Contribution of Quality of Experience to optimize multimedia services : the case study of video streaming and VoIP

Muhammad Sajid Mushtaq

École Doctorale MSTIC

Thèse de doctorat
Spécialité: Informatique et Réseaux

Muhammad Sajid MUSHTAQ

soutenue le 15/04/2015

Titre:

Apport de la Qualité de l'Expérience dans l'optimisation de services multimédia: Application à la diffusion de la vidéo et à la VoIP

Contribution of Quality of Experience to optimize multimedia services: The case study of video streaming and VoIP

Thèse dirigée par:

Directeur de Thèse: Prof. Abdelhamid MELLOUK
Co-encadrant: Dr. Brice AUGUSTIN

Jury:

Examinateurs:

Prof. Andre-Luc Beylot, IRIT, ENSEEIHT, Toulouse, France
Prof. Scott Fowler, Linköping University, Sweden

Rapporteurs:

Prof. Ana Rosa Cavali, Télécom ParisSud, Paris, France
Prof. Christophe Chassot, LAAS-CNRS, Toulouse, France

Titre: Apport de la Qualité de l'Expérience dans l'optimisation de services multimédia: Application à la diffusion de la vidéo et à la VoIP

Résumé: L'émergence et la croissance rapide des services multimédia dans les réseaux IP ont créé de nouveaux défis pour les fournisseurs de services réseau, qui, au-delà de la Qualité de Service (QoS) issue des paramètres techniques de leur réseau, doivent aussi garantir la meilleure qualité de perception utilisateur (Quality of Experience, QoE) dans des réseaux variés avec différentes technologies d'accès. Habituellement, différentes méthodes et techniques sont utilisées pour prédire le niveau de satisfaction de l'utilisateur, en analysant l'effet combiné de multiples facteurs. Dans cette thèse, nous nous intéressons à la commande du réseau en intégrant à la fois des aspects qualitatifs (perception du niveau de satisfaction de l'usager) et quantitatifs (mesure de paramètres réseau) dans l'objectif de développer des mécanismes capables, à la fois, de s'adapter à la variabilité des mesures collectées et d'améliorer la qualité de perception. Pour ce faire, nous avons étudié le cas de deux services multimédia populaires, qui sont : le streaming vidéo, et la voix sur IP (VoIP). Nous investiguons la QoE utilisateur de ces services selon trois aspects : (1) les méthodologies d'évaluation subjective de la QoE, dans le cadre d'un service vidéo, (2) les techniques d'adaptation de flux vidéo pour garantir un certain niveau de QoE, et (3) les méthodes d'allocation de ressource, tenant compte de la QoE tout en économisant l'énergie, dans le cadre d'un service de VoIP (LTE-A). Nous présentons d'abord deux méthodes pour récolter des jeux de données relatifs à la QoE. Nous utilisons ensuite ces jeux de données (issus des campagnes d'évaluation subjective que nous avons menées) pour comprendre l'influence de différents paramètres (réseau, terminal, profil utilisateur) sur la perception d'un utilisateur d'un service vidéo. Nous proposons ensuite un algorithme de streaming vidéo adaptatif, implémenté dans un client HTTP, et dont le but est d'assurer un certain niveau de QoE et le comparons à l'état de l'art. Notre algorithme tient compte de trois paramètres de QoS (bande passante, taille de mémoires tampons de réception et taux de pertes de paquets) et sélectionne dynamiquement la qualité vidéo appropriée en fonction des conditions du réseau et des propriétés du terminal de l'utilisateur. Enfin, nous proposons QEPEM (QoE Power Efficient Method), un algorithme d'ordonnancement basé sur la QoE, dans le cadre d'un réseau sans fil LTE, en nous intéressant à une allocation dynamique des ressources radio en tenant compte de la consommation énergétique.

Mots-clés: Qualité d'Expérience, Services Multimédia, Méthodes subjectives, Crowdsourcing, Plateforme de test, Méthodes de streaming adaptatif, LTE-A, DRX, Consommation énergétique.

Unité de recherche: Laboratoire Images, Signaux et Systèmes Intelligents (LISSI), EA 3956, UPEC.

Title: Contribution of Quality of Experience to optimize multimedia services: The case study of video streaming and VoIP.

Abstract: The emerging and fast growth of multimedia services have created new challenges for network service providers in order to guarantee the best user's Quality of Experience (QoE) in diverse networks with distinctive access technologies. Usually, various methods and techniques are used to predict the user satisfaction level by studying the combined impact of numerous factors. In this thesis, we consider two important multimedia services to evaluate the user perception, which are: video streaming service, and VoIP. This study investigates user's QoE that follows three directions: (1) methodologies for subjective QoE assessment of video services, (2) regulating user's QoE using video a rate adaptive algorithm, and (3) QoE-based power efficient resource allocation methods for Long Term Evaluation-Advanced (LTE-A) for VoIP. Initially, we describe two subjective methods to collect the dataset for assessing the user's QoE. The subjectively collected dataset is used to investigate the influence of different parameters (e.g. QoS, video types, user profile, etc.) on user satisfaction while using the video services. Later, we propose a client-based HTTP rate adaptive video streaming algorithm over TCP protocol to regulate the user's QoE. The proposed method considers three Quality of Service (QoS) parameters that govern the user perception, which are: Bandwidth, Buffer, and dropped Frame rate (BBF). The BBF method dynamically selects the suitable video quality according to network conditions and user's device properties. Lastly, we propose a QoE driven downlink scheduling method, i.e. QoE Power Efficient Method (QEPEM) for LTE-A. It efficiently allocates the radio resources, and optimizes the use of User Equipment (UE) power utilizing the Discontinuous Reception (DRX) method in LTE-A.

Keywords: Quality of Experience, Multimedia Service, Subjective Methodologies, Testbed, Crowdsourcing, Adaptive Streaming Method, Scheduling, Power, Long Term Evolution-Advanced (LTE-A), Discontinuous Reception (DRX).

Research Unit: Images, Signals and Intelligent Systems (LISSI) Laboratory, EA 3956, UPEC.

Table of Contents

List of Figures

List of Tables

Acknowledgements

"He who does not thank people, does not thank ALLAH."

The Messenger of Allah (peace be upon him).

First of all, I humbly thank the Almighty **ALLAH** (God), the Merciful and the Beneficent, Who bless me with knowledge, health, thoughts and cooperative people that enable me to achieve this research. It is a pleasure to thank all those people who made this dissertation possible.

I would like to pay sincerest thanks to my supervisor, Prof. Abdelhamid Mellouk, whose continuous guidance, encouragement, and support from the start to the end of this research work, enabled me to develop a better understanding of research subject, and provide all necessary knowledge to complete this research work. I would like to say that it is excellent experienced to work with such a great researcher, who is constantly encouraging and willing to help me.

I am extremely grateful to my co-supervisor Dr. Brice Augustin, for his valuable advice and assistance throughout my research work, and his comments greatly improved the content of the papers from which this thesis has been partly extracted.

I would like to thank Dr. Scott Fowler for his assistance, guidance, and encouragement throughout my work have been extremely helpful.

Many thanks to all the colleagues and friends at the LiSSi Laboratory, who supported and helped me with their capabilities and valuable discussions to complete this research work.

Last but not least, I am extremely grateful to my wife and family members for their support, and continuous encouragement to achieve all this.

Dedication

I would like to dedicate my dissertation work to my wife and my son "Muaaz". A special gratitude to my family members and loving parents, whose encourage me to complete my thesis work.

Acronyms

2D	Two-dimensional
3D	Three-dimensional
3GPP	3rd Generation Partnership Project
4G	Four Generation
5G	Fifth Generation
BBF	Bandwidth, Buffer, dropped Frame rate
BCQI	Best Channel Quality Indicator
BLER	Block Error Ratio
B-frame	Bidirectional Frame
CBR	Constant Bit Rate
CDF	Cumulative Distribution Function
CDMA-HDR	Code Division Multiple Access High Data Rate
CDN	Content Delivery Network
CPU	Central Processing Unit
CQI	Channel Quality Indicator
DASH	Dynamic Adaptive Streaming over HTTP
DT	Decision Tree
DRX	Discontinuous Reception
eNodeB	Evolving NodeB
FFT	Fast Fourier Transform
FP	False Positive
GBR	Guaranteed Bit Rate
GOP	Group of Picture

HD	High Definition
HDS	HTTP Dynamic Streaming
HLS	HTTP Live Streaming
HSDPA	High Speed Downlink Packet Access
HTTP	Hyper Text Transfer Protocol
ICIC	Inter-cell interference coordination
IDR	Instantaneous Decoding Refresh
I-frame	Information Frame
IP	Internet Protocol
IPTV	Internet Protocol Television
ISI	Inter Symbol Interference
ISO	International Organization for Standardization
ITU-T	International Telecommunication Union-Telecommunication
k-NN	k-Nearest Neighbours
LTE	Long Term Evaluation
LTE-A	Long Term Evaluation-Advanced
MCS	Modulation and Coding Scheme
ML	Machine Learning
M-LWDF	Modified Largest Weighted Delay First
MME	Mobility Management Entity
MMS	Microsoft Media Server
MOS	Mean Opinion Score
MPEG	Moving Picture Experts Group
MSS	Microsoft Silverlight Smooth Streaming
MTC	Machine Type Communication
NAT	Network Address Translation
NAL	Network Abstraction Level
NB	Naive Bayes
NetEm	Network Emulator
NGN	Next Generation Networks
NNT	Neural Networks

NRT	Non-Real Time
NS-2	Network Simulator-2
OFDMA	Orthogonal Frequency Division Multiple Access
OSMF	Open Source Media Framework
OTT	Over-the-Top
P2P	Peer-to-Peer
PC	Personal Computers
PDCCH	Physical Downlink Control Channel
PESQ	Perception Evaluation of Speech Quality
P-frame	Predicted Frame
PF	Proportional Faire
PLR	Packet Loss Rate
PPS	Picture Parameter Set
QoE	Quality of Experience
QEPEM	QoE Power Efficient Method
QoS	Quality of Service
RB	Resource Block
RF	Random Forest
R-factor	Rating Factor
RoI	Region of Interest
RR	Round Robin
RRC	Radio Resource Control
RRM	Radio Resource Management
RT	Real Time
RTP	Real Time Protocol
RTMP	Real Time Messaging Protocol
RTSP	Real Time Streaming Protocol
RTT	Round Trip Time
SFT	Segment Fetch Time
S-GW	Serving Gateway
SINR	Signal to Interference and Noise Ratio

SNR	Signal to Noise Ratio
SPS	Set Parameter Set
SVM	Support Vector Machines
TCP	Transmission Control Protocol
TP	True Positive
TTI	Transmission Time Interval
UE	User Equipment
UDP	User Datagram Protocol
UMTS	Universal Mobile Telecommunications System
VoD	Video on Demand
VoIP	Voice over IP
WiMax	Worldwide Interoperability for Microwave Access
WLANs	Wireless Local Area Networks
WWW	World Wide Web

Chapter 1

Introduction

1.1 Motivation

The emerging multimedia services become a main contributor in the ever increasing Internet Protocol (IP) traffic. In the last few years, we could witness the tremendous growth of multimedia services, specially online video streaming services, which have prevailed in the global Internet traffic with a larger distinct share. According to Cisco forecast report, the total global consumer of Internet video traffic will be 69% of all consumer Internet traffic in 2017, thus increasing by 57% percent in 2012. This 69% does not consider the video exchange through Peer-to-Peer (P2P) file sharing. However, if we add all forms of video (TV, Video on demand[VoD], Internet and P2P) the fraction will be 80% to 90% of global consumer traffic by 2017 [49]. Generally, network operators use different methods to improve the end-to-end Quality of Service (QoS), but these schemes are not enough to satisfy the end user. Therefore, service providers change their strategies from QoS-oriented towards the user-oriented, because a high user's satisfaction is a main objective in their business.

