence,

MT_ID 5 establishes a new video streaming multicast connection on FwdSubID 2.

The above sequence of steps correctly verifies the admission decision process and the retune process associated with the multicast connection as explained in the section 4.2.3.

6.5 CAC Processor with only pre-emption capability

The Pre-emption Controller is triggered by the CAC Processor during the congestion when it is required to drop an existing connection in order to admit a higher priority connection as described in Section 4.2.4. In this section, the working of the Pre-emption Controller is verified under different test cases. Table 6-16 summarizes the test scenarios and test cases, together with test objectives.

Test Cases	Test Objectives
a) Effect of pre-emption procedure	To compare the system performance under following conditions: • pre-emption allowed • pre-emption not allowed • pre-emption randomly allowed
b) Effect of pre-emption algorithms	To compare the system performance using the following pre-emption algorithms: • Greedy • SubSetSum • Fuzzy
c) Effect of Connection Inactive Mechanism (CIM)	To compare the effect on the performance of the system using pre-emption algorithms with and without CIM procedure

Table 6-16 Simulation scenarios and test cases for *CAC Processor with only pre-emption capability*

Common Simulation Parameters		Values
MT Class		1,2,3
Number of MTs		20
Link condition		Low
Number of available forward subbands		1
Number of connections		32
Forward bearer type		F80T4.5X8B
Video Streaming	Number of connections	8
	QoS (kbps)	32
	Source Utilization	0.8
	Mean Burst Period	0.1
	Avg. Holding time (sec)	300
	Avg. Inter-arrival time (sec)	5
Netted Voice	Number of connections	8
	QoS (kbps)	60
	Source Utilization	0.3
	Mean Burst Period	0.01
	Avg. Holding time (sec)	240
	Avg. Inter-arrival time (sec)	5
Web Browsing	Number of connections	8
	QoS (kbps)	32
	Source Utilization	0.2
	Mean Burst Period	0.01
	Avg. Holding time (sec)	200
	Avg. Inter-arrival time (sec)	5
Email	Number of connections	8
	QoS (kbps)	120
	Source Utilization	0.2
	Mean Burst Period	0.01
	Avg. Holding time (sec)	150
	Avg. Inter-arrival time (sec)	5

Table 6-17 Scenario configuration for *CAC Processor with only Pre-emption capability*

Table 6-17 summarizes the MATLAB scenario configuration for the given scenario. Four types of unicast traffic are considered; the video streaming, the netted voice, the web browsing and the email, where each traffic type generates 8 connection requests. Only one forward subband of the type, F80T4.5X8B, is considered. The number of MTs is taken as 20 and the link condition for all MTs is assumed to be low. The admission control algorithm and the subband selection method are chosen as the *adaptive* and the *MinConnSubSel* respectively.

6.5.1 Test Case 1: Effect of the pre-emption procedure

In this test case, the performance of the system is compared under the following three conditions:

- *Pre-emption allowed* – All the connections are configured to be pre-emptable.
- *Pre-emption not allowed* – No connections are configured to be pre-emptable.
- *Pre-emption randomly allowed* – Some connections are selected to be pre-emptable on a random basis.

The configuration parameter, *Pre-emption Allowed*, defined in Table 5-3 is used to set the above three conditions.

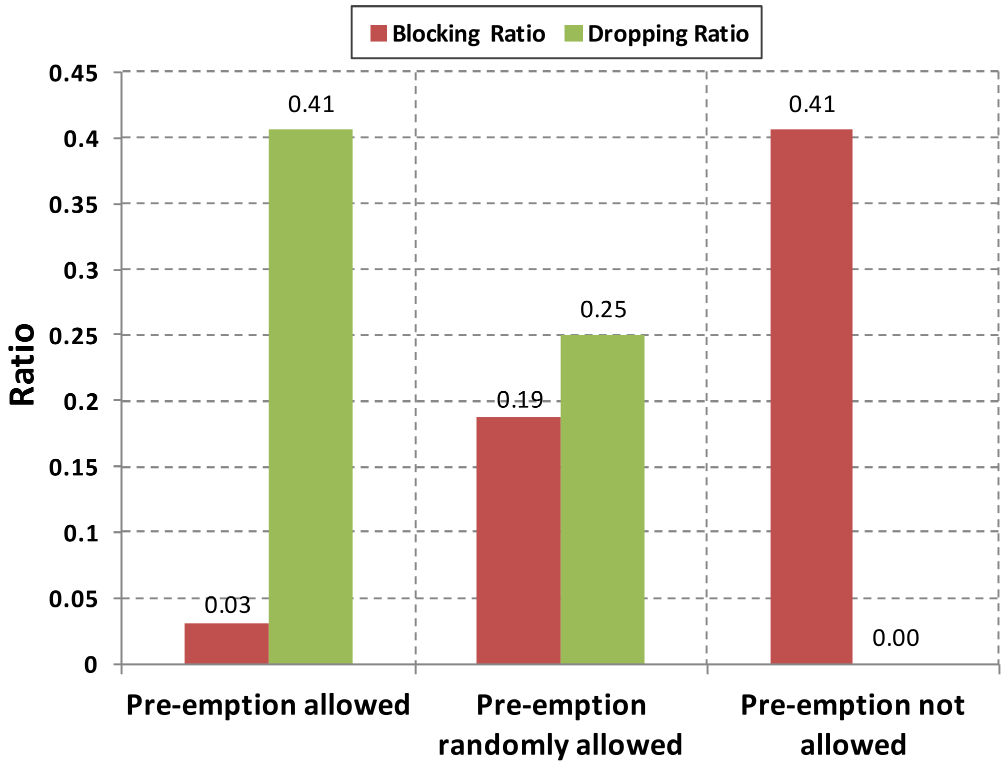

Figure 6-41 Effect of the pre-emption procedure on the blocking and the dropping ratio

Figure 6-41 shows the effect of the pre-emption procedure on the blocking and the dropping ratio. As can be seen, the blocking ratio is the highest for *Pre-emption not allowed* and lowest for *Pre-emption allowed*. However, reverse is true for the dropping ratio. Such results are expected. With the pre-emption procedure enabled, fewer number of connections are blocked as the connection of a higher priority are admitted into the system by dropping of one or more existing lower priority connections during congestion. This in turn implies a higher dropping ratio. On the other hand, with no pre-emption allowed, the connections are blocked irrespective of the priority once the given forward subband is fully occupied. Hence, the blocking ratio is very high and since there is no dropping of the connections, the dropping ratio is zero.

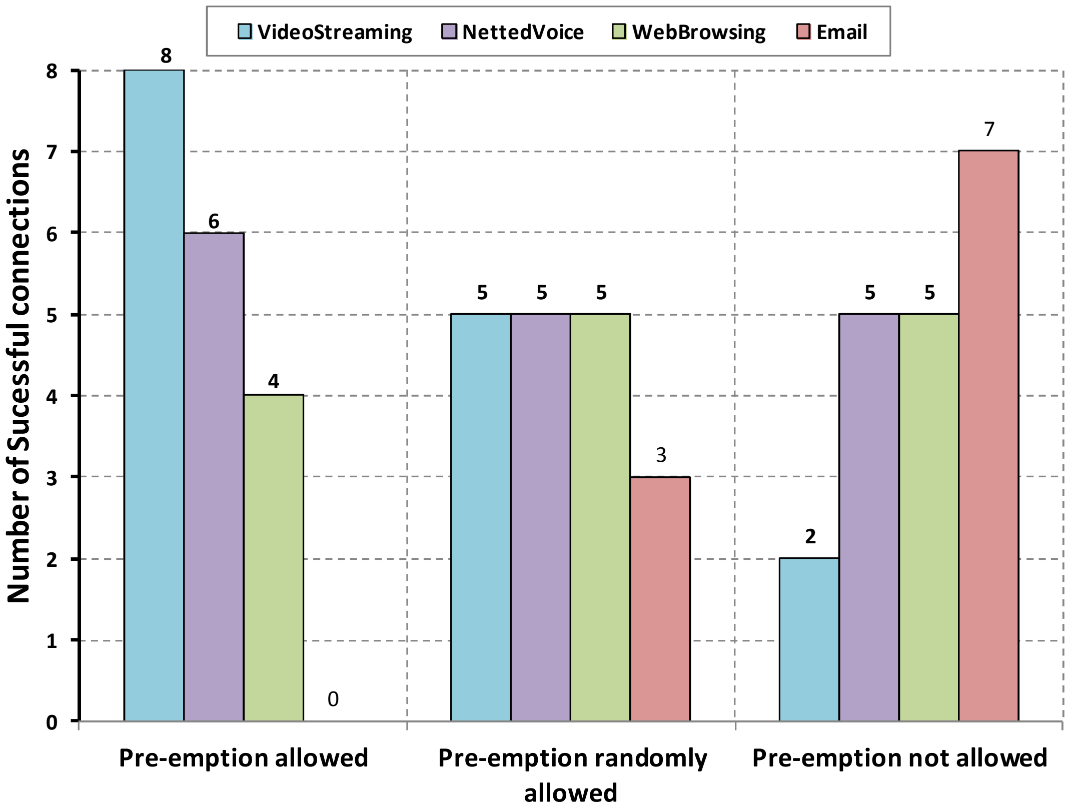

Figure 6-42 Effect of pre-emption procedure on the number of successful connections

Figure 6-42 shows the effect of the pre-emption procedure on the number of successful connections. As can be seen for *Pre-emption allowed*, all the video streaming connections are admitted since they are the highest priority connections. As the priority drops for the different traffic in the given order, netted voice > web browsing > email, the number of successful connections also reduces as most of these lower priority connections are dropped. Since email connections have the lowest priority, they are dropped for all other higher priority connections and hence, there are no successful email connections. On the other hand for *Pre-emption not allowed*, the connections are not admitted according to their priority resulting in only 2 video streaming connections as opposed to 8 in the case when pre-emption is allowed. The connections are either admitted if there is enough capacity or the connection is dropped. Under the condition, *Pre-emption randomly allowed,* only 5 video streaming connections are successful. Although, video streaming connections cannot be configured to be pre-emptable since they are the highest priority, only 5 could be admitted whereas the rest have been blocked. For other lower priority connections, the connections are dropped to admit a higher priority connection depending on the pre-emptable status of the connection resulting in 3 successful lowest priority email connections.