It is difficult for a network service provider to guarantee a high user satisfaction in various networks with different access technologies. Wireless communication systems use different access technologies ranging from different IEEE standards of Wireless Local Area Networks (WLANs) to broadband Fourth Generation (4G) mobile cellular networks. Cisco forecast report states that the global mobile data traffic will increase nearly by 11-fold in

2018 [50]. The multimedia traffic will be the main contributor over the wireless communication system. It is a big challenge for future Fifth Generation (5G) wireless networks, to provide these services in an efficient way in order to deal with the end users' quality expectations. To cater this problem, Cloud Computing is considered a fundamental part of the next-generation (i.e. 5G) cellular architecture that provides powerful computing platform to support ultra high-definition video services (e.g. Live IPTV, 2D/3D video, Video on Demand "VoD", Interactive gaming, etc.) to fulfil the demand of end users.

The cloud computing improves end users' experience by managing these services at remote data centers. Because of this trend, a large number of remote data centers have emerged, which is made possible by the availability of fast and reliable internet networks. In cloud computing, many applications and services are available to users remotely. As a consequence, users expect the best network QoS with a high quality standard [56].

The concept of Quality of Experience (QoE) has recently gained greater attention in both wired and wireless networks, especially in future networks (e.g. 5G). Its main objective is not only to consider and evaluate the network QoS, but also to better estimate the perceived quality of services by customers. In fact, the aim of network service providers is to provide a good user experience with the usage of minimum network resources. It is essential for network service providers to consider the impact of each network factors on user perception, because their businesses are highly dependent on users' satisfaction. According to Daniel R. Scoggin, *"The Only way to know how customers see your business is to look at it through their eyes"*.

There are some well-know quotes from the industry experts and other people, who high lighted the importance of customer's experience:

"The Customer's perception is your reality". Kate Zabriskie (Founder Business Training Works) .

"A satisfied customer is the best business strategy of all". Michael LeBoeuf (Businessman.

"The customer experience is the next competitive battleground." Jerry Gregoire (CIO,

Dell Computers.

"Your most unhappy customers are your greatest source of learning." Bill Gates (Businessman, Microsoft's Founder).

"Know what your customers want most and what your company does best. Focus on where those two meet." Kevin Stirtz (Book writer 'More Loyal Customer')

"The first step in exceeding your customer's expectation is to know those expectations." Roy H. Williams (Businessman).

In this context, it is necessary to understand the user/customer quality requirements, and hence this objective is defined via the term "QoE". Network service providers and researchers are making strong efforts to develop mechanisms that measure the user perceived quality while using the multimedia service (e.g. video streaming, etc.) [25]. QoE represents the real quality experience from the users' perceptive when they are watching the video streaming, or using any other multimedia service. QoE is defined as "the measure of overall acceptability of an application or service perceived subjectively by the end user" [85]. The European Network on Quality of Experience in Multimedia Systems and Services, (Qualinet) [87], also defines QoE in other perspectives, which are

"Quality of Experience (QoE) is the degree of delight or annoyance of the user of an application or service. It results from the fulfillment of his or her expectations with respect to the utility and / or enjoyment of the application or service in the light of the user's personality and current state."

QoE: "Degree of delight of the user of a service. In the context of communication services, it is influenced by content, network, device, application, user expectations and goals, and context of use."

The tremendous growth in consumer electronic devices with enhanced capabilities, along with the improved capacities of wireless networks have led to a vast growth in multimedia services. The new trends in the electronic market have developed a large variety of smart mobile devices (e.g. iPhone, iPad, Android, ...) which are powerful enough to support a wide range of multimedia applications. Meanwhile, there is an increasing demand for high-speed data services; 3rd Generation Partnership Project (3GPP) introduced the new radio access technology, LTE and LTE-Advanced (henceforth referred as LTE) which has the capability to provide larger bandwidth and low latencies on a wireless network in order to fulfill the demand of User Equipments (UEs) with acceptable Quality of Service (QoS). A large number of data applications are also developed for smart mobile devices, which motivates users to access the LTE network more frequently [26].

Voice over IP (VoIP) and Video streaming are key multimedia traffic services, that are widely used. VoIP is a popular low cost service for voice calls over IP networks. The success of VoIP is mainly influenced by user satisfaction, in the context of quality of calls as compared to conventional fixed telephone services. The main challenge for VoIP service is to provide the same QoS as a conventional telephone network, i.e. reliable and with a QoS guarantee. In conventional networks, the bearer quality is managed as a single quality plan, while in Next Generation Networks (NGNs), it is also necessary to manage end-users QoE. In a wireless system, the unpredictable air interface behaves differently for each UE. In these circumstances, it is necessary to monitor the QoE in the network on a call-by-call basis [86]. We consider the VoIP traffic in LTE scheduler to allocate the radio resource based on the user's QoE.

Video streaming is a main and growing contributor to Internet traffic. This growth comes with deep changes in the technologies that are employed for delivering video content to end-users over the Internet. To meet the high expectation of users, it is necessary to analyze video streaming services thoroughly in order to find out the degree of influence of (technical and non-technical) parameters on user satisfaction. Among these factors, one can find network parameters, which represent the QoS. Delay, jitter and packet loss are the main parameters of QoS, and they have a strong influence on user (dis)satisfaction. In addition to network parameters, some other external environmental factors have a great impact on user perceived quality, such as video parameters, terminal types, and psychological factors.

Generally, researchers use two methods to assess the quality of multimedia services: the subjective method and the objective method. The subjective method is proposed by the International Telecommunication Union-Telecommunication (ITU-T) [31], which is used to find out the users' perception of the quality of video streaming. The Mean Opinion Score (MOS) is an example of the subjective measurement method in which users rate the video quality by giving five different point scores from 5 to 1, where 5 is the best and 1 is the worst quality. However, the objective method uses different models of human expectations and tries to estimate the performance of a video service in an automated manner, without involving human. The subjective and objective methods, to evaluate the QoE, have their own importance, and they complement each other instead of replacing each other. It is very difficult to measure subjectively the MOS of in-service speech quality because MOS is a numerical average value of a large number of user's opinion. Therefore, many objective speech quality measurement methods are developed to make a good estimation of MOS. The E-model [77] and Perception Evaluation of Speech Quality (PESQ) [27] are objective methods for measuring the MOS scores. PESQ cannot be used to monitor the QoE for real-time calls, because it uses a reference signal and compares it to the real degraded signal to calculate the MOS score. Therefore, we have used the E-model computational method to calculate the MOS score of conversation quality by using the latency (delay), and packet loss rate with the help of the transmission rating factor (R-factor) [77].

1.2 Thesis Structure

The thesis is organized into six chapters. The brief description of chapter is presented as follows:

Chapter 2 - Literature Review and Related Work:

This chapter reviews the general literature and related works done in relation to this thesis. The chapter is divided in three sections that correspond to the contribution of each chapter. The analysis of QoE is not an easy task, because all the factors that directly or indirectly influence the user's perceived quality have to be considered. Researchers use distinct methods to correlate the network QoS parameters with user's QoE. Mostly, the developed methods are based on testbed experiments involving different equipments, methods, and tools. The datasets, collected at the end of a testbed experiment, are analyzed to observe the influence of different factors on user's QoE. The user's profile is also built-up based on of testbed experiments. Similarly, rate adaptive video streaming approaches are evaluated via the testbed experiment, where performance parameters of three important elements (client, server, and network) are considered to evaluate the proposed methods. Lastly, we focus on LTE-A networks, and discuss the various scheduling methods used to allocate radio resources to the UE based on different criteria by taking into account different parameters. The role of power saving method is also discussed within the context of different wireless systems, and we highlight its impact upon the performance on the system.

Chapter 3 - Methodologies for Subjective Video Streaming QoE Assessment:

In this chapter, we discuss two approaches to collect a subjectively dataset for assessing the user's QoE while using video services. These approaches take the form of a controlled, and an uncontrolled environmental framework. In the controlled environment, a laboratory testbed is implemented to collect the datasets and user's QoE in the perspective of different parameters (QoS parameters, video characteristic, device type, etc.). The data is stored in the form of a MOS value. The dataset is then used to analysis the correlation between QoS and QoE by using the six Machine Learning (ML) classifiers. The dataset also consists of user's profile that is built-up by collecting the information from users. The user's profile is used to investigate the impact of different parameters on user perception. In the uncontrolled environment, an application tool based on crowdsourcing is described, that can be used to investigate the users' QoE in a real environment. It subjectively collects user's opinion about video quality, and during the watching of the video, it stores the real-time network performance parameters in a local SQL database. Additionally, the tool measures and stores the real time performance characteristics of the end user device in terms of system memory, performance capacity, CPU usage and other parameters.

Chapter 4 - Regulating QoE for Adaptive Video Streaming:

This chapter describes the general video rate adaptive system, and highlights the key elements that play an important role to regulate video streaming service at the client side. The adaptive video streaming architecture is discussed, which mainly consists of three components; client, delivery network, and server. We propose a novel client-based rate adaptive video streaming algorithm that dynamically selects the suitable video segment based on dynamic network conditions, and client parameters. The proposed BBF method takes into account three important QoS parameters in order to regulate the user's QoE for video streaming service over HTTP, which are: Bandwidth, Buffer, and dropped Frame rate (BBF). The BBF is evaluated with different buffer lengths, and our results illustrate that a longer buffer length is less affected with dynamic bandwidth, but it does not efficiently utilize network resources. The BBF performance is compared with Adobe's OSMF streaming method, and results show that BBF method effectively manages the situation of sudden dropping in bandwidth, and dropped frame rate when the client system does not have enough resources to decode the frames. In case of lower buffer length, the BBF switches to the lower video quality in an aggressive way, and optimizes the user's QoE by avoiding the stalling, and pausing during video playback.

Chapter 5 - QoE Based Power Efficient LTE-A Downlink Scheduler:

This chapter presents the general overview of the LTE-A wireless network. We focus on the downlink scheduling method, because downlink is more important than uplink due to high-traffic flows. The QoE based LTE-A downlink scheduling algorithm is proposed for delay sensitive multimedia traffic (VoIP). The general architecture of a LTE-A scheduler is presented, and main elements that play an important role in the scheduling are presented along with three communication layers of LTE-A network. The performance of proposed downlink scheduler, i.e QoE Power Efficient Method (QEPEM), is evaluated along efficient power utilization of User Equipment (UE). The goal is to develop a downlink scheduling algorithm that allocates the radio resources to the UE by taking into account user's QoE along with the power saving method, i.e Discontinuous Reception (DRX). The performance of QEPEM is evaluated and compared with traditional scheduling methods, which are Proportional Fair (PF) and Best Channel Quality Indicator (BCQI). The QEPEM method endeavours to enhance the QoE and provide better QoS by decreasing the packet losses, improve fairness among the UE and considering the QoS requirement of multimedia service (e.g., delay). Simulation results show that the QEPEM performs in a superior way than traditional schedulers along with better user's experience, because it allocates resources efficiently among the UEs.

Chapter 6 - Conclusions and Future Work:

This chapter concludes the thesis work, and includes the future investigations. The chapter summarizes the results for distinct methods are used in order to investigate the concept of QoE for multimedia services through the analysis of technical and non-technical parameters. It also addresses the challenges to investigate user QoE for multimedia services, and high light the impacts of different parameters on user perception. Several future research directions and open issues can be derived from our work. We present several future directions to further explore the different factors on user's QoE.

1.3 Main Contributions

The main contributions of our work are summarized as follows:

1. We present two subjective methods, which are used to collect datasets for assessing QoE of video service, and analyses the impact of different parameters. In first method, we setup a testbed experiment in a controlled environment according to International Telecommunication Union-Telecommunication (ITU-T) [31]; however, in second method, we propose a crowdsourcing tool for assessing QoE in un-controlled environment. In controlled environment approach, we measure the influence of different parameters on the user perceived QoE, while watching the video service. The impact of different parameters (QoS parameters, video characteristic, device type, etc.) on user perception is recorded in the form of MOS value. The subjective collected dataset is used to investigate the correlation between QoS and video QoE. Six ML classifiers are used to classify the collected dataset. In case of mean absolute error rate, it is observed that Decision Tree (DT) has a good performance as compared to all other algorithms. An instance classification test is also performed to select the best model, and results clearly show that performance of RF, and DT are approximately at the same level. Finally, to evaluate the efficiency of DT and RF, a statistical analysis of classification is done, and results show that RF performs slightly better than DT [1].

2. The datasets is also used to investigate the impact of different QoS parameters on user's profile, and comprehensive study of users' profile gives useful information for network service providers to understand the behaviour and expectation of end users. The analysis shows that interesting videos' content has more tolerance than non-interesting videos' content. Similarly, the users for HD videos' content are more sensitive in the delay and packet loss, while for Non-HD videos' content, the users have more tolerance levels. Based on users' profile analysis, the network service

[1]**M.Sajid Mushtaq**, Brice Augustin, and Abdelhamid Mellouk, *Empirical study based on Machine Learning Approach to Assess the QoS/QoE correlation*. In 17th European Conference on Network and Optical Communications (NOC 2012), Barcelona, Spain, June 20-22, 2012.

provider can efficiently utilize their resources to improve user satisfaction [2].

3. In un-controlled environment, a crowdsourcing application tool is developed that can be used to investigate the users' QoE in real-time environment. The application tool uses the feedback form to subjectively record the user's perception. It can monitor and store the real time performance parameters of QoS (packet loss, delay, jitter and throughput). Instead of QoS networks, the tool also measures the real time performance characteristics of the end user device in terms of system memory, performance capacity, CPU usage and other parameters [3].

4. The client-side HTTP rate adaptive BBF method is proposed that adapts the video quality based on three main QoS parameters, such as dynamic network bandwidth, user's buffer status, and dropped frame rate. The BBF is evaluated with different buffer length, and it is observed that a longer buffer length is less affected with dynamic bandwidth, but it is also not efficiently utilized the network resources. The BBF is evaluated and compared with Adobe's OSMF streaming method. It is observed that BBF successfully manages situation as compared to OSMF, in terms of sudden drop of bandwidth, and dropped frame rate when the client system does not have enough resources to decode the frames. Additionally, BBF method optimizes the user's QoE by avoiding the stalling, and pausing during video playback [4] [5].

5. The downlink scheduling algorithm $QEPEM$ is proposed for delay sensitive traffic (VoIP). The QEPEM method endeavours to enhance the QoE and provide better QoS by decreasing packet losses, improve fairness among the UE and considering the

[2] **M.Sajid Mushtaq**, Brice Augustin, and Abdelhamid Mellouk. *QoE: User Profile Analysis for Multimedia Services*. In Proc. of IEEE International Conference on Communications (ICC), Sydney, Australia, June 10-14, 2014.

[3] **M.Sajid Mushtaq**, Brice Augustin, and Abdelhamid Mellouk. *Crowd-sourcing Framework to Assess QoE*. In Proc. of IEEE International Conference on Communications (ICC), Sydney, Australia, June 10-14, 2014.

[4] **M.Sajid Mushtaq**, Brice Augustin, and Abdelhamid Mellouk. *Regulating QoE for Adaptive Video Streaming using BBF Method*. In Proc. of IEEE International Conference on Communications (ICC), London, UK, June 10-14, 2015.

[5] **M.Sajid Mushtaq**, Brice Augustin, and Abdelhamid Mellouk. *HTTP Rata Adaptive Algorithm with High Bandwidth Utilization*. In Proc. of IFIP/IEEE International Conference on Network and Service Management (CNSM), Rio, Brazil, November 17-21, 2014.

QoS requirement of multimedia service. It can assure QoS in the power saving environment with high users' satisfaction [6]. The QEPEM method maximizes the user's QoE by using the user perception in its scheduling decision, and its performance is compared with the traditional schemes according to different QoS attributes through simulations. It is observed that packet loss rate has more influence on QoE as compared to delay. The QEPEM method is evaluated in the power saving mode and the impact of the power saving on QoS and QoE is also examined. In the power saving environment, the QEPEM method performance is remarkably better than the traditional schedulers with better user's experience because it allocates resources efficiently and fairly among the UEs [7].

[6]**M.Sajid Mushtaq**, Scott Fowler, Abdelhamid Mellouk, and Brice Augustin. *QoE/QoS-aware LTE downlink scheduler for VoIP with power saving*. In Elsevier International Journal of Networks and Computer Applications (JNCA); DOI: 10.1016/j.jnca.2014.02.01.

[7]**M.Sajid Mushtaq**, Abdelhamid Mellouk, Brice Augustin, and Scott Fowler. QoE Power-Efficient Multimedia Delivery Method for LTE-A, IEEE System Journal, to appear, 2015.

List of Publications

Journals (Rate A)

All journals are indexed in Journal Citation Reports (JCR), WebOfSciences.

Submitted:

1. **M.Sajid Mushtaq**, Brice Augustin, and Abdelhamid Mellouk. *Methodologies to Assess QoE for Multimedia Traffic*, submitted in ACM Transaction on Multimedia Computing, Communications, and Applications, in March 2015.

Accepted:

1. **M.Sajid Mushtaq**, Abdelhamid Mellouk, Brice Augustin, and Scott Fowler. *QoE Power-Efficient Multimedia Delivery Method for LTE-A*, IEEE System Journal, to appear, 2015.

2. **M.Sajid Mushtaq**, Scott Fowler, Abdelhamid Mellouk, and Brice Augustin. *QoE/QoS-aware LTE downlink scheduler for VoIP with power saving*. In Elsevier International Journal of Networks and Computer Applications (JNCA); DOI: 10.1016/j.jnca.2014.02.01.

Conferences with Proceedings (Rate B)

1. **M.Sajid Mushtaq**, Scott Fowler, Brice Augustin, and Abdelhamid Mellouk. *QoE in 5G Wireless Cellular Network based on Mobile Cloud Network*. In IEEE International Workshop on Multimedia Cloud Communication, along with 34th IEEE International Conference on Computer Communications (INFOCOM), Hong Kong, China, April 26 - May 1, 2015.

2. **M.Sajid Mushtaq**, Brice Augustin, and Abdelhamid Mellouk. *Regulating QoE for Adaptive Video Streaming using BBF Method*. In Proc. of IEEE International Conference on Communications (ICC), London, UK, June 10-14, 2015.

3. **M.Sajid Mushtaq**, Brice Augustin, and Abdelhamid Mellouk. *HTTP Rata Adaptive Algorithm with High Bandwidth Utilization*. In Proc. of IFIP/IEEE International Conference on Network and Service Management (CNSM), Rio, Brazil, November 17-21, 2014.

4. **M.Sajid Mushtaq**, Brice Augustin, and Abdelhamid Mellouk. *QoE: User Profile Analysis for Multimedia Services*. In Proc. of IEEE International Conference on Communications (ICC), Sydney, Australia, June 10-14, 2014.

5. **M.Sajid Mushtaq**, Brice Augustin, and Abdelhamid Mellouk. *Crowd-sourcing Framework to Assess QoE*. In Proc. of IEEE International Conference on Communications (ICC), Sydney, Australia, June 10-14, 2014.

6. **M.Sajid Mushtaq**, Brice Augustin, and Abdelhamid Mellouk, *QoE-Based LTE Downlink Scheduler for VoIP*. In Proc. of IEEE Wireless Communication and Networking Conference (WCNC), Istanbul, Turkey, April 6-9, 2014.

7. **M.Sajid Mushtaq**, Brice Augustin, and Abdelhamid Mellouk, *Empirical study based on Machine Learning Approach to Assess the QoS/QoE correlation*. In 17th European Conference on Network and Optical Communications (NOC 2012), Barcelona, Spain, June 20-22, 2012.

8. **M.Sajid Mushtaq** , Abdussalam Shahid and Scott Fowler, *QoS-Aware LTE Downlink Scheduler for VoIP with Power Saving*. In 15th IEEE International Conference on Computational Science and Engineering (CSE), Paphos, Cyprus, December 5-7, 2012.

Book Chapter:

1. **M.Sajid Mushtaq**, Brice Augustin, and Abdelhamid Mellouk, *"QoE Approaches for Adaptive Transport of Video Streaming Media"*, Wiley Ed/ISTE Book "Quality

of Experience Engineering for Customer Added Value Services: From Evaluation to Monitoring", (Abdelhamid Mellouk, Antonio Cuadra-Sanched, Ed.), ISBN:978-1-84821-672-3, Chapter 8, pp 151-170, 2014.

Chapter 2

Literature Review & Related Work

In this chapter, we review the some literature in conjunction with related work. We divide the related work in three main sections that represent the contribution of each work presented in the succeeding chapters. First, we present different methods that are generally used to collect QoE dataset. The dataset is used to investigate the impact of different parameters on the user perceived QoE. The dataset also contains user's profile which consists of user personal detail, and other key information related to service under testing. Second, we review the different standards, and proposed video rate adaptive methods in the literature. Third, we discuss the various scheduling methods that allocate resources to UE by considering the different QoS parameters, and others elements including power status.