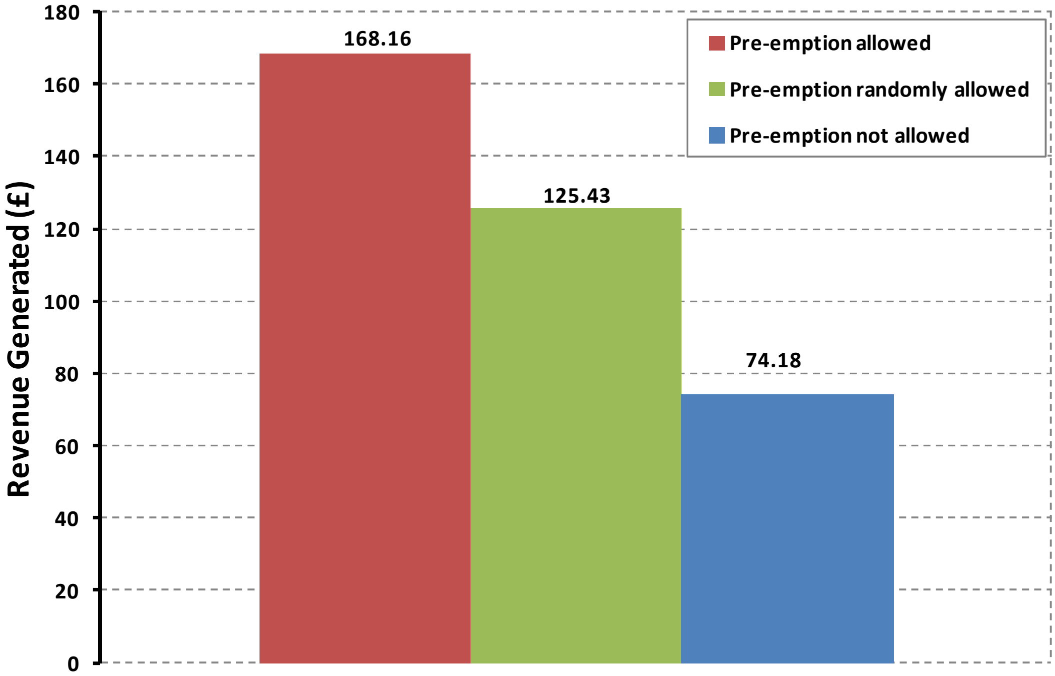

Figure 6-43 Effect of pre-emption procedure on the revenue generation

Figure 6-43 shows the effect of pre-emption procedure on the revenue generated by the network. As can be seen, the revenue generation is the highest for *Pre-emption allowed* and lowest for *Pre-emption not allowed*, and lies in between for *Pre-emption randomly allowed*. This is directly related to the number of successful connections for each priority connections. For example, the pre-emption procedure tries to maximize the higher priority connections which have the highest revenues whereas with no pre-emption, the highest numbers of successful connections are email connections as shown in Figure 6-42 and hence, *Pre-emption not allowed* has the lowest revenues.

6.5.2 Test Case 2: Effect of pre-emption algorithms

In this test case, the following three pre-emption algorithms are tested and compared against each other:

- *Greedy*
- *SubSetSum*
- *Fuzzy*

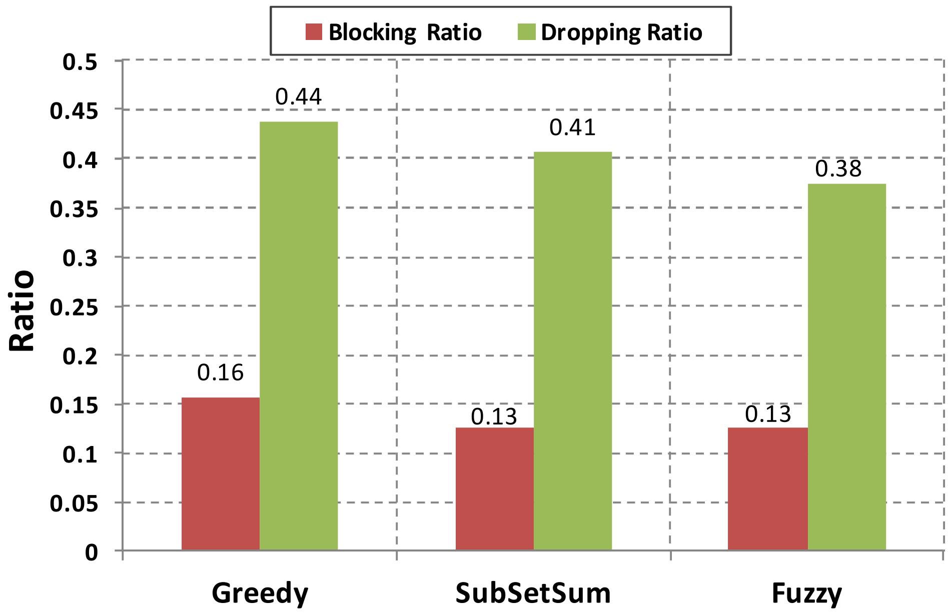

Figure 6-44 Comparison of the blocking and the dropping ratio for different pre-emption algorithms

Figure 6-44 compares the blocking and the dropping ratio for different pre-emption algorithms. As can be seen, the blocking ratio is slightly higher for the *Greedy* algorithm than the *SubSetSum* and the *Fuzzy* algorithms which have same blocking ratio. Also, the dropping ratio is the highest for the *Greedy* algorithm followed by the *SubSetSum* and the *Fuzzy* algorithms. This is expected as the *Greedy* algorithm admits the higher priority connections by pre-empting as many lower priority connections as required starting with the lowest resource using connection. In doing so, a large number of lower priority connections are dropped. Hence, the dropping ratio is the highest. This also results in increased blocking of the new higher priority connections as there may not be enough lower priority connections to drop. The *SubSetSum* algorithm drops the connections in an optimum way by selecting the connection from the list of pre-emptable connections such that its resource consumption is just enough to admit a higher priority connection. Hence, as compared to *Greedy*, it drops fewer connections. *Fuzzy* algorithm considers multiple criteria such as priority, remaining service time, and the connection capacity, while deciding the connections to be pre-empted. The rules are set such that the connection with the lowest priority, longest remaining time and smallest connection capacity has the highest chance of being pre-empted. Such rules ensure that the connections with short remaining service time are less probable to be dropped allowing more successful departures from the system in turn increasing the possibility of admitting more connections and hence, reducing the dropping ratio.

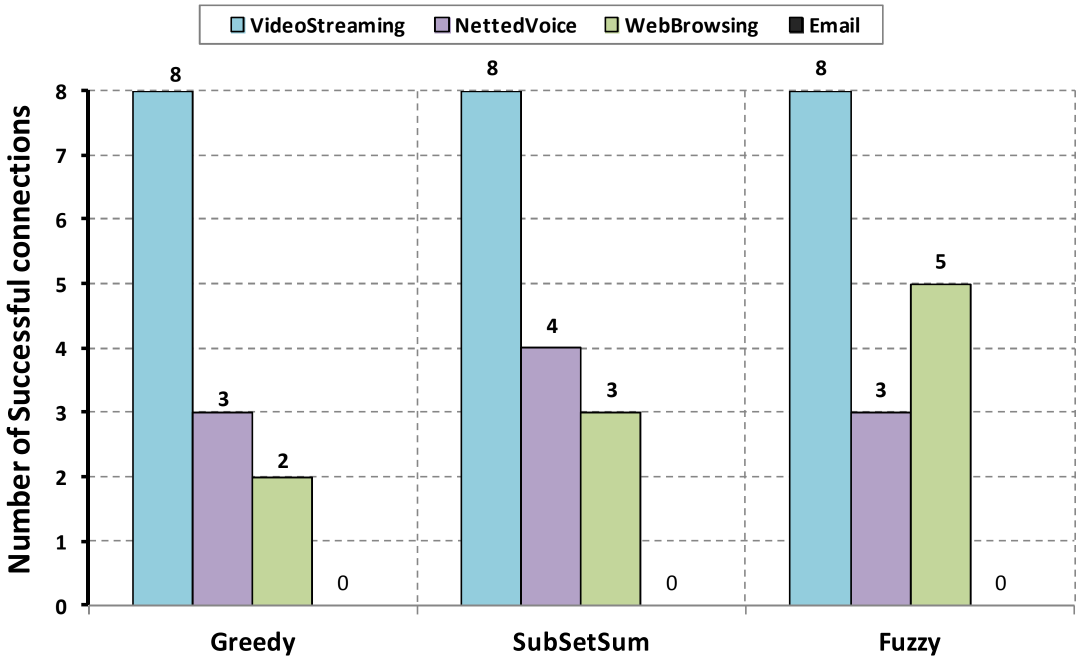

Figure 6-45 Comparison of number of successful connections for different pre-emption algorithms

Figure 6-45 compares the number of successful connections for different pre-emption algorithms. As can be seen, all 8 video streaming connections and no email connections are admitted for each pre-emption algorithms. This is expected as the video streaming are the highest priority connections and the email are the lowest priority connections. For the netted voice and web browsing connections, the *Fuzzy* algorithm admits the maximum connections followed by *SubSetSum* and *Greedy*. This is in accordance with the increased dropping ratio shown in Figure 6-44 in the same order. As more connections are dropped, fewer connections are successful.