2.1 Introduction

In the middle of the last century, the multimedia video service started and it spread out rapidly with the introduction of television. In the late 90's, Internet service enabled the viewing of online recorded videos. Later, with the continuous innovation in Internet broadband service, the network service provider offered more capacity and high-speed download link to the end user, that boomed the video streaming service over the IP network. Cisco predicts that the total global consumer of Internet video traffic will represent 69% of all consumer Internet traffic in 2017 [49]. Nowadays, the watching of online video contents is easily possible thanks to the availability of a large variety of consumer electronics devices.

The remarkable growth in video-enabled electronics devices, comprising Personal Computers (PCs), Smartphones, Tablets, Internet-enabled Television, and accessibility of high speed Internet (WiFi/3G/4G) are key factors for the growing popularity of online video content. The earlier trends of TV media change quickly, and reached a point where a large number of consumers expect the availability of video services on any device over any network connection, but delivered at the same high quality as they expect from a conventional TV service.

The explosive advancement in the core and radio link capacity, the future 5th Generation (5G) networks is expected to provide high-speed links to each user (upto 10 Gbps) [105]. The enhancement of wireless communication system opens a new door of opportunity for providing a High Definition (HD) video streaming to users, at all time. The world trend is moving towards "Everything over IP", and the significant benefit of future 5G is to provide different types of services e.g. Voice, Text, and high quality Video by using the Internet Protocol (IP) network. The IP infrastructure is quickly replacing the traditional system in order to offer more services to users at low cost. IP networks offer best-effort services, therefore Quality of Service (QoS) of video streaming can be degraded by packet loss, delay, jitter, and throughput, which also degrades the Quality of Experience (QoE). The Internet is an unmanaged network, and transmission of video streaming requires new mechanisms in order to provide the highest quality video streaming to the users, as they are expected from the managed TV delivery networks.

2.2 Subjective Test

Internet is a collection of diverse network, where video delivery from source to destination is carried out through distinct unique elements, which have complex interactions. The video service is more susceptible of impairments and problems as compared to data and voice services. Unlike a data service, the video service generally has no second chance for retransmission of lost data, because user can visibly observe the impact of lost video packet, while in case of a data service, the user is unaware about retransmission of lost data. The network QoS is a key factor that influence the user perceived QoE. A large number of research works have been achieved to correlate QoS with QoE in search of capturing the degree of user entertainment. Some other techniques are also developed to evaluate and predict the users' QoE, in order to deliver a better quality of service to end-users. In the controlled environment, many testbed studies have been undertaken, involving different tools, equipments and methods.

2.2.1 Controlled Environment Approach

The controlled environment approach refers to laboratory test experiment, where all environmental factors are fixed that can influence the user perceived experience. International Telecommunication Union-Telecommunication (ITU-T) has defined the recommendation to setup and carry out the laboratory testbed experiment [51]. In [79], a testbed experiment is proposed, to explore how network QoS affects the QoE of HTTP video streaming. In [36], a testbed is implemented to collect data with the help of ten participants, correlating stream state data with video quality ratings. These datasets were used to develop self-healing networks, i.e., having the ability to detect the degradation of video streaming QoE, react and troubleshoot network issues. The correlation of QoE-QoS is studied in [102] by controlling QoS parameters (packet loss, jitter, delay) of networks. Because subjective campaigns are, by nature, quite limited in size and number of participants, it is impossible to cover all possible configurations and parameter values. However, a QoE prediction model is proposed in [2], for the unseen cases based on primarily limited subjective tests. This model reduces the need of cumbersome subjective tests, to the price of a reduced accuracy. To overcome the weakness of [2], a Learning-based prediction model is proposed in [75]. In [74], a machine learning technique is proposed using a subjective quality feedback. This technique is used to model dependencies of different QoS parameters related to network and application layer to the QoE of the network services and summarized as an accurate QoE prediction model.

Large research works have carried out in order to provide the application services with acceptable quality. The researchers study the different techniques to correlate the network's QoS with end user perceived QoE. Some other methods are also developed to provide the better QoS for evaluating and predicting the user's QoE. Generally, the developed methods are studied and examined in the form of experiments by setting up the testbed, which consists of different equipments, methods and tools. The datasets, collect in the end of testbed's experiment, are analyzed by observing the impact of different factors subjectively perceived by end users. The user's profile is also built-up as an outcome of this testbed.

In [62], a testbed experiment is implemented to assess the QoE model for video streaming service using the QoS parameters in the wired-wireless network. In this paper, the authors just consider the QoS parameter to estimate the perceived QoE of end-users and do not consider the important information related to users' profile. Similarly, a testbed experiment is done in [79] which also simply consider the QoS parameter and investigate to show that how network QoS affects the QoE of HTTP video streaming. In [61], the authors propose the objective method for measuring the QoE by using the QoS parameters. In this paper, the QoS and QoE correlation model is proposed and the QoE evaluation method using the QoS parameter in the converged network environment. A lot of research works are done to predict the QoE based on the QoS parameters. The correlation between QoE-QoS is studied in [102], where authors investigate how the controlled QoS parameters (packet loss, jitter, delay) of networks influence the QoE. In [41], authors highlight the problem with existing QoE model, which do not take into account the historical experience of user satisfaction while using the certain service. This important psychological influence factor is called memory effect, which plays a vital role to meet the expectation of end-users for better QoE.

A lot of studies are done on user's profile, but mostly investigations are relating with World Wide Web (WWW). In that circumstance, it is very important for the service provider to find out the pattern that clearly pointed out the utilization of information at the end system. In [12], authors use the fuzzy clustering algorithm to analysis the e-learning behaviour of the user. The analysis of cluster helps the teacher to understand students in a better way by considering their interest, personality and other informations. In [98], authors describe a method which presents the information to the end user by considering user's profile. The user's profile is a key factor which can be very helpful for the network service providers to offer the service that is acceptable for end users. In our work, we intend to investigate the statistical analysis of QoS parameters and their impact on end users. It helps the network service provider to utilize its resources efficiently and get high user satisfaction by maintaining the certain threshold of QoS parameters.

2.2.2 Uncontrolled Environment Approach

The investigation of QoE is not a simple task, because all the variables that directly or indirectly influence the user's perceived quality should be considered. Researchers study the different techniques to correlate the network QoS with end user's QoE. Some other methods are also developed to provide the better QoS in order to evaluate and predict the end user's QoE. Generally, it is considered that by providing the better network QoS will result the good QoE, and it is true to some extent. However, always providing the good parameters of network QoS will not guarantee to satisfy the end user, and it occurs due to some uncontrollable or external environment factors, such as video parameters, terminal characteristics, and psychological factors.

In uncontrolled environment, the crowdsourcing method is an alternative of laboratory testing approach for assessing the QoE of video service. In crowdsourcing environment, a testing task (e.g. video) is allocated to a large group of anonymous users, who can participate in the testing task from different parts of the world via Internet using their own devices. In [62], a testbed experiment is implemented to assess the QoE model for video streaming service using the QoS parameters for the wired/wireless network. In this paper, authors just consider the QoS parameter to estimate the perceived QoE of end-users and do not consider the important information relating to users' profile and terminal properties. In [79], QoE is evaluated for HTTP video streaming. In this paper, different network QoS parameters (packet loss, delay and throughput) are used, and observed the impact of QoS parameters in the form of stalling event. The testbed is implement in a controlled environment (laboratory), and each test condition used only one video streaming clip with 10 users. In this study, authors do not consider the property of terminal and the few numbers of participants providing their quality experience based on one video, do not reflect the reliability of QoE. In [42], a crowdsourcing approach is presented to assess the QoE for TCP based online video streaming service, YouTube. In this paper, authors only consider the influence of stalling event (as a key factor) on user's perceived quality. The authors do not take into account the QoS parameters and characteristics of terminal, which have the greater impact on QoE.

A web-based crowdsourcing platform to assess the QoE is presented in [13]. This platform is designed in such a way that researchers have administrative control, which defines the type of multimedia test, register or update experiment profiles, setting or description of crowdsourcing test and finally after the test they download the results logs files. The test's participant also gets a reward as a payment. The reliability of the end results cannot be proved due to the following reasons; remote and unknown participant, some participants may submit the incorrect results in order to earn more money by completing the more test; some participant can not understand the test description correctly and complete the task incorrectly. In [42] and [37], authors also use the paid crowdsourcing platform which is called mircroworkers. The microworkers has a large number of registered workers who participate in the crowdsourcing experiments. This is also a paid platform that can face the same problems as we have discussed earlier. In this work, we present our developed crowdsourcing framework to assess the QoE of online video streaming. It is a user-friendly framework, which is very easy to install and use without complexity. The proposed framework has the capability to capture and store the important informations that help in analysis and evaluating the QoE.

2.3 Adaptive Video Streaming Methods

Video streaming over the Hypertext Transfer Protocol (HTTP) is highly dominant due to the availability of Internet support on many devices, and it easily traverses NATs and firewalls, unlike other media transport protocols such as RTP/RTSP. The adaptive video streaming over HTTP becomes attractive for service providers, as it not only uses the existing infrastructure of Web downloading (thus saving an extra cost), but it also gives the ability to change the quality of video (bitrate) according to available bandwidth for increasing user's perceived quality. Video streaming over HTTP is an easy and cheap way to move data closer to network users, and the video file is just like a normal Web object.

Initially, it was considered that the Transmission Control Protocol (TCP) is not suitable for video streaming, because of its properties of reliability and congestion control. Indeed, a reliable data transmission can cause a large retransmission delay, and congestion control causes a throughput variation. Consequently, earlier researchers considered the

User Datagram Protocol (UDP) as the underlying transport protocol, as it is an unreliable connectionless protocol that simplifies data transmission. Later on, it was proved that TCP mechanisms for reliable data transmission and congestion control do not effectively degrade video quality, especially if the client player has the ability to adapt to the the large throughput variation. Additionally, the use of TCP over HTTP does not face any problem of data filtering (through firewalls and NATs), because they allow to pass the HTTP file through port 80, like regular Web objects.

Earlier, HTTP-based video streaming application used the progressive download method (HTTP over TCP) and thanks to its simplicity this method became very popular for viewing online video contents. This method has some limitations that degrades the QoE, because it lacks the rich features of video streaming, e.g. trick modes such as fast-forward seek/play, rewind, and often freezing or rebuffering due to the shortage of bandwidth. The new emerging approach for adaptive streaming not only replaces the progressive download but it also covers the shortcoming features. The adaptive streaming is a pull-based media streaming approach that consists in progressive download and a streaming method [8].

The evolution of the adaptive video streaming leads to a new set of standards from well-known organizations, i.e., Adobe, Microsoft, Apple, and 3GPP/MPEG. These standards are widely adopted as they increase user's QoE by providing video service over HTTP, but in an adaptive manner, according to network conditions and device characteristics. The HTTP adaptive streaming technologies provided by these organizations are Adobe HTTP Dynamic Streaming (HDS), Microsoft Silverlight Smooth Streaming (MSS), Apple HTTP Live Streaming (HLS), and MPEG Dynamic Adaptive Streaming over HTTP (DASH).