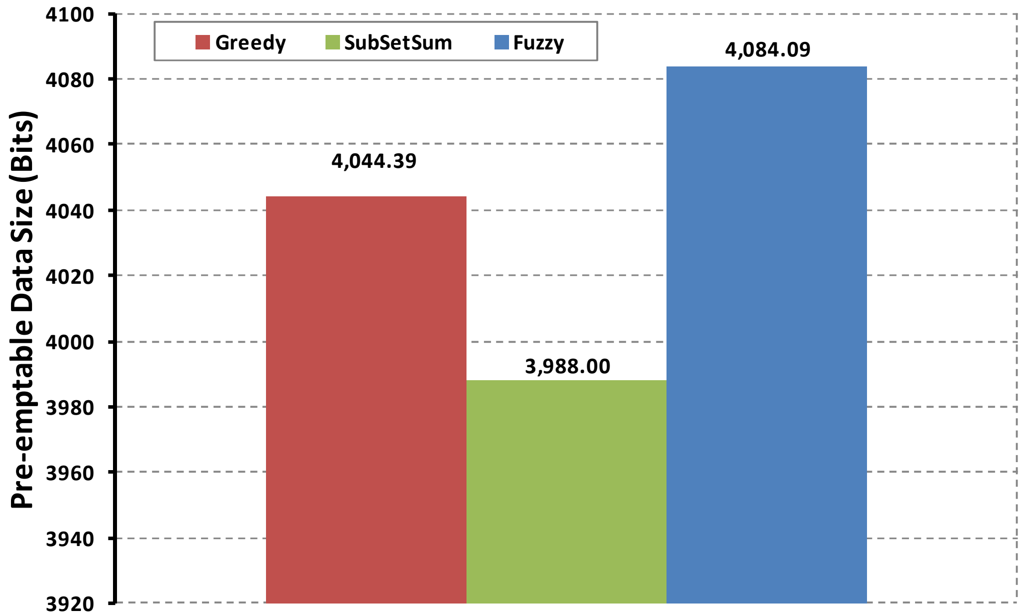

Figure 6-46 Comparison of pre-emptable data size for different pre-emption algorithms

Figure 6-46 compares the pre-emptable data size in bits for different pre-emption algorithms. As can be seen, the *SubSetSum* has the lowest pre-emptable data size followed by the *Greedy* and the *Fuzzy*. This is expected as the *SubSetSum* algorithm tries to minimize the pre-emptable data size by selecting the connections to be pre-empted such that its resource consumption is just enough to admit a higher priority connection. The *Greedy* algorithm also tries to keep the pre-emptable data size minimum by pre-empting as many lower priority connections as required starting with the lowest resource using connection. However, it does not do it in an optimum manner and hence, the pre-emptable data size is slightly higher than the *SubSetSum*. For the *Fuzzy* algorithm, although the rules ensure that the connections with low connection capacity have the highest chance of being pre-empted which helps to minimize the pre-emptable data size, however, at the same time the algorithm also tries to keep the dropping ratio minimum and to maximize the number of successful connections. In doing so, the pre-emptable data size is higher than both the *Greedy* and the *SubSetSum* algorithms.

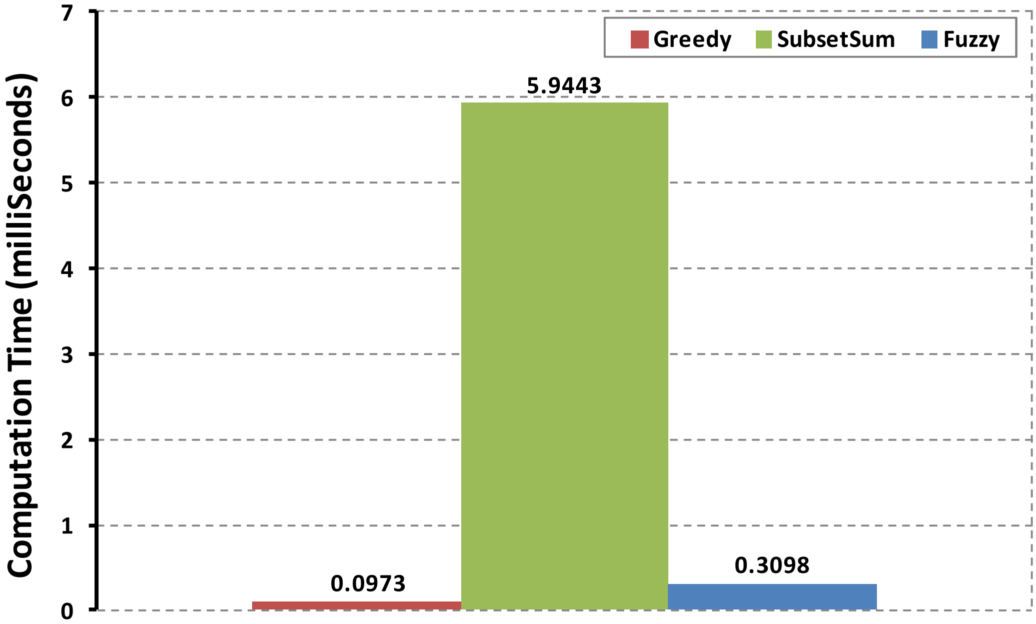

Figure 6-47 Comparison of computation time for different pre-emption algorithms

Figure 6-47 compares the computation time for different pre-emption algorithms. As can be seen, the *SubSetSum* algorithm has the highest computation time followed by the *Fuzzy* and the *Greedy* algorithms. This is expected since the *SubSetSum* algorithm is based on the SubSetSum problem which is computationally heavy and requires a lot of loop iterations to find the optimum solution. The *Fuzzy* algorithm also provides the optimum solution. However, it is easy to understand and takes much less time to compute as compared to *SubSetSum*. Although the *Greedy* algorithm gives higher blocking and dropping ratio, it is easy to implement and computationally very fast.

6.5.3 Test Case 3: Effect of Connection Inactive Mechanism (CIM)

In this test case, the effect on the performance of the system is compared using *Fuzzy* pre-emption algorithm with and without the CIM procedure.

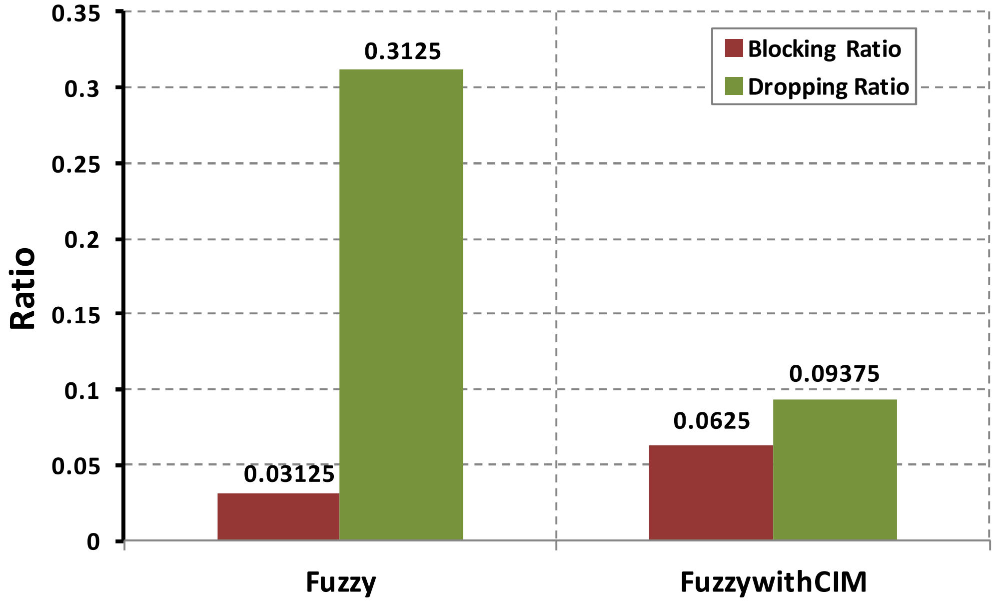

Figure 6-48 Comparison of the blocking and the dropping ratio using *Fuzzy* pre-emption algorithm with and without CIM

Figure 6-48 compares the blocking and the dropping ratio using *Fuzzy* pre-emption algorithm with and without CIM. As can be seen, there is a slight increase in the blocking ratio with CIM as compared to without CIM. However, there is a large decline in the dropping ratio for the same. This is expected as with the CIM procedure, the connections selected to be pre-empted are not dropped immediately. However, they are converted into *Non-Active* state where the resources used by the selected connections are released to admit a higher priority connection. These *Non-Active* connections requests for the resources after the set time and if there are enough resources, the connection is made *Active* again otherwise it is dropped. Such mechanism allows the opportunity for assigning the resources to the connections at a later time as compared to instant dropping of the connections and hence, reducing the dropping ratio.

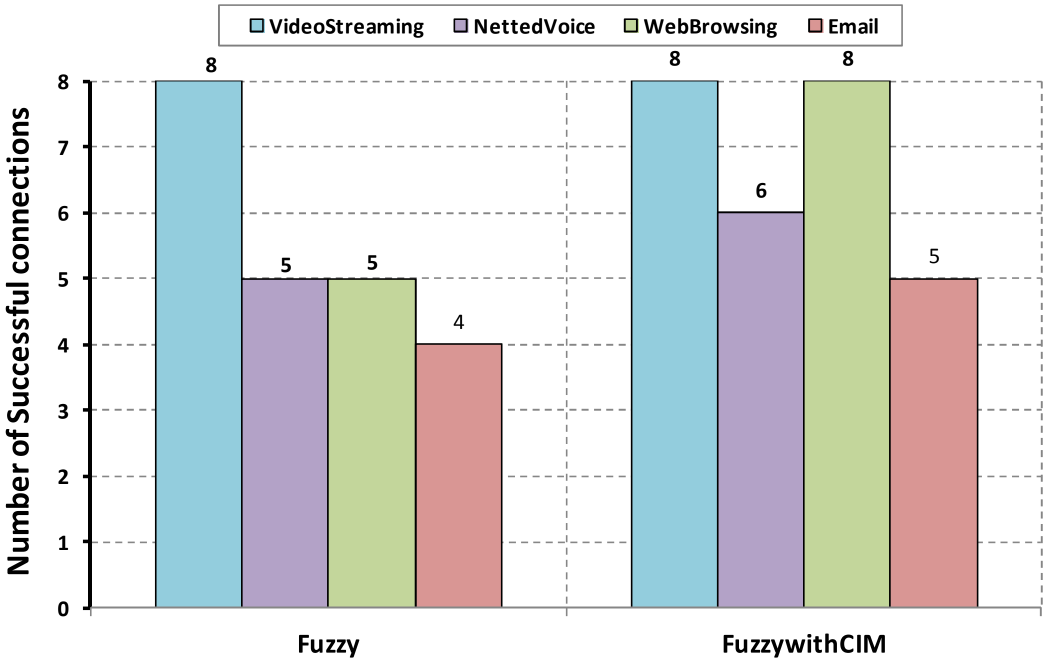

Figure 6-49 Comparison of number of successful connections using *Fuzzy* pre-emption algorithm with and without CIM

Figure 6-49 shows the number of successful connections using *Fuzzy* pre-emption algorithm with and without CIM. As can be seen using CIM the number of successful connections increases. This is due to the fact that the number of connections dropped reduces using CIM. Hence more connections can successfully finish their service time.

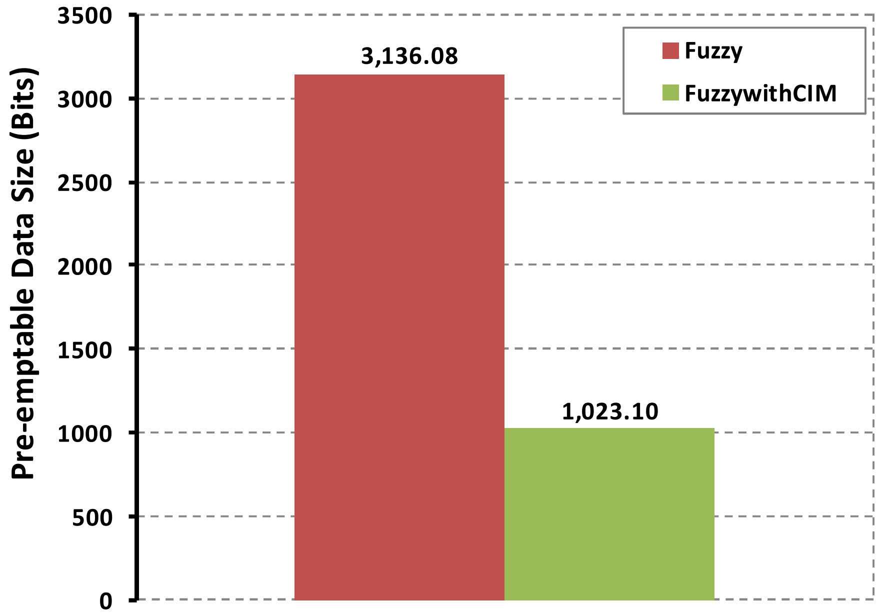

Figure 6-50 Comparison of the pre-emptable data size using *Fuzzy* pre-emption algorithm with and without CIM

Figure 6-50 compares the pre-emptable data size using fuzzy pre-emption algorithm with and without CIM. As can be seen, the amount of pre-emptable data size is much lower with CIM as compared to without CIM. This is an expected result as it is a direct impact of the number of connections dropped which are much lower with CIM.

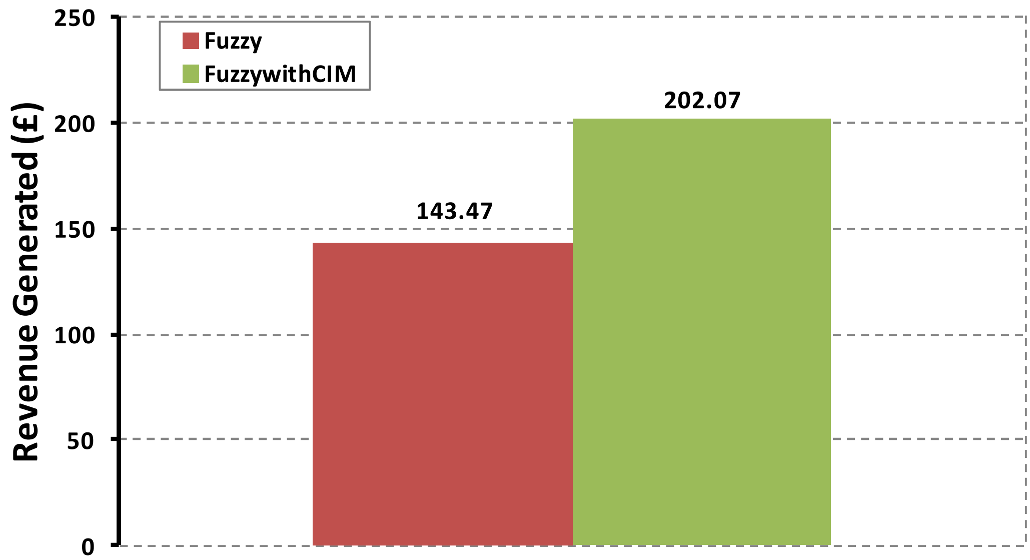

Figure 6-51 Comparison of the revenue generation using *Fuzzy* pre-emption algorithm with and without CIM

Figure 6-51 compares the revenue generation using *Fuzzy* pre-emption algorithm with and without CIM. As can be seen, the revenue generation is higher with CIM as compared to without CIM. This is directly correlated with the number of successful connections for each priority connections which are higher with CIM as shown in Figure 6-49.

6.6 CAC Processor with both retune and pre-emption capabilities

The CAC framework consists of three different functionalities; the CAC Processor, the Retune Controller, and the Pre-emption Controller. These functionalities have been tested and verified separately in the sections 6.3, 6.4, and 6.5 respectively. In this section, the simulation configuration is such that the functionalities; retuning and pre-emption, are both enabled allowing the working of the CAC framework as a whole to be verified.

Common Simulation Parameters		Values
Number of MTs		20
MTClass		1,2,3
Link condition		Low
Number of available forward subbands		2
Bearer Types		F80T4.5X8B
Video Streaming	Number of connections	15
	QoS (kbps)	32
	Source Utilization	0.8
	Mean Burst Period	0.1
	Avg. Holding time (sec)	300
	Avg. Inter-arrival time (sec)	5
Netted Voice	Number of connections	15
	QoS (kbps)	60
	Source Utilization	0.3
	Mean Burst Period	0.01
	Avg. Holding time (sec)	240
	Avg. Inter-arrival time (sec)	5
Web Browsing	Number of connections	15
	QoS (kbps)	32
	Source Utilization	0.2
	Mean Burst Period	0.01
	Avg. Holding time (sec)	200
	Avg. Inter-arrival time (sec)	5
Email	Number of connections	15
	QoS (kbps)	120
	Source Utilization	0.2
	Mean Burst Period	0.01
	Avg. Holding time (sec)	150
	Avg. Inter-arrival time (sec)	5

Table 6-18 Scenario configuration for *CAC Processor with both retune and pre-emption capabilities*

Table 6-18 summarizes the MATLAB configuration for the given simulation. Four types of unicast traffic are considered; the video streaming, the netted voice, the web browsing and the email, where each traffic type generates 15 connection requests. Two forward subbands of the type, F80T4.5X8B, are considered. The number of MTs is taken as 20 and the link condition for all MTs is assumed to be low. The admission control algorithm, the subband selection method, and the pre-emption algorithm are chosen as the *adaptive,* the *MinConnSubSel,* and the *Fuzzy* algorithms respectively, wherever applicable.

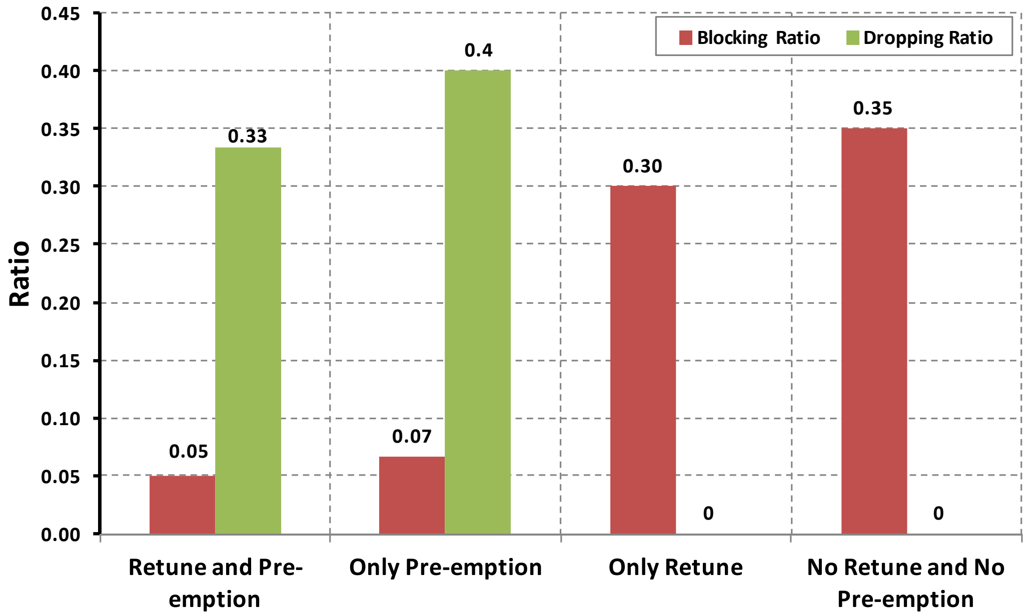

Figure 6-52 Comparison of the blocking and the dropping ratio for different settings of CAC framework

Figure 6-52 compares the blocking and dropping ratio for the following simulation configuration settings:

- *Retune and Pre-emption* – CAC Processor with both retune and pre-emption capabilities
- *Only Pre-emption* – CAC Processor with only pre-emption capability
- *Only Retune* – CAC Processor with only retune capability
- *No Retune and No Pre-emption* – CAC Processor without retune and pre-emption capabilities

Comparing *Retune and Pre-emption* with *Only Pre-emption*, it can be seen that the blocking and the dropping ratio is slightly lower for the *Retune and Pre-emption*. This is expected as under congested state, *Only Pre-emption* allows the new connection on a given forward subband to be either blocked or admitted by dropping the existing connections. However, *Retune and Pre-emption* allows the new connection to be admitted by retuning to another forward subband under congested state as opposed to dropping the existing connections on the given subband or blocking the new connection if there are not enough lower priority connections to be dropped. This results in the decline in the blocking and the dropping ratio. Comparing the *No Retune and No Pre-emption* with *Only Retune*, the blocking ratio is slightly lower for *Only Retune.* This is expected as *No Retune and No Pre-emption* either admits or blocks the new connection on the given forward subband. On the other hand, *Only Retune* allows the users to move from one forward subband to another under the condition when the given subband is congested, resulting in the decline in the number of connections blocked and hence, the blocking ratio. As the pre-emption capability is not enabled in either configuration, the dropping ratio is shown to be zero.