2.3.1 Traditional Streaming vs Adaptive Streaming

In the traditional IP streaming, the video is delivered to users through a number of proprietary 'stateful' protocols such as RTSP (Real Time Streaming Protocol), Adobe's RTMP (Real Time Messaging Protocol), and Microsoft's MMS (Microsoft Media Server). These protocols make a dynamic point-to-point link between user devices and the streaming server in order to handle the state of the video. The user and server must have synchronized video's states, e.g., playing, pause, stop, etc. Generally, traditional video streaming is delivered over UDP, an unreliable connectionless protocol that degrades the user's QoE because of packet losses. The complex synchronization between client and server allow the traditional video streaming to adapt the variation in network bandwidth, but as an outcome, those adaptive protocols were not widely adopted due to their complexity. RTSP is a good example of a traditional video streaming protocol as shown in Figure 2.1, where the client connects to the video streaming server until it sends a disconnection request to the server, and the server keeps monitoring the state of the client. The default RTSP packet size is 1452 bytes. When a video is encoded at the rate of 1 Mbps, each packet will carry almost 11 milliseconds of video.

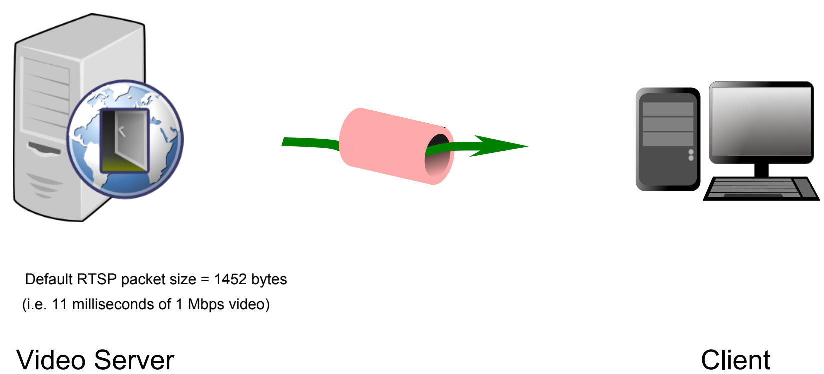


Figure 2.1 – RTSP Traditional Video Streaming

In equivalence, the success of HTTP technologies provides the opportunity to develop Content Delivery Networks (CDNs) and network operators effectively manage the 'stateless' HTTP protocol networks. The innovation in the HTTP video streaming was started by Move Networks, it is called Adaptive Streaming. This adaptive streaming increases the quality and resolution of video content according to the handling capability of the user device, throughout the data network. The adaptive streaming server maintains different copies of the same video content that vary in bit-rate, and client can switch to high quality content according to available bandwidth.

In HTTP adaptive streaming, the source video content (either a file or live stream) is broken into file segments, called fragments, chunks or segments, using the desired format, which contain video codec, audio codec, encryption protocol, etc. Generally, the segment length is between 2-10 seconds of the stream. The segment file consists either in a multiplexing container that mixes the data from different tracks (video, audio, subtitles, etc.) or it can be a single track. The stream is divided into chunks at boundaries of video Group of Picture (GOP), identified by an IDR frame. The IDR is such a frame that can be decoded independently, without looking for other frames, and each chunk does not depend on previous and successive chunks. The file segments are hosted on a regular HTTP server. The general HTTP adaptive streaming is shown in Figure 2.2.

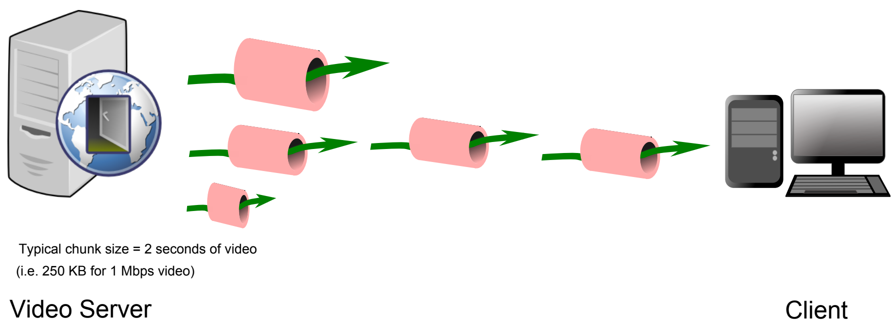


Figure 2.2 – Adaptive Video Streaming

Generally, video adaptive methods are divided into three main categories: 1) Transcoding-based, 2) Scalable encoding-based, and 3) Multiple bitrate switching.

1. Transcoding-based: It adapts the video content that corresponds to a specific bitrate during on-the-fly transcoding of the raw data [89]. This technique is good, because it can limit the frame rate, compression, and video resolution. However, it requires more processing power, and has a poor scalability, because transcoding is done separately for each client, as a result it is difficult to implement in CDNs.

2. Scalable Encoding-based: It is an important adaptation method that used scalable codec like H264/MPEG-4 SVC [63], [65]. Without recode the raw video data; the both spatial and temporal scalability is successfully achieved to adapt the video resolution and frame rate. This method has the advantage over transcoding-based technique, because it reduces the processing load by encoding the raw video date one time, and used the scalability features of the encoder to adapt on the fly. However, this approach has limitations, e.g. it cannot deploy in CDNs, as a special server is

required for adaptation logic, while content cannot be cached in standard proxies. Additionally, the video adaptation decision depends on used codec, that restricts the video content provider to use the limited codecs. [19].

3. Multiple Bitrate or Stream-switching: The leading streaming systems have been adopted this streaming method, e.g. Adobe HTTP Dynamic Streaming (HDS) [39], Microsoft Smooth Streaming (MSS) [80], Apple HTTP Adaptive Live Streaming (HLS) [40], Netflix for its popular video on demand service [83], Move Networks for live service of several TV networks [84]. MPEG introduces the Dynamic Adaptive Streaming over HTTP (DASH) method to promote the standardization and compatibility of stream switching systems [96]. It is standardized by ISO to transport the adaptive streaming over HTTP using the existing infrastructure [1]. The video raw content is encoded into different bitrates that results many versions of single video, and streaming method selects the suitable video bitrate version according to user's available bandwidth. This method has the advantage to reduce processing load, because one-time video encoding is required, and later no more processing is needed to adapt the video as per variable bandwidth. It also does not depend on employed codec, and encoder can work efficiently for each video quality level or version. The main disadvantage is more storage space required, and adaptation process only selects the available discrete video quality version.

It is a challenging task for researchers to efficiently transport the video streaming in a rate-adaptive manner over the TCP in conjunction with HTTP, particularly for delivering the High Definition (HD) video to the end users in order to achieve best QoE. Researchers propose different rate-adaptive methods by considering the dynamic behaviour of network conditions for achieving the specific goals in the perspective of distinct metrics.

Earlier, the sender-driven based rate adaptation is considered as a main method, where the sender/server estimated the client side parameters, and adapted the video streaming according to network situation. In [66], an adaptive method proposed that estimate the buffer occupancy of client at the server side, and adapted the video quality in order to maintain the client's buffer level above certain threshold value.

Recently, the rate adaptive approaches have been deviated from sender-driven based towards receiver-driven, where a client decides to adopt the video streaming quality by monitoring its parameters, and network conditions. In [71], authors proposed a receiver-driven rate adaptation algorithm for video streaming over the HTTP. The proposed method was evaluated by using the NS-2 simulator with the exponential and constant bit-rate background traffic. The method estimated the network bandwidth by using smoothed HTTP throughput that measured based on the Segment Fetch Time (SFT). The results clearly show that the proposed algorithm does not select the appropriate video quality, because it shows the fluctuation in the selection of proper video quality. In [5] authors high lighted the behavior of different adaptive players for HTTP video streaming in order to check their stability in different scenarios. In [58], authors observed the HTTP based adaptive streaming method in terms of fairness, efficient, and stability.

A receiver-driven rate adaptation algorithm proposed in [76], where proposed algorithm estimated the network bandwidth, and based on the client buffer length it chose an appropriate video quality. The authors evaluated the algorithm in different bandwidth scenarios, and it tried to keep the target buffer interval between 20 to 50 seconds. It is noticed that larger buffer length minimized the number of video quality shifts, because it was less affected with instantaneous variation in network conditions, and also it did not consider the impact of frame drops rate.

QoE-aware algorithm based on Dynamic adaptive streaming over HTTP (DASH) is discussed in [78] for video streaming. The main idea in video delivery was to optimize the user's perceived quality experience. Authors showed that frequent change of video rate significantly degrade the user' QoE, and it proposed to change the step by step video rate based on available bandwidth.

A rate adaptive algorithm based on bandwidth estimation for HTTP video streaming system is proposed in [109]. The authors proposed the new method for bandwidth estimation, and based on past transmission history, the algorithm predicted the amount of data that client could download during a certain interval in the future. The authors evaluated the proposed algorithm in terms of stalling frequency with Constant Bitrate (CBR), and did not consider the impact of sudden drop of bandwidth, and dropped video frame metrics.

2.4 Scheduling and Power Saving Methods

Many factors directly or indirectly influence the performance of wireless networks and UEs. Amongst these performance metrics, scheduling scheme has gained greater importance to efficiently allocate the radio resources amongst the UEs. The emerging and fastest growing multimedia services such as Skype, GTalk and interactive video gaming have created new challenges for wireless communication technologies, especially in terms of resource allocation and power optimization of User Equipments (UEs) as they both have high impact on system performance and user's satisfaction. The efficient resources and power optimization are very important in the next generation communication systems (e.g. 5G), because new multimedia services are more resources and power hungry. Having more traffic flow in the downlink as compared to uplink, the resource allocation schemes in the downlink are more important than uplink.

2.4.1 Scheduling Methods

Scheduling is a process of allocating the physical radio resources among the users, as to fulfil the QoS requirements of multimedia services. The aim of a scheduling scheme is to maximize the overall system throughput while keeping fairness, delay and packet loss rate within QoS requirements to satisfy end-users QoE.

Generally, users are classified on their traffic characteristics, such as real time and non-real time traffic. For real time traffic (e.g. video, VoIP and gaming), scheduling must guarantee that QoS requirements are satisfied. The packet loss rate and delay play a vital role in user experience. Packets in real time traffic must arrive to the user within a certain delay threshold, otherwise the packet is considered as lost or discarded. The scheduling decisions can be made on the basis of the following parameters; MOS, QoS parameters, traffic type, Channel Quality Indicator (CQI), resource allocation history, buffer status both at the eNodeB and UE.

The Best Channel Quality Indicator (BCQI) scheme assigns radio resources only to those UEs, which have reported the best channel conditions in the uplink through the CQI feedbacks to the corresponding eNodeB. In the meantime, those UEs that suffer from bad channel conditions will never get radio resources [92]. As a result of the BCQI scheme, the overall system throughput increases, but some UEs never get the resources, especially the ones that are far away from eNodeB, because of bad channel conditions. Thus, the BCQI scheme performs well in terms of throughput but poorly in terms of fairness among the UEs.