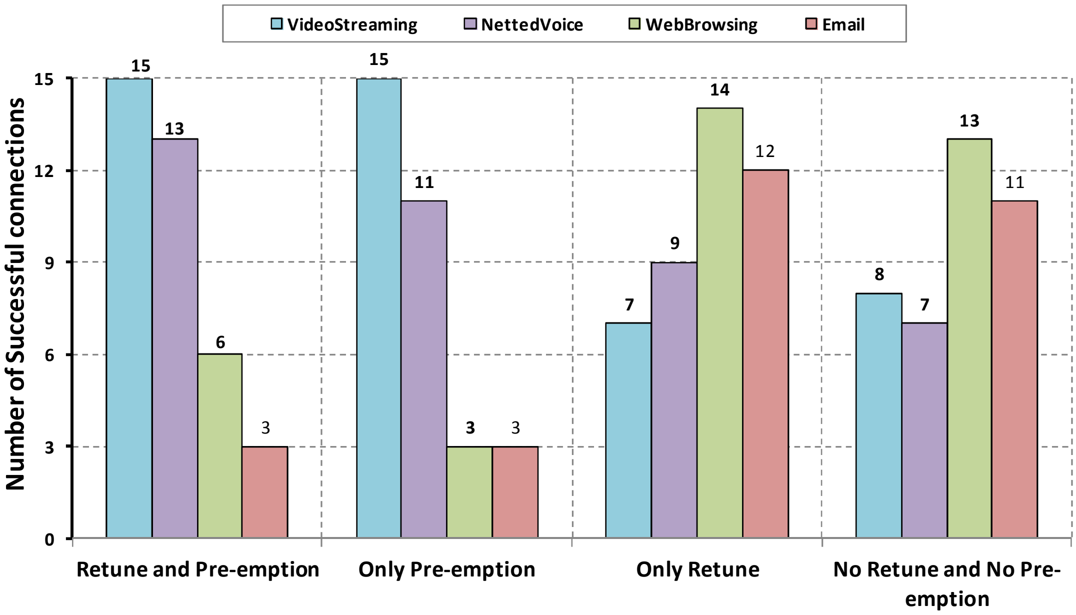

Figure 6-53 Comparison of number of successful connections for different settings of CAC framework

Figure 6-53 compares the number of successful connections for the different simulation configuration settings. As can be seen, when pre-emption procedure is allowed such as in the setting *Retune and Pre-emption* and *Only Pre-emption,* all the video streaming connections are successful as these connections are given the highest priority and cannot be pre-empted. Comparing the lower priority connections such as netted voice, web browsing and email, it can be seen that there are more successful connections for *Retune and Pre-emption* as compared to *Only Pre-emption.* This is due to the fact that more connections are dropped and blocked for *Only Pre-emption* as compared to *Retune and Pre-emption* as explained above. When no pre-emption is applied such as for the settings, *No Retune and No Pre-emption* and *Only Retune,* there is no precedence value assigned to the type of traffic. As can be seen the total number of successful connections are slightly more for *Only Retune* as compared to *No Retune and No Pre-emption.* This is explained by the slightly lower blocking ratio for *Only Retune* as compared to *No Retune and No Pre-emption* as explained above.

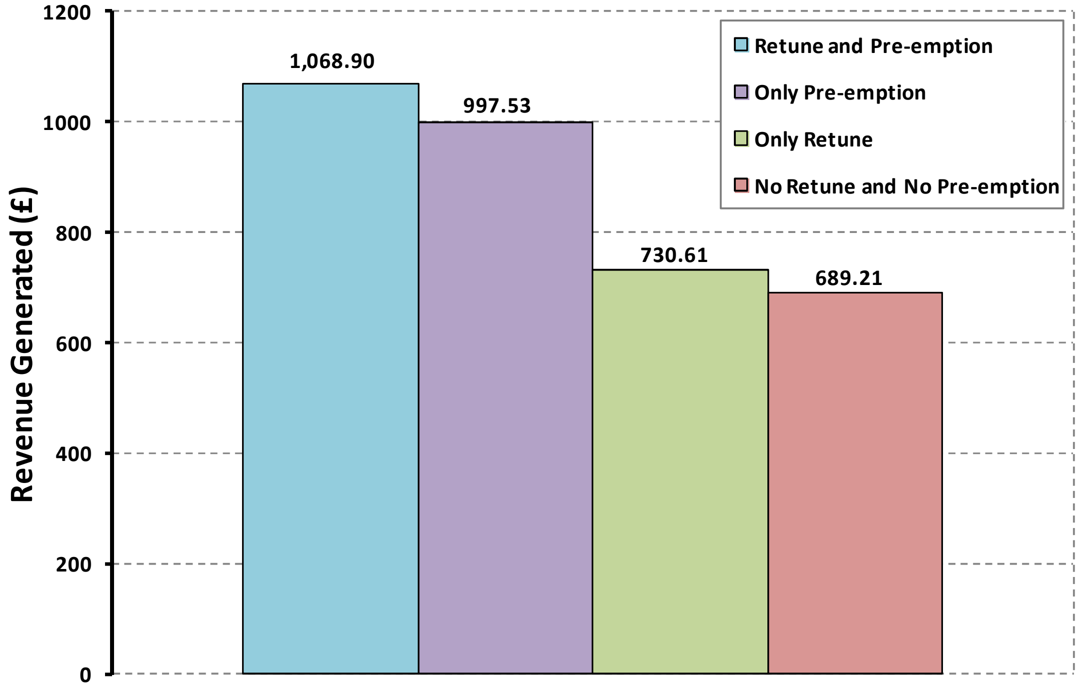

Figure 6-54 Comparison of revenue generation for different simulation configuration settings

Figure 6-54 compares the revenues generated for the different simulation configuration settings. As can be seen, the revenue generation is the highest for *Retune and Pre-emption* and the lowest for *No Retune and No Pre-emption.* This is directly correlated with the number of successful connections for each priority connections. The highest priority connection such as video streaming is assumed to generate the highest revenue followed by the netted voice, web browsing and the email. The *Retune and Pre-emption* and *Only Pre-emption* maximize the number of higher priority successful connections. However, *Retune and Pre-emption* has higher successful connections than *Only Pre-emption* and hence the revenue generation is the highest. On the other hand, *Only Retune* and *No Retune and No Pre-emption*, do not consider the priority of the connections resulting in the lower number of successful connections of higher priorities. Since, the number of successful connections for *No Retune and No Pre-emption* is lower than the *Only Retune*, the revenue generated is the lowest.

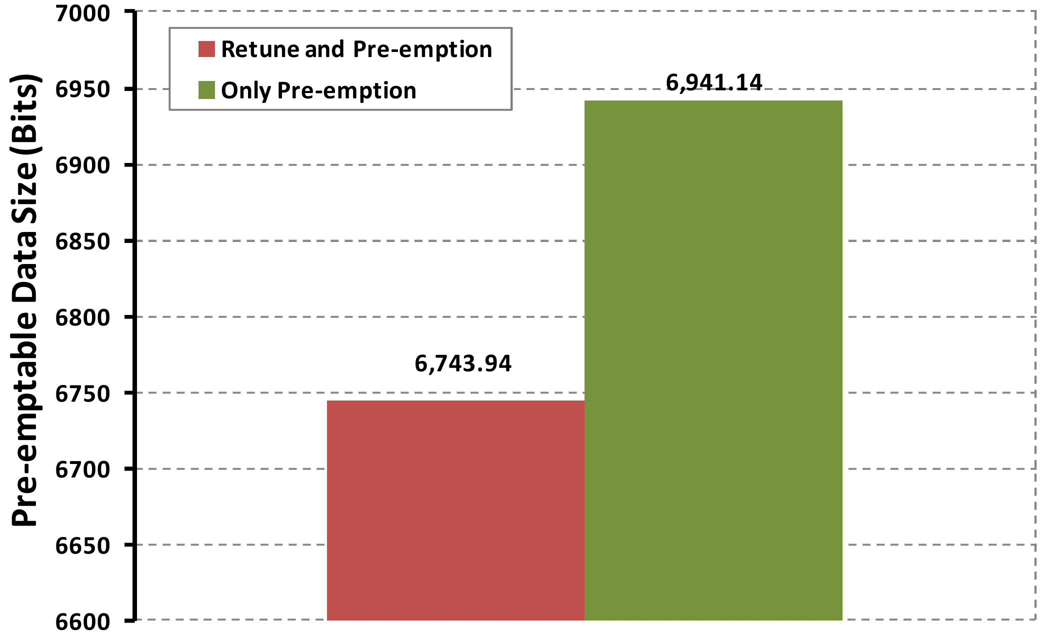

Figure 6-55 Comparison of pre-emptable data size for *Retune and Pre-emption* and *Only Pre-emption* simulation configuration settings

Figure 6-55 compares the pre-emptable data size for *Retune and Pre-emption* and *Only Pre-emption* Simulation configuration settings. As can be seen, the pre-emptable data size is lower for *Retune and Pre-emption* as compared to *Only Pre-emption*. This is an expected result as the number of connections dropped is lower for *Retune and Pre-emption* and hence, the bandwidth released by the pre-empted connections is less.

6.7 Summary

This chapter described different simulation scenarios and the results obtained. The simulations have been broadly classified according to the three different functionalities of the CAC framework; CAC Processor, Retune

Controller and the Pre-emption Controller. The following four classifications have been presented:

1) CAC Processor without retune and pre-emption capabilities
2) CAC Processor with only retune capability
3) CAC Processor with only pre-emption capability
4) CAC Processor with both retune and pre-emption capabilities.

In each simulation classification, various scenarios have been described. In the first simulation, it is proved that the adaptive admission control algorithm performs better than the non-adaptive algorithm by taking into consideration variable subband capacity such as adapting to the changing link conditions. Also, the results indicate the correct working of the admission decision process for multicast traffic. In addition, the results showing the behaviour and the performance of the system under a mix of unicast and multicast traffic has been presented.

In the second simulation, it is shown that the proposed *MinConnSubSel* subband selection method is better than the *Random* subband selection method. For the unicast traffic, the effect of the number of MTs for same number of connection on the number of retunes is presented. The results indicate an efficient retuning and higher system performance with lower number of connections per MT. For the multicast traffic, the correct working of the retune process is shown in the results.

In the third simulation, it is seen that the Fuzzy pre-emption algorithm performs better than the SubSetSum and the Greedy pre-emption algorithms for most of the performance metrics except for the pre-emptable bandwidth which is highest for the Fuzzy algorithm. The results also present a comparison between the system with the pre-emption procedure enabled and disabled. It is concluded that the system performs better with the pre-emption in terms of the connections admission according to the priority. A further enhancement in the performance of the pre-emption algorithms have been shown by the CIM procedure. The results indicate a large decline in the dropping ratio and the pre-emptable bandwidth with CIM as compared to using Fuzzy pre-emption algorithm without CIM.

Finally in the fourth simulation, a comparison has been made between all the simulation classifications. The results indicate the CAC framework as a whole with all the functionalities performs better than the other simulation settings.

In summary, the CAC framework proposed in this thesis provides retune mechanism and the pre-emption mechanism along with the admission control procedure for a S-UMTS system which work together to increase the performance of the system. Each element of the CAC framework has been tested and verified in this chapter.