In order to overcome the fairness problem of BCQI, the Round Robin (RR) scheme was developed. It distributes radio resources equally among the UEs to gain high fairness. As a result, the overall system throughput is degraded because it does not consider the channel conditions of the UEs. To handle the constraints of high throughput and fairness, the Proportional Fair (PF) scheme was developed. PF uses an approach based on the trade-off between maximum achievable average throughputs and fairness.

A Channel-Adapted and Buffer-Aware Packet Scheduling scheme for the LTE communication system is proposed in [70]. This scheme makes scheduling decisions on QoS for Real Time (RT) services, which are based on three elements: CQI and UE buffer status feedback on the uplink, and it treats real-time and non-real-time UEs traffic separately. However, this scheduling scheme does not consider the packet delay factor which can increase the packet loss rate and degrade user satisfaction.

A two-layer scheduling scheme is discussed in [9], which maintains the fairness of radio resources and high throughput. The packet delay and Guaranteed Bit Rate (GBR) are vital parameters of an LTE system, which influence the QoS and determine the overall user QoE for the current service. However, this proposed scheme does not consider these important parameters. In [20], an admission control and resource allocation packet scheduling scheme is presented. It combines the time-domain scheduling and frequency-domain scheduling which maximizes the throughput while making sure that the user's delay never crosses the threshold value, and a user gets at least a minimum throughput to fulfil the QoS requirements. The QoS requirements are fulfilled by assigning more resources to those users which have critical delay and throughput (i.e. larger delay or minimum throughput). This proposed algorithm fulfils the QoS requirements of real-time and non-real-time traffic by considering the throughput and delay of each user, but it does not consider the channel conditions when assigning the resources to users.

A cross-layer resource allocation scheme for Inter-cell interference coordination (ICIC) was proposed in [72] for LTE networks. The potential of game theory is used to solve an optimization problem, so that the total numbers of RBs in different cells are treated adequately, and similarly the convergence of the algorithm is guaranteed. This proposed method is evaluated with two scheduling methods, which are PF and Modified Largest Weighted Delay First(M-LWDF) with fixed power allocation, and only the system throughput is considered as a performance metric. The Cumulative Distribution Function (CDF) of the normalized user throughput is used to compare the fairness of the proposed cross-layer scheme with MAX C/I, RR, and PF. The proposed method does not take into account the packet delay, GBR and other QoS parameters of the LTE networks which influence the QoE of the end-user. In [95], the congestion exposure mechanism is used to feedback the real-time objective QoE information in the network, as perceived by end-users. The authors, proposed a new queue management technique based on QoE metrics. Our proposed method is also using the real-time feedback of UEs to make the scheduling decision.

2.4.2 DRX Power Saving Method

The increasing demand of high speed data service, and dramatic expansion of network infrastructure, trigger an enormous increase of energy consumption in wireless network. Today, the optimal energy consumption has become a major challenge, and to overcome this challenge the different methods are proposed for efficient use of power energy of the different elements in wireless network infrastructure.

The DRX power saving method is used in different wireless communication systems with the main purpose to prolong the battery life through monitoring the UE activities. It is based on simple procedure, when there is not any transmitted data then it saves the power by switch-off the UE wireless transceiver. During the sleep state of the UE, the DRX method considerably increases the packet delay.

The DRX mechanism of UMTS is investigated in [107] with the help of an analytical model, where only DRX functionality consists of two parameters; Inactivity Time and the DRX cycle, between the NodeB and UE for saving the power of the UE. The effects of DRX cycles are observed by considering the timers, queue length and packet waiting times. In [112], the authors present an analytical model, which prove the LTE DRX mechanism has the ability to save more power than UMTS [90] DRX method.

The power saving methods for two different WiMAX standards, IEEE 802.16e and IEEE 802.16m are discussed in [14]. In this paper survey, the authors highlight the important issues related to power saving mechanism in WiMAX networks and address the several problems to improve its efficiency.

The influence of Transmission Time Interval (TTI) sizes, including the effects of LTE DRX Light and Deep Sleep mode on power consumption are evaluated in [34] for Voice and Web traffic. This study work does not consider the impact of these parameters on QoS. In [10] the DRX-aware scheduling is proposed which includes the DRX status as a scheduling decision parameter to reduce packet delay caused by the DRX sleep duration. The scheduling priority is directly proportional to delay of a head of line packet delay in relation to the remaining active time before a UE enters into sleep mode. In [28] semi-persistent scheduling scheme for VoIP is developed using the DRX. First it organizes the UEs into the scheduling candidate set (SCS) based on the UE buffer information at the eNodeB, the DRX status and the persistent resource allocation pattern. It calculates the priority metric for the UEs in SCS by favoring the UEs who require retransmissions then the UEs whose packet delay of unsent packet in the eNodeB buffer is close to delay threshold. Both schemes presented in [10] and [28] use DRX mechanism to optimize power usage and offer solutions to the problems caused by the sleep interval of increased packet delay and packet loss. However, both schedulers do not consider GBR requirement of UEs.

In [3], the performance of DRX mechanism is evaluated in terms of DRX cycle lengths and related timer values, by observing their effect on VoIP traffic service over the High Speed Downlink Packet Access (HSDPA) network. However, the battery life of UE might a key limiting factor in providing satisfactory user experience. The results showed that longer DRX cycle saves more UE power but at the same time VoIP capacity over HSDPA can be compromised in the case when there are not suitable selection of DRX parameters are applied.

In [111], the authors present the semi-Markov chain model to analysis the impact of DRX mechanism in LTE network with Machine Type Communication (MTC) traffic, while in [59], the authors proposed the method for modelling the DRX mechanism in LTE wireless networks with the help of Poisson traffic. In the same way, in [35], the analytical model is used to study the influence of fixed and adjustable DRX cycle mechanism in LTE network, using the bursty packet data traffic with the help of semi Markov process. However, these proposed methods [111], [59], and [35], do not consider the QoS features such as fair resources allocation, packet loss rate and throughput, which are badly effected with the DRX mechanism in LTE networks.

The impact of LTE DRX Light Sleep mechanism on QoS is examined in [81], using the VoIP traffic model. However, the performance is evaluated only with the LTE DRX Light Sleep Cycle, and Deep Sleep Cycle was not considered. In [57], the DRX optimization is performed for the mobile internet application by considering the DRX inactivity timer and the DRX cycle length with two users. This method is evaluated with only two users, and it also does not take into account the impact on other QoS parameters like fairness, throughput, packet loss rate, and GBR requirement for RT traffic.

Chapter 3

Methodologies for Subjective Video Streaming QoE Assessment

In the previous chapter, we review the general literature and related works done in relation to this thesis. The last chapter is divided into three sections that correspond to the three main contributions. This chapter presents the first contribution referred to subjective methods for evaluating the user's QoE using video streaming. In this chapter, we describe two significant subjective methods, i.e. controlled environment and uncontrolled environment methods, that used to collect QoE datasets in the form of a Mean Opinion Score (MOS). Later, the dataset is then used to analysis the correlation between QoS and QoE.

3.1 Introduction

It is a challenging task for service providers to assess the perceived Quality of Experience (QoE) for multimedia services. Generally, user's QoE for video service is measured in a totally controlled environment (e.g. experimental testbed), because it provides the freedom to easily measure the impact of controlled network parameters. However, in a real time uncontrolled environment, it is hard to assess the Quality of Service (QoS) perceived by the end-user, due to the time-varying characteristics of network parameters. In an uncontrolled environment, crowdsourcing is a technique used to measure the user's QoE on the client side.

This chapter presents the methodologies to asses the QoE for video services. It is essential to investigate how different factors contribute the QoE, in the context of video streaming delivery over dynamic networks. Important parameters which influence the QoE are: network parameters, characteristics of videos, terminal characteristics and users' profiles. The two important subjective methods are described that used to collect QoE datasets in the form of a Mean Opinion Score (MOS).

In a controlled environment, the subjective laboratory experiments are conducted in order to collect QoE datasets in the form of MOS scores. The impacts of different factors are evaluated using video services, and users perceived quality opinions are stored in the datasets. The collected datasets are used to analyse the correlation between QoS and QoE for video service. The Machine Learning (ML) methods are used to classify a QoE dataset collected using a real testbed experiment. Six classifiers are evaluated, and we determined the most suitable one for the task of QoS/QoE correlation.

The analysis of the users' profile provides vital information, which can help service providers in managing their resources efficiently, by analysing users' behaviour and expectation. The datasets are also used to investigate the influence of different QoS parameters on the user's profile to achieve the best QoE for multimedia video services. The comprehensive study of user's profile in the perspective of different factors, makes the network service provider aware of the behaviour and expectation of end users.

In the uncontrolled environment, a tool based on crowd-sourcing is presented, that measures the QoE of online video streaming in real time, as perceived by end-users. The tool also measures important QoS network parameters in real-time (packet loss, delay, jitter and throughput), retrieves system information (memory, processing power etc.), and other properties of the end-user's system. The proposed approach provides the opportunity to explore the user's quality perception in a wider more realistic domain. The chapter contains our contribution in three conference papers [1] [2] [3].

[1]**M.Sajid Mushtaq**, Brice Augustin, and Abdelhamid Mellouk. *QoE: User Profile Analysis for Multimedia Services*. In Proc. of IEEE International Conference on Communications (ICC), Sydney, Australia, June 10-14, 2014.

[2]**M.Sajid Mushtaq**, Brice Augustin, and Abdelhamid Mellouk. *Crowd-sourcing Framework to Assess QoE*. In Proc. of IEEE International Conference on Communications (ICC), Sydney, Australia, June 10-14, 2014.

[3]**M.Sajid Mushtaq**, Brice Augustin, and Abdelhamid Mellouk, *Empirical study based on Machine Learning Approach to Assess the QoS/QoE correlation*. In 17th European Conference on Network and Optical

3.2 Metrics Affecting the QoE

QoE is very subjective by nature, because of its relationship with user's point of view and its own concept of a "good quality". The ability to measure QoE would give network operators some sense of the contribution of the network's performance to the overall customer satisfaction, in terms of reliability, availability, scalability, speed, accuracy and efficiency. As a starting point, it is necessary to identify precisely the factors that affect QoE, and then try to define methods to measure these factors. We categorize these factors in three types, as follows.

3.2.1 Network Parameters

QoE is influenced by QoS parameters, which highly depend on network elements. Key factors are packet loss rate, jitter and delay. The impact of each individual or combined factors lead to blocking, blurriness or even blackouts with different levels of quality degradation of video streaming.

Packet Loss

Packet losses have a direct effect on the quality of video presented to end users. Packet losses are occurring due to the congestion in the networks and late arrival of packets at application buffers. If packet loss is occurring, then it becomes difficult for the video decoder to decode properly the video streaming. This results in the degradation of video quality.

Jitter

Jitter is another important QoS parameter, which has a great impact on video quality. It is defined as the variance of packet arrival times at the end-user buffer. It occurs when packets travel on different network paths to reach the same destination. It causes jerkiness and frozen video screens.