Chapter 7 : CONCLUSION AND FUTURE WORK

7.1 Conclusion

The objective of this thesis work is to design a CAC framework for a UMTS-compatible satellite system with variable link capacity supporting both unicast and multicast traffic, which aims at satisfying the following general RRM requirements:

- The support of the UMTS traffic classes; Streaming, Interactive, and Background;
- The support of simultaneous operation of different types of service to the same terminal and to different mobile terminals with different capabilities;
- The support of service prioritisation and pre-emption schemes;
- The support for different physical bearer types;
- The support for one or more Multicast Radio Access Bearer Services simultaneously;
- The support for Multicast Radio Access Bearer Service and Unicast Access Bearer Services simultaneously from the same MT.

The proposed CAC framework comprises of three different functionalities: the admission control procedure, the retune procedure and the pre-emption procedure, which work together to fulfil the above RRM requirements. The benefit of such a design lies in the fact that the different functionalities of the framework are not interleaved, allowing the independent enhancement of each functionality, if required in the future. Another advantage of this framework is its scalability and flexibility. For instance, introducing new types of services with different capabilities require modifications only in the sub-function of the admission control procedure. Another example of the flexibility of this framework is that changing the number of forward subbands supported by the system requires only modification in the retune procedure without the need of a complete redesign of the CAC framework. In order to evaluate the performance of the designed framework, it has been simulated using the MATLAB Simulink simulation tool. MATLAB is a high-performance language which integrates computing, visualization, and programming in an easy-to-use environment. Simulink is a graphical extension of the MATLAB. It provides the user a Graphical User Interface (GUI) which allows the end-user to configure the system parameters with great ease before initializing a simulation without actually modifying the code. Using Simulink, the system model can be organized into a hierarchy and hence, the model can be built using both top-down and bottom-up approaches.

Different algorithms have been proposed for the functionalities of the CAC framework which are summarised in the following subsections.

7.1.1 Admission Control Procedure

Two admission control algorithms are presented which are tested against each other in the given system. The non-adaptive admission control algorithm performs by checking the available capacity of the given link to decide if the new connection request can be admitted in the system. It is a simple admission control algorithm based on the equivalent capacity approach presented in the literature [7]. The non-adaptive admission control algorithm works as a basic admission control algorithm. It does not consider the specific characteristics of the system in consideration such as the variable link capacity which may lead to the deterioration in the system performance in terms of high blocking ratio and low resource utilization. As a result, an adaptive admission control algorithm has been proposed to extend the non-adaptive algorithm by taking into account the characteristics of the radio channel and the supported type of mobile terminals with different antenna sizes. The algorithm not only takes into account the traffic characteristics of the requesting connection and the resources available in the system while making the admission decision, it also adapts to the changes in the link condition experienced by each user due to the weather conditions, mobility and other factors. In addition, it takes into account the class of the MT which affects the range of data rates supported by a given user. The results have shown that the adaptive admission control algorithm performs better than the non-adaptive in terms of the number of admitted connections, blocking ratio, channel utilization efficiency and user data throughput efficiency. For the multicast traffic, the adaptive admission control algorithm works similar to that for the unicast traffic except that the data rate supported by a multicast group is re-evaluated every time a new multicast connection joins an existing multicast group. The data rate at which the multicast content is sent in a particular multicast group depends on the terminal of the lowest capability, in terms of the supported data rate, within that multicast group. This allows the multicast data to be received by all the MTs belonging to a particular multicast group. The simulation were set for a single class of multicast traffic and mixed classes of multicast traffic. The results have shown that there is no effect of the number of MTs joining a multicast group on the channel utilization efficiency. It remains constant as long as there is at least one MT in the multicast group. The change in the channel utilization efficiency with varying MT classes is also presented in the results.

7.1.2 Retune Mechanism

The purpose of the retune mechanism in conjunction with the admission control procedure is to improve the overall system performance by allowing the user to move from one forward subband to another if the requested link is congested in order to admit the user on the new link. The retune process is triggered when the requesting connection cannot be admitted on the given link. A greedy retune procedure is proposed which aims at reducing the number of connections retuned from one link to another by selecting the connections with the highest resource usage to retune. The results indicate an improvement in the system performance with the retune procedure enabled as compared to the admission control procedure with no retune.

Another factor which affects the retuning is the subband selection method. Two subband selection methods have been proposed in this thesis: *MinConnSubSel*, and *Random*. The *MinConnSubSel* method selects the forward subband with the minimum number of active connections from the list of available subbbands. Such selection performs a simple load balancing between the subbands. On the other hand, the *Random* method randomly selects a forward subband from the list of available subbands. The results have established that the system performs better with the *MinConnSubSel* than the *Random* subband selection method in terms of a reduction in the number of retunes, and a lower blocking ratio.

7.1.3 Pre-emption Procedure

The purpose of the pre-emption procedure is to maintain the performance of the system and to support the connection precedence and pre-emption when the network is in a congested state. Three pre-emption control algorithms have been proposed in this thesis; Greedy, SubSetSum and Fuzzy. The Greedy pre-emption algorithm performs by pre-empting the connections with the lowest resource usage. The aim of this algorithm is to reduce the bandwidth which is released by pre-empting the connections. However, in doing so, it greedily pre-empts more connections. The SubSetSum algorithm is proposed to minimize the number of connections pre-empted while minimizing the bandwidth released by the pre-empting connections. Hence, it is an improvement over the Greedy algorithm. The algorithm performs by dropping one or more lower priority connections, such that the total resource consumption of these connections is just enough to accommodate the new connection request. This enables high priority connection requests to be admitted without dropping more than the necessary existing lower priority connections, thus minimizing the bandwidth released by the pre-emptable connections and also reducing the number of connections pre-empted. The simulation results demonstrate that the SubSetSum pre-emption algorithm provides better performance in terms of lower blocking and dropping ratios compared to the Greedy pre-emption algorithm. Hence, higher numbers of connections are successful using the SubSetSum algorithm. Also, the data size released by the pre-emptable connections is lower for the SubSetSum algorithm. However, the computation time is much higher than the Greedy pre-emption algorithm. The SubSetSum algorithm is mathematically complex and requires more computation time. Hence, a further improvement over the SubSetSum pre-emption algorithm has been proposed by the use of an intelligent Fuzzy pre-emption algorithm. It makes use of the expert system knowledge to provide a better system performance as compared to the Greedy and the SubSetSum pre-emption algorithms. The results indicate lower blocking and dropping ratios, a higher number of successful connections, and a shorter computation time.

Another mechanism in relation to the pre-emption has also been proposed which is called as Connection Inactive Mechanism (CIM) which is used to further enhance the performance of the system. Each pre-emption algorithm can be used with and without CIM leading to six potential solutions to the pre-emption algorithm. In contrast with the simple pre-emption procedure where the selected pre-emptable connections are simply dropped, this mechanism changes the state of the pre-empted connection to be '*Non-Active'* for a period of time which varies with the type of the connection. Under the '*Non-Active'* state, the resources used by the selected connections are released and at the end of the elapsed time, the pre-empted connections request for resources again. In the case that the required resources cannot be granted at the time, the '*Non-Active*' connection is finally dropped else it is admitted. The results have shown an improvement in the system performance by using the pre-emption algorithm with CIM as opposed to the simple pre-emption algorithm in terms of lower dropping ratio, a higher number of successful connections, lower pre-emptable data size, and a higher revenue generation.

7.2 Future Work

The work presented in this thesis provides the foundation which may be further extended and studied. Some of these potential research directions are discussed in following subsections.

7.2.1 Admission Control in the return link

The current CAC framework provides the admission control in the forward link from the satellite to the terminals. The physical frames are transferred in using TDM every 80ms. The admission control predicts whether there are enough resources for the connection in the 80ms frame based on the type of the traffic, the source traffic characteristics, the link condition etc. The current implementation assumes that a connection admitted in the forward link can obtain resources in the return link and hence, an admission control procedure is not necessary for the return link. In the future, it may be interesting to provide admission control in the return link also. The return link is based on TDMA scheme, where the return frames consist of different time slots. An analysis similar to the one presented in this thesis, can be obtained for the return link admission control.

7.2.2 Fuzzy Inference Engine improvement

In section 6.5 while comparing the three proposed pre-emption algorithms; Greedy, SubSetSum and Fuzzy, the Fuzzy pre-emption algorithm performed better than the other two for all performance metrics except for the total *pre-*

emptable bandwidth. The results show that the Fuzzy algorithm leads to the highest *pre-emptable bandwidth* followed by the Greedy and the SubSetSum algorithms. This is an expected result as the SubsetSum algorithm tries to minimize the *pre-emptable bandwidth* using the SubSetSum problem, in an optimum way, by which the connections are pre-empted based on the resource utilized such that the pre-empted resource is just enough to admit the new higher priority connection. Similarly, the Greedy algorithm, also aims to keep the *pre-emptable bandwidth* at a minimum value. Although the *pre-emptable bandwidth* performance metric is better for the SubSetSum and the Greedy algorithm, the Fuzzy algorithm provides a more flexible way to change the behaviour by using the different IF-THEN rules. It would be interesting to analyse and design different sets of IF-THEN rules in order to further enhance the performance of the algorithm. Also, it would be beneficial to see if there are any more improvements by adding another input membership function such as the link condition of the user along with other inputs, then designing a new set of IF-THEN rules to see the impact on the system performance.

7.2.3 Enhancing the load balancing functionality

In Section 4.2.2.1.1, the *Subband Selector* functional block of the CAC Processor selects the forward subband for the new connection. One of the proposed methods for the subband selection, *MinConnSubSel,* performs by selecting the forward subband with the minimum number of active connections. This results in a simple load balancing between the forward subbands. The results presented in this thesis have shown that the system performs better with the *MinConnSubSel* subband selection method than with the *Random* subband selection method. It may be interesting to provide a separate load balancing functionality in the CAC framework which is triggered by the CAC processor whenever a new connection request comes. A fuzzy expert system can be used to implement the load balancing function which will balance the resources across two or more subbands.

7.2.4 Incorporation of the CAC framework in a complete RRM framework

In this thesis work, the CAC framework has been designed and studied as a standalone entity which predicts the resources used by the connections based on the information such as the traffic source characteristics, the link condition, and the class of the MT to decide whether to admit or reject the connection. In the future, it may be possible to incorporate the remaining RRM functionalities such as the Scheduler, the Resource Allocator and the Handover Control to build a complete RRM framework for a S-UMTS system. The CAC can provide the QoS requirements of the established connections to the scheduler. The scheduler then assigns the radio resources to the connections based on their QoS requirements. The resource allocator allocates the resources to each connection and the handover control function will be responsible for the handover decision between the BCts.

REFERENCES

[1] M. R. Chartrand, *Satellite Communications for the nonspecialist*, 1st ed.: SPIE-The International society for optical engineering, 2004.

[2] M.Richharia, *Mobile Satellite Communications: Principles and Trends*, 1st ed.: Pearson Education, 2001.

[3] B. R. Albert, *Introduction to Satellite Communication* 2nd ed.: Artech House Space Publication Series, 1998.

[4] B. G. Evans, *Satellite Communication Systems*, 3rd ed.: The Institute of Electrical Engineers, 2000.

[5] Y. Zhang, D. D. Lucia, B. Ryu, and S. K. Dao, "Satellite Communications in the Global Internet: Issues, Pitfalls, and Potential," in *INET*, Kuala Lumpur, Malaysia, June 1997.

[6] G. Giambene, *Resource Management in Satellite Networks: Optimization and Cross-Layer Design*, 1st ed.: Springer Science, 2007.

[7] R. Guerin, H. Ahmadi, and M. Naghshineh, "Equivalent capacity and its application to bandwidth allocation in high-speed networks," in *IEEE Journal in Selected Areas in Communications*, vol. 9, pp. 968-981, September 1991.

[8] M. A. Rahin and M. Kara, "Call Admission Control Algorithms in ATM networks: A Performance Comparison and Research Directions," in *Research report, University of Leeds*, Feburary 1998.

[9] H. G.Perros and K. M.Elsayed, "Call Admission Control Schemes: A Review," in *IEEE Communications Magazine*, vol. 34, pp. 82-91, November 1996.

[10] Y. Tian, "A Survey on Connection Admission Control in ATM Networks," in *Proceedings of DePaulCTI Research Symposium*, Chicago, IL, Novemeber 1999.

[11] A.W.Berger and W.Whitt, "Extending the effective bandwidth concept to network with priority classes," in *IEEE Communications magazine*, vol. 36, Issue 8, pp. 78-83, 1998.

[12] Z.Fan and P.Mars, "Effective bandwidth approach to connection admission control for multimedia traffic in ATM networks," in *Electronics Letters*, vol. 32, Issue 16, pp. 1438-1439, 1996.

[13] G.Rosanski and M.Bromirski, "On the effectiveness of effective bandwidth for call admission control in military ATM networks," in *IEEE Military Communications Conference*, vol. 3, pp. 1507-1511, 1997.

[14] S.Jamin, P. B. Danzig, S.J.Shenker, and L.Zhang, "A measurement-based admission control algorithm for integrated services packet network," in *EEE/ACM Transactions on Networking*, Feburary 1997.

[15] S. Georgoulas, P. Trimintzios, G. Pavlou, and K. H. Ho, "Measurement Based Admission Control for Real-time Traffic in IP Differentiated Services Networks," in *ICT Proceedings*, Cape Town, South Africa, May 2005.

[16] M. H. Ahmed, "Call Admission Control in Wireless Networks: A Comprehensive Survey," in *IEEE Communications Surveys & Tutorials*, vol. 7, pp. 49-68, 2005.

[17] D.Hong and S.S.Rappaport, "Traffic model and performance analysis for cellular mobile radio telephone systems with prioritized and nonprioritized handoff procedures," in *IEEE Transactions on Vehicular Technology*, vol. 35, Issue. 8, pp. 77-92, August 1986.

[18] C.J.Chang, T.T.Su, and Y.Y.Chiang, "Analysis of a cutoff priority cellular radio system with finite queueing and reneging/dropping," in *IEEE/ACM Transactions on Networking*, vol. 2, Issue. 4, pp. 166-175, April 1994.

[19] G.Min, X.Jin, and S.R.Velentzas, "Performance analysis of the guard channel scheme with self-similar call arrivals in wireless mobile networks," in *IEEE Global Telecommunications Conference* pp. 1-5, December 2008.

[20] R.Ramjee, D.Towsley, and R.Nagarajan, "On optimal call admission control in cellular networks," in *ACM Wireless Networks*, vol. 3, pp. 29-41, 1997.

[21] S.Kalyanasundaram, E.K.P.Chong, and N.B.Shroff, "Admission control schemes to provide class-level QoS in multiservice networks," in *International Journal of Computer and Telecommunications Networking*, vol. 35, pp. 2-3, Feburary 2001.

[22] K. Yu, X. Wang, S. Sun, L. Zhang, and X. Wu, "A statistical connection admission control for multiservice IEEE 802.16 network," in *69th IEEE Vehicular Technology Conference*, pp. 1-5, April 2009.

[23] D.Mitra, M.I.Reiman, and J.Wang, "Robust Dynamic admission control for unified cell and call QoS in statistical multiplexers," in *IEEE Journal on Selected Areas in Communications*, vol. 16, Issue. 5, June 1998.

[24] J.Yao, J.W.Mark, T.C.Wong, Y.H.Chew, K.M.Lye, and K.C.Chua, "Virtual partitioning resource allocation for multiclass traffic in cellular systems with QoS constraints," in *IEEE Transactions on Vehicular Technology*, vol. 53, Issue. 3, May 2004.

[25] J. Ni, D. H. K. Tsang, S. Tatikonda, and B. Bensaou, "Threshold and reservation based call admission control policies for multi service resource sharing systems," in *Annual Joint Conference of the IEEE Computer and Communications Socities*, vol. 2, pp. 773-783, March 2005.

[26] O. Yu and S. Khanvilkar, "QoS Provisioning over GPRS Wireless Mobile Links," in *IEEE Wireless Communications and Networking Conference*, vol. 1, pp. 322-326, 2002.

[27] S. K.Das, S. K.Sen, K. Basu, and H. Lin, "A Framework for Bandwidth and Degradation and Call Admission Control Schemes for Multiclass Traffic in Next-Geneartion Wireless Networks," in *IEEE Selected Areas in Communications*, vol. 21, pp. 1790- 1802, December 2003.

[28] N. Nasser and H. Hassanein, "Multi-Class Bandwidth Allocation Policy for 3G Wireless Networks," in *Local Computer Networks Proceedings*, pp. 203-209, October 2003.

[29] H. Wu, L. Li, B. Li, L. Yin, I. Chlamtac, and B. Li, "On handoff performance for an integrated voice/data cellular system," in *IEEE*

International Symposium on Personal, Indoor and Mobile Radio Communications, vol. 5, pp. 2180-2184, December 2002.

[30] G.Choudhury, K.Leung, and W.Whitt, "An algorithm to compute blocking probabilities in multi-rate multi-class multi-resources loss models," in *Advances in Applied Probability*, vol. 27, pp. 1104-1143, 1995.

[31] R. Akl and A. Parvez, "Global versus Local Call Admission Control in CDMA Cellular Networks," in *Proceedings of CITSA 04: Communications, Information and Control Systems, Technologies and Applications*, vol. 2, pp. 283-288, July 2004.

[32] Y. Iraqi and R. Boutaba, "A Novel Distributed Call Admission Control for Wireless Mobile Multimedia Networks," in *Third ACM International Workshop on Wireless Mobile Multimedia* pp. 21-27, 2000.

[33] M.Ghaderi and R. Boutaba, "Call admission control in mobile cellular networks a comprehensive survey," in *Journal on Wireless Communications and Mobile Computing*, vol. 6, Issue. 1, Feburary 2006.

[34] N.Nasser and H.Hassanein, "Optimized bandwidth allocation with fairness and service differentiation in multimedia wireless networks," in *Journal of Wireless Communications and Mobile Computing*, vol. 8, Issue. 4, May 2008.

[35] Y.Xiao, C.L.P.Chen, and Y.Wang, "Optimal admission control for multiclass of wirelesss adaptive multimedia services," in *IEICE Transactions on Communications*, pp. 795-804, April 2001.

[36] D. Karabudak, C.-C. Hung, and B. Bing, "A Call Admission Control Scheme using Genetic Algorithms," in *Proceedings of the ACM symposium on Applied Computing*, Nicosia, Cyprus, pp. 1151-1158, 2004.

[37] D. Todinca, S. Holban, P. Perry, and J. Murphy, "FuzzyLogic Based Admission Control for GPRS/EGPRS Networks," in *Proceedings of 6th International Conference on Technical Informatics, CONTI 2004, Transactions on Automatic Control & Control Science*, Timisoara, Romania, vol. 4, pp. 205-210, May 2004.

[38] J. Ye, X. Shen, and J. W. Mark, "Call Admission Control in Wideband CDMA Cellular Networks by Using Fuzzy Logic," in *IEEE Transactions on Mobile Computing*, vol. 4, pp. 129-141, April 2005.

[39] L. Barolli, M. Durresi, K. Sugita, A. Durresi, and A. Koyama, "A CAC Scheme for multimedia Applications Based on Fuzzy Logic," in *19th International Conference on Advanced Information Networking and Applications (AINA)*, vol. 1, pp. 473-478, March 2005.

[40] R.-G. Cheng, C.-J. Chang, and L.-F. Lin, "A QoS-provisioning neural fuzzy connection admission controller for multimedia high-speed networks," in *IEEE/ACM Transactions on Networking*, vol. 7, pp. 111-121, Feburary 1999.

[41] X. Yang and J. Bigham, "A Call Admission Control Scheme using NeuroEvolution Algorithm in Cellular Networks," in *20th International Joint Conference on Artificial Intelligence* Hyderabad, India, January 2007.

[42] Y.Zhang, S.Dai, C.Zhou, and L.Li, "Bandwidth allocation scheme and CAC for adaptive multimedia services in wireless cellular networks," in *IEEE International Conference on Communications*, pp. 346-352, May 2005.

[43] F.Yu, V.W.S.Wong, and V.C.M.Leung, "A new QoS provisioning method for adaptive multimedia in cellular wireless networks," in *IEEE Transactions on Vehicular Technology*, pp. 1899-1909, May 2008.

[44] L.Castanet, A.B.Alamanac, and M.Bousquet, "Interference and Fade Mitigation Techniques for Ka and Q/V Band Satellite Communication Systems," in *Proceedings COST-280*, 2003.

[45] E. Alberty, S. Defever, and C. Moreau, "Adaptive Coding and Modulation for the DVB-S2 Standard Interactive Applications: Capacity Assessment and Key System Issues," in *IEEE Wireless Communications*, vol. 14, pp. 61-69, August 2007.