However, the effects of jitter can be nullified or reduced, to some extent, by adding a large receiving buffer at the end user and delay the play out time of the video. Nevertheless,

Communications (NOC 2012), Barcelona, Spain, June 20-22, 2012.

when packets arrive out of order, after the expiration of a buffering time this packet is discarded by the application. In this context, jitter has the same influence as packet loss [104].

Delay

Delay is defined as the amount of time taken by the packet to travel from its source until its reception at the destination. Delay has a direct influence on user perception while watching the video. If the delay exceeds a certain threshold, then its effect is a freeze and lost blocks of video. The threshold of delay values varies according to the nature of the multimedia service.

3.2.2 Video Characteristics

The characteristics of video are defined in terms of frame and resolution rate, codec and types of content. The impact on the users' satisfaction by reducing bitrate of video streaming services according to the available bandwidth is presented in [55]. The video content types can also influence users' opinions. In case of "interesting" video contents, a user will be more tolerant, and low quality will not influence user's experience as much as in case of a boring content. In [73], authors found that if users show enough interests in the video content, then they can accept even an extremely low frame rate. In this study, a group of participants interested in soccer were selected. The participants gave a very high acceptable rate (80%), although they watched a video with only 6 frames per second. This result clearly shows that if there is a sufficient interest in the topics, then the human visual system can tolerate the relatively gross interruptions and users can tolerate a very low quality video streaming.

Uncompressed video requires a large amount of storage and bandwidth, to be streamed over a network. Therefore, a large number of video codecs were developed (H.262, H.263, H.264, WVID, WMV3, etc) to compress the video in an effective and efficient way, so that acceptable quality of videos can be maintained. Each codec has its own standard way to compress the video contents, providing various video quality levels. The quality levels of video codecs explain the important impact of codecs on users' perceptions.

Generally, the user's interest is measured by monitoring the access's frequency of a specific video (e.g. on Internet). However, this approach is unsuitable to represent the users' interest and preference for the video content. The optimize method to measure the user's interest for a specific video is to record the number of clicks and stays in time. The total time that the user spends in watching the video, provides the significance information about the user's interest.

3.2.3 Terminal Types

The consumers' electronic devices expand largely with the rapid growth of new advancements in telecommunication industries, and they offer a large number of products available for modern multimedia services. These new generation devices are present in different sizes, processing power, advanced functionality, usage and so many other aspects. The different kinds of terminal devices face another problem i.e. the aspect ratio of the end user device and available video content. Internet browsers have the capability to provide the relevant information about the device properties such as screen resolution, operating system, browser type, etc. These key informations can be used to find out the impact of various system parameters on the end user's QoE. It is possible to analyse the impact of different terminal devices on the end user's QoE by using target test sets of different end user devices. The terminal devices can be classified into three categories: Personal Computers, Television (TV), Mobile devices, and . All these terminal devices influence user satisfaction while using video streaming services. For example, it is pointless to send HD video streaming on a low processing terminal equipped with a small screen.

Television (TV)

A tremendous growth is observed in the television market. The companies are offering different TV models with amazing features. These features can be summed up as follows

- Screen Size (40, 65 inch)
- Size (WxHxD, e.g. 1062x700x46.9 mm).
- HD format (720p, 1080i, 1080p).

- Color System.
- TV Type (LED, Plasma)
- 3D Capable.
- Support for Tablet, Smart Phone and other devices.

Computers

Currently, there are a lot of different categories of computer available in the market. It has become hard for people to select the perfect computer, because each computer provides different features. In fact, it is difficult to select the right model, as it all depends on the use of the computer to achieve the desired goal. Some users prefer to have the good performance while on the other hand some give high priority for portability. In case of laptop computer devices, there are also other features that must be considered like battery life, gaming performance and screen quality. The important elements of computer are given below

- Screen Size (e.g. 17 inch).
- Thin screen.
- Processing Power.
- 3D graphics cards with its own memory and processing power.
- Operating System.
- Memory power.

Smart Mobile Devices

Recent advance research have developed a large variety of smart mobile devices, which are powerful enough to support a wide range of multimedia traffic (e.g. video streaming, VoIP, multiplayer interactive gaming etc.) and also legacy mobile services (e.g. voice, SMS, MMS). These new multimedia applications require high data rates and processing power to enhance the end user experience. The essential elements of the smart mobile devices are given below.

- Display and Size (800 x 1280 pixels, 10.1 inches)
- Processing Power (Quad-core 1.4 GHz)
- Memory size (upto 64 GB)
- Stereo Sound quality.
- Video output support.
- Wireless Connectivity (e.g. WiFi, GSM, UMTS, LTE/LTE-A)
- Battery life

3.2.4 Psychological

The QoS network parameters (packet loss, delay, and jitter) use to ensure service guarantees in order to maximize the application performance. However, the QoS fails to determine an important element of human perception about the current service, since the behaviour of the human being is hard to predict. It becomes necessary for network service providers to take into account a large number of parameters and metrics that directly reflect the user's emotional behaviour, to find the adequate quality level for multimedia services.

In case of specific multimedia service, the human perception varies from one person to another. The estimation of quality level depends on many factors, which are related to this person's preferences and surrounding environment. Some of these factors are classified as follows:

- User characteristics (age, sex, background knowledge, language, familiarity with the task).
- The situation characteristics and viewing conditions (noisy space, simultaneous number of users, at home, in a car).
- The user's behaviour and his/her attention on the video being played.

3.3 Machine Learning Classification Methods

We use Machine Learning (ML) methods to classify a preliminary QoE dataset that is collected from the laboratory experiment, as described in section 3.4.1. Based on these datasets, we evaluate how ML methods can help in building an accurate and objective QoE model that correlates low-level parameters with (high-level) quality. We evaluate six classifiers and determine the most suitable one for the task of QoS/QoE correlation.

ML is concerned with the design and development of programs and algorithms which have the capability to automatically improve their performance either on the basis of their own experience over time, or earlier data provided by other programs. The general functions provided by ML are training, recognition, generalisation, adaptation, improvement and intelligibility. There are two types of ML, i.e. unsupervised and supervised learning. Unsupervised refers to find the hidden structure in unlabelled data in order to classify it into meaningful categories, while Supervised Learning assumes that the category structure or hierarchy of the database is already known. They require a set of labelled classes and return a function that maps the database to the pre-defined class labels. In other words, it is the search for algorithms that reason from externally supplied instances to produce general hypotheses. It makes predictions about future instances in order to build a concise model that represents the data distribution. In our case we are considering Supervised Learning, and we are interested in classification methods because of the discrete nature of our datasets. We have applied six ML data classification methods on our datasets, which are Naive Bayes (NB), Support Vector Machines (SVM), k-Nearest Neighbours (k-NN), Decision Tree (DT), Random Forest (RF) and Neural Networks (NNT).

Naive Bayes

The Naive Bayes (NB) classifier is a probabilistic model that uses the joint probabilities of terms and categories to estimate the probabilities of categories given in a test document. The naive part of the classifier comes from the simplifying assumption that all terms are conditionally independent of each other in a given category. Because of this independence assumption, the parameters for each term can be learned separately, and as a result this simplifies and speeds up the computation operations [6].

Support Vector Machines

Support Vector Machines (SVM) are a very powerful classification method, used to solve the two-class-pattern recognition problem. It analyses the data and tries to identify patterns so that a classification can be done. The idea here is to find the optimal separating hyperplane between two classes, by maximizing the margin between the closest points of these two classes. SVM classifies data that have the possibility to be linearly separable in their origin domain or not. The simple linear SVM can be used if the data is linearly separable. When the data is non-separable in their original domain through the hyperplane, then it can be projected in an higher order dimensional Hibert space. By using a kernel function, it is possible to linearly separate the data in a higher dimensional space [108].

K-Nearest Neighbors

The k-Nearest Neighbours (k-NN) method is an instance-based ML method and it is considered a very simple method as compared to all other ML classification methods. In supervised statistical pattern recognition, the k-NN method often performs better than other methods. There is no need of prior supposition of distribution, when the training sample is drawn. It works in a very simple and straightforward way: to classify any new test sample, it compares the new test sample with all other samples in the training set. The category labels of these neighbours are used to estimate the category of the test sample. In other words, it calculates the distance of the new test sample with the nearest training sample, and then at this point finds out the classification of the sample [53].

Decision Tree

Decision Tree (DT) is a method used to create a model to predict the value of a target variable based on several input variables. The structure of DT consists of the following elements: (1) internal nodes, that tests an attribute; (2) branches, corresponding to attribute values, and (3) leaf nodes, which assign a classification. Instances are classified by starting at the root node, and based on the feature values, the tree is sorted down to some leaf node. It is a simple classifier which can efficiently classify new data and compactly store them. It has the capability of reducing complexity and automatically features selection.

DT has build-in property to estimate the suitable features that separate the objects, which represent different classes. The information about the prediction of classification can be easily interpreted, thanks to its tree structure. Finally, the accuracy of DT is less affected by user-defined factors as compares to the k-NN classifier [88].

Random Forest

Random Forest (RF) is an ensemble classifier, that uses multiple models of several DTs to obtain a better prediction performance. It builds on many classification trees and a bootstrapped sample technique is used to train each tree on the set of training data. This method only searches for a random subset of variables in order to find out a split at each node. For the classification, the input vector is submitted to each tree in the RF, and each tree votes for a class. Finally, RF chooses the class which with the highest number of votes. It has the ability to handle larger input data sets than other methods [7].

Neural Networks

A Neural Network (NNT) is a structure of a large number of units (neurons) linked together in a pattern of connections. The interconnections are used to send signals from one neuron to the other. The calculation by neural networks is based on the spread of information between basic units of computation. The possibilities of each one are small, but their interconnection allows a complex overall calculation. The behaviour of a neural network is determined by its architecture: number of cells, how they are connected and the weights assigned to each connection. Each connection between two neurons is characterized by its weight, that measures the degree of influence of the first neuron on the second one. The weight is updated during a training period. This method has the ability to solve multivariate non-linear problems. Its performance is degraded when it is applied on a large number of training datasets [7].

3.4 Experimental Environment for QoE Assessment

In general, the QoE assessment is done by using the subjective method, because it tries to match the real perception of users while using a service. Generally, two distinct approaches are available to collect QoE datasets: a crowdsourcing, and a controlled environment approach. In crowdsourcing, one assigns the video testing task to a large number of anonymous users who can participate from different regions of the world from their own environment. Our proposed crowdsourcing approach is presented in section 3.4.3. In parallel to the crowd-source approach, there is an orthogonal approach in which the experiment environment is totally controlled.

Controlled Environment Approach

The controlled environment approach is a laboratory test environment, which is specially designed to fix the environmental factors that can influence the user's viewing experience. International Telecommunication Union-Telecommunication (ITU-T) defines the recommendation to setup the laboratory test and describe the criteria for selecting the participants to conduct the test. The ITU-T recommendation [51], has provided the guidelines to conduct the subjective tests in a controlled environment, including the selection of participants who represent the users of a service. Indeed, to obtain the subjective notation according to ITU-T recommendation [52], participants should be non-experts, in the sense that they should not be directly concerned with image or video quality as part of their normal work. This approach has the following advantages

- The testing environment is totally under control.
- Easy to monitor the influence of an individual parameter.
- Freedom to select the participants who belong to different background, profession, age group, etc.