[46] H.Koraitim and S.Tohme, "Resource Allocation and Connection Admission Control in Satellite Networks," in *IEEE Journal on Selected Areas in Communications*, vol. 17, pp. 360-372, Feburary 1999.

[47] Y.M.Jang, "Estimation and Prediction-Based Connection Admission Control in Broadband Satellite Systems," in *ETRI Journal*, vol. 22, pp. 40-50, December 2000.

[48] F.D.Priscoli and A.Pietrabissa, "Control-based Connection Admission Control and Downlink Congestion Control procedures for satellite networks," in *Journal of Franklin Institute*, vol. 346, Issue. 9, pp. 923-944, November 2009.

[49] M.Marchese and M.Mongelli, "Adaptive Call Admission and Bandwidth Control in DVB-RCS Systems," in *IEEE International Conference on Communications*, Kyoto, pp. 1-5, June 2011.

[50] Y.Qian, R.Q.Hu, and C.Rosenberg, "Integrated Connection Admission Control and Bandwidth on Demand Algorithm for a Broadband Satellite Network with Heterogeneous Traffic," in *IEICE Transactions on Communications*, vol. 3, pp. 895-905, 2006.

[51] Y.H.Aoul, A.Nafaa, and A.Mehaoua, "Buffer Occupancy- Based CAC in Converged IP and Broadcasting Networks," in *IEEE International Conference on Communications*, Glasgow, pp. 19-25, June 2007.

[52] Y.H.Aoul and T.Taleb, "An Adaptive Fuzzy based CAC scheme for uplink and downlink congestion control in converged IP and DVB-S2 networks," in *IEEE Transactions on Wireless Communications*, vol. 8, Issue. 2, pp. 816-825, Feburary 2009.

[53] F.Alagoz, B.R.Vojcic, D.Walters, A.AlRustamani, and R.L.PickHoltz, "Fixed versus Adaptive Admission Control in Direct Broadcast Satellite Networks with Return Channel Systems," in *IEEE journal of Selected Areas in Communications*, vol. 22, Issue. 2, pp. 238-249, Feburary 2004.

[54] J.Swiko and I.Rubin, "Connection Admisison Control for capacity varying networks with stochastic capacity change times," in *IEEE/ACM Transactions on Networking* vol. 9, Issue. 3, pp. 351-360, June 2001.

[55] P. Pech, M. Bousquet, and J. Radzik, "A Ka band OBP satellite system simulation platform involving fade mitigation techniques with channel attenuation time series and static radar images of rain cell

evolution," in *21st AIAA International Communications Satellite Systems Conference*, April 2003.

[56] D.K.Petraki, M.P.Anastasopoulos, and P.G.Cottis, "Call Admission Control in Satellite networks under rain fading," in *IEEE Communications Letters*, vol. 12, Issue. 5, May 2008.

[57] M. Anastasopoulos, D. Petraki, A. Vasilakos, and P. Cottis, "Call admission control scheme for multiclass services under rain fading for satellite networks," in *IEEE Transactions on Wireless Communications*, vol. 8, Issue. 5, May 2009.

[58] F. Davoli, M. Marchese, and M. Mongelli, "Optimal resource allocation in satellite networks: certainty equivalent approach versus sensitivity estimation algorithms," in *International Journal of Communication Systems*, vol. 18, Issue. 1 pp. 3-36, Feburary 2005.

[59] N. Celandroni, F. Davoli, and E. Ferro, "Static and Dynamic Resource Allocation in a Multiservice Satellite Network with Fading," in *International Journal of Satellite Communications and Networks*, vol. 21, pp. 469-487, 2003.

[60] N. Celandroni, F. Davoli, E. Ferro, and A. Gotta, "Complete partitioning allocation policies in a rain faded satellite environment," in *Technical report*, 2003.

[61] K.Kordyboch and S.Nousianinen, "Radio Resource Management in WCDMA-based Networks in Emergency Situations," in *Proceedings of 18th International Teletraffic Congress*, Berlin, Germany, August 2003.

[62] W.Shen and Q.An.Zeng, "Resource Allocation Schemes in integrated heterogeneous wireless and mobile networks," in *Journal of Networks*, vol. 2, Issue. 5, September 2007.

[63] E.Salvadori and R. Battiti, "Quality of Service in IP over WDM: considering both service differentiation and transmission quality," in *IEEE International Conference on Communications*, pp. 1836-1840, June 2004.

[64] J.Zhou and C.Beard, "Comparison of combined pre-emption and queuing schemes for admission control in a cellular emergency network," in *IEEE Wireless Communications and Networking Conference*, pp. 122-128, April 2006.

[65] J.Zhou and C.Beard, "A controlled Pre-emption scheme for emergency applications in cellular networks," in *IEEE Transactions on Vehicular Technology*, vol. 58, Issue. 7, pp. 3753-3764, September 2009.

[66] G. Maral and M. Bousquet, *Satellite Communications systems-System Techniques and Technologies*, 5th ed.: John Wiley and Sons Ltd., 2009.

[67] A. Richardson, *WCDMA design handbook* Cambridge University Press, April 2005.

[68] T. Ivankovic, "Support of multimedia broadcast/multicast service in UMTS networks " in *Proceedings of the 8th International Conference on Telecommunications* vol. 1, pp. 91-98, June 2005

[69] (2011, 10th May). *Inmarsat Company Website*. Available: http://www.inmarsat.com/Downloads/English/BGAN/Collateral/bgan_overview_brochure_EN.pdf?language=EN&textonly=False.

[70] P. Febvre, "Multicast-enabling a 3GPP-based Mobile Satellite Network," in *The Institute of Engineering and Technology Conference on Wireless Broadband Conference* London, pp. 135-154, May 2006.

[71] (2011, 2nd September). *ESA Website*. Available: http://telecom.esa.int/telecom/www/object/index.cfm?fobjectid=11369.

[72] M.Karaliopoulos, K.Narenthiran, and B.Evans, "Satellite radio interface and radio resource management strategy for the delivery of Multicast/Broadcast services via an integrated satellite-terrestrial system," in *IEEE Communications Magazine*, vol. 42, pp. 108-117, September 2004.

[73] Inmarsat Confidential BGAN System Definition Manual. *System Architecture and Interfaces,* Volume 1, Chapter 3.

[74] ETSI, "GEO- Mobile Radio Interface Specifications; GMR-2 General System Description," TS 101 377-01-03, October 1999.

[75] G. Jilg, "Inmarsat - products and strategies," in *Workshop on Satellites in IP and Multimedia*, December 2002.

[76] R.E.Sheriff and Y.F.Hu, *Mobile Satellite Communication Networks* John Wiley and Sons Ltd., 2001.

[77] G.E.Corazza, *Digital Satellite Communications*: Springer, May 2007.

[78] L.Harte, R.Dreher, S.Kellog, and T.Schaffnit, *Comprehensive guide to wireless technologies: Cellular, PCS,Paging, SMR and Satellite*: Althos, 2000.

[79] A.Jamilipour, *Low Earth Orbital Satellites for Personal Communication Networks*, 1st ed.: Artech House, 1997.

[80] F. J. Hens and J. M. Caballero, *Triple Play: Building the converged network for IP, VoIP and IPTV*: John Wiley and Sons Ltd., 2008.

[81] ETSI, "Satellite Earth Station and Systems (SES); Satellite Component of UMTS/IMT2000; G-family; Part1: Physical Channels and Mapping of Transport Channels into Physical Channels," TS 101 851-1.

[82] ETSI, "Satellite Earth Station and Systems (SES); Satellite Component of UMTS/IMT2000; G-family; Part2: Multiplexing and Channel Coding," TS 101 851-2.

[83] ETSI, "Technical Specification Group Services and System Aspects, Iu Principles," 3G TR 23.930.

[84] Inmarsat Confidential BGAN System Definition Manual. *Satellite Air Interface Overview,* Volume 1, Chapter 6.

[85] Inmarsat Confidential BGAN System Definition Manual. *Adaptation Layer,* Volume 4, Chapter 2.

[86] Inmarsat Confidential BGAN System Definition Manual. *Bearer Connection Layer Operation,* Version 1.0.

[87] Inmarsat Confidential BGAN System Definition Manual. *Bearer Control Layer Operation,* Volume 4, Chapter 4.

[88] Inmarsat Confidential BGAN System Definition Manual. *Physical Layer Interface,* Volume 2, Chapter 5.

[89] Inmarsat BGAN Extension Phase 2 Programme. *RRM Requirements and Constraints,* Version 1.0.

[90] 3GPP, "End-to-End QoS Concept and Architecture (Release 6)," TS 23.007, June 2005.

[91] N.Naseer and H.Hassanein, "Multi-Class bandwidth allocation policy for 3G wireless networks," in *28th Annual IEEE International Conference on local computer networks*, p. 203, 2003.

[92] J.Banks, J.S.Carson, and B.L.Nelson, *Discrete Event System Simulation*, 2nd ed., 1995.

[93] N.O.L.Ng and C.K.Tham, "Connection Admission control of ATM network using integrated MLP and fuzzy controllers," in *Computer Networks*, vol. 32, pp. 61-79, May 2000.

[94] M.Qian, Y.Huang, J.Shi, Y.Yuan, L.Tian, and E.Dutkiewicz, "A Novel Radio Admission Control Scheme for Multiclass Services in LTE systems," in *IEEE GLOBECOM*, pp. 1-6, 2009.

[95] H.Wang, D.Prasad, O.Teyeb, and H.P.Schwefel, "Performance Enhancements of UMTS networks using end-to-end QoS provisioning," in *Wireless Personal Multimedia Communications*, pp. 18-22, September 2005.

[96] P.M.L.Chan, R.E.Sheriff, Y.F.Hu, P.Conforto, and C.Tocci, "Mobility management incorporating fuzzy logic for a heterogenous IP enviornment," in *IEEE Communications Magazine*, vol. 39, pp. 42-51, December 2001.

[97] G.J.Klir and B.Yuan, *Fuzzy sets and fuzzy logic: Theory and Applications* New York:Prentice-Hall, Inc, 1994.

[98] E.H.Mamdani and S.Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," in *International Journal of Man-Machine Studies*, vol. 7, Issue. 2, pp. 1-13, 1975.

[99] T. J. Ross, *Fuzzy Logic with Engineering Applications* 2nd ed.: John Wiley and Sons Ltd., 2004.

[100] (2011, 22nd August). *Mathworks Inc.* Available: http://www.mathworks.com/products.