The controlled environment approach also has some limitations to assess the performance of QoE.

- It is a time consuming test.
- Limited number of participants, who are willing to spend time in the laboratory test and express their perception of quality for the video service.
- It is an expensive approach, in order to buy the special equipments and apparatus to conduct the test.
- It is difficult to setup the particular laboratory environment in order to resemble the real world environment.

Crowdsourcing Environment Approach

The crowdsourcing environment is an alternative to the laboratory testing approach for assessing the QoE of video service. In this approach, a testing task (e.g. video quality) is assigned to a large number of anonymous users who can participate from different regions around world, via the Internet. It is an efficient approach in which collected datasets represent the opinion of a large number of participants on their quality experience. There are some advantages of this approach, which are;

- Provides an open environment that represents the real user's QoE while using the service.
- Helps in gathering the large amount of QoE data for analysis.
- Allows the remote participant of a large number of anonymous participants.
- Collects QoE parameters in real time.
- Completes the testing task within a short period of time.
- Saves the cost of setting a real-world environment and expensive equipment.

The crowdsourcing environment approach also has some disadvantages in order to assess the performance of QoE.

- Provides an un-controlled environment, which represents the real user's environment

- Different environment for each participant.
- Requires installation of software at end-user device.
- Requires some description or training for each participant in order to conduct the testing task.

3.4.1 Testbed Experiment

We conduct a testbed experiment to analyse the impact of distinct parameters on users perceived quality in video streaming, a subjective test is carried out with the participation of 45 persons. The participants watch the video streaming and rate the quality of the different videos.

In this testbed experiment, the QoS parameters (packet loss, jitter and delay) are varied in a fully controlled manner. Further, their influence on user perception is recorded in the form of a MOS. In addition, another parameter is taken under observation, the conditional loss. The conditional loss reflects the loss probability of the next packet, given that the current packet has already been lost. As most real-time applications exhibit a certain tolerance against infrequent packet losses, this metric helps in concentrating losses on a single part of the sequence, which makes the losses occasional. For our experiment, the relevant parameters and their selected values are given in Table 3.1.

Table 3.1 – QoS Metrics

Parameters	Value
Delay	0ms, 30ms, 60ms,100ms 120ms
Jitter	0ms, 4ms, 8ms, 16ms, 32ms
Packet Loss	0% to 5% with a step of 0.5%
Conditional Loss	0%, 30%, 60%, 90%

In this experiment, we consider the users participation according to ITU-R Rec. BT.500-11 [52]. Indeed, to obtain a subjective notation according to this recommendation, participants should be non-experts, in the sense that they should not be directly concerned with image or video quality as part of their normal work. User characteristics are also stored for analysis purposes, which include user's participant profile like age, gender, familiarity with video streaming, and interest in video content as presented in Table 3.2. End-user devices are Mobile, Tablet, Notebook, Samsung HD Screen, Dell desktops with Intel core duo processor, and a display size set to 1024×740. Mozilla Firefox is used as the Web navigator.

Table 3.2 – User Characteristics

Users Profiles	Values
Age	18 to 30 years
Gender	Male , Female
Familiarity with the video streaming	Rarely, Weekly, Daily
Interest in the content	Interested, Not Interested

There are 25 HD and Non-HD video streams selected for this experiment, with different motion complexities (high, alternating, and low) but with same frame rate (25 frames per second) and video codec (H.264). These videos are related to different fields of interests (e.g. politics, sports, news, social life, commercial ads, and animated cartoons). In our experimental analysis, we used NetEm as a network emulator to control QoS parameters. This tool can emulate the properties of Wide Area Networks (WAN), and its functionalities are evaluated in [60] .

Experimental Setup

Generally, the laboratory experimental setup consists in three important elements: a video streaming server, a video client, and the Network Emulator (NetEm), which emulates a core and cloud network. This basis experimental setup is illustrated in Figure 3.1. The traffic flows between the server, and the client is forwarded via the network emulator. The emulator introduces artificial delay, jitter and packet loss within a dedicated connection. In the example in Figure 3.1 , the client sends the request message to the video server and in response, the requested video is sent to the client via NetEm. In the end of video, the client provides its feedback as the perceived video quality in the form of MOS score, which is stored in a SQL database.

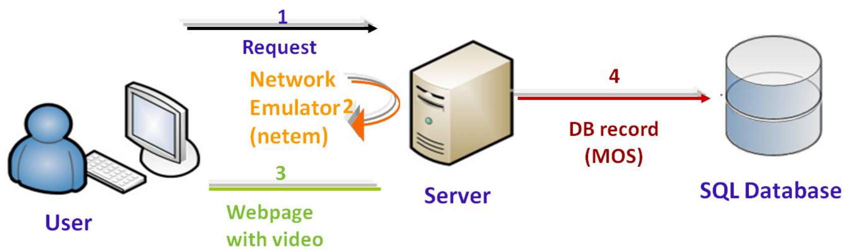


Figure 3.1 – Example: Basic Testbed Setup

Our experimental setup is shown in Figure 3.2. We have stored 25 videos at the server side, and the client can reach them through a private Web site. The client device (either wireless or wired) connects to the Web site to read the description of the experiment and provide the personal information (age, gender, etc.). Users are unaware of the QoS parameters' settings on the videos, and they are asked to rate the perceived quality (in the form of MOS score) after watching each video. The client side consists of different devices which are; desktop devices, Tablet, and Mobile; while the streaming server and the shaper (NetEm) are configured on a Linux OS. The resultant QoE of each video is stored in the database, as a MOS score.

In this experiment, a total of 45 users are participating in which 20 are female and 25 are male participants. Most of them belong to the age group ranging from 18 to 30 years old. We collected $25 * 45 = 1125$ samples in our database, which means that we have 1125 different combinations of all settled parameters, associated with a MOS value for each combination. However, we reduced this number after a deeper look over on the dataset, to average repeated lines and try to eliminate parasite ones.

A laboratory-based test is a time consuming study, but it is easy to investigate the influence of each factor on a desired service. In our experiment, we have collected the suitable dataset in order to investigate the impact of different factors on QoE, as users' profile. The number of participants and video contents in our testbed is good enough as compare to [79], where only one video clip and ten participants conduct the laboratory test.

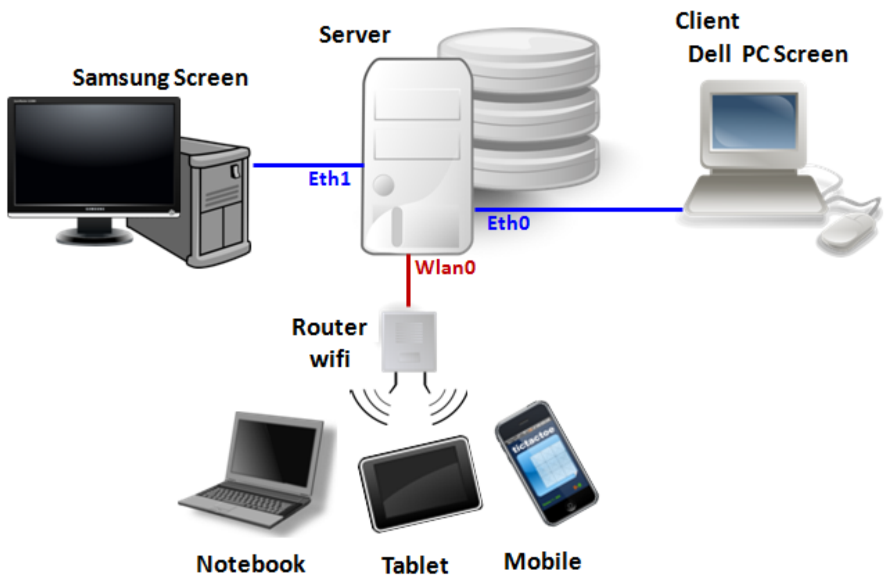


Figure 3.2 – Experimental Setup

Initially, datasets resulting from the controlled experiment were processed and cleaned from any parasite information. Therefore, we have a dataset that is ready to apply for data analysis. As an input to our ML tool, we are considering all nine parameters, which are gender, frequency of viewing, interest, delay, jitter, loss, conditional loss, motion complexity and resolution. In order to minimize biases, we perform 4-fold-cross-validation to estimate the error rate efficiently, using the following procedure: a single sub-sample is chosen as testing data, and the remaining 3 sub-samples are used as training data. This procedure is repeated 4 times, in which each of the 4 sub-samples is used exactly once as the testing data. All results are averaged and a single estimation is obtained. The modelling process is done by using the six classifying models to find out the best one and offers the best model. Recall that these six classifying models are: Naives Bayes (NB), 4-Nearest Neighbour (4-NN), Support Vector Machine (SMV), Decision Tree (DT), Random Forest (RF) and Neural Network (NNT). We use the WEKA tool to run those different algorithms on the dataset. This tool gives information about the classification model that was generated, along with its performance and imperfection with detailed averaged statistics. We consider the mean absolute error rate to compare the error rate between the different models. The results are illustrated in Figure 3.3. In terms of classification this figure shows that DT has the minimum absolute error rate, with a value of 0.126, followed by the RF model with 0.136. SVM has the highest error rate with 0.26. The results clearly depicts that the DT model and RF model are the most reliable models on the current datasets.

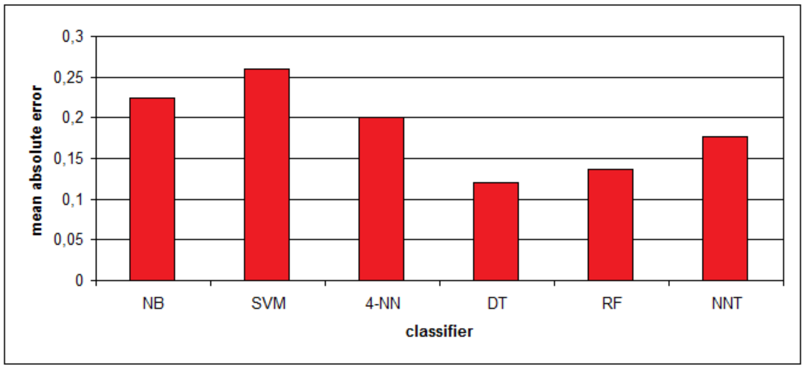


Figure 3.3 – Mean Absolute Error Rate for Six Classifiers

To choose the best model, we also perform an instance classification test on the six algorithms, in terms of the number of correctly classified instances. Figure 3.4 shows that two methods correspond to the best classification: RF with 74.8% of correctly classified instances, followed by the DT model with 74% of correctly classified data. The worst model is 4-NN model with 49% of correctly classified instances. These results again clearly demonstrate that the DT and RF models are the best models, according to our datasets.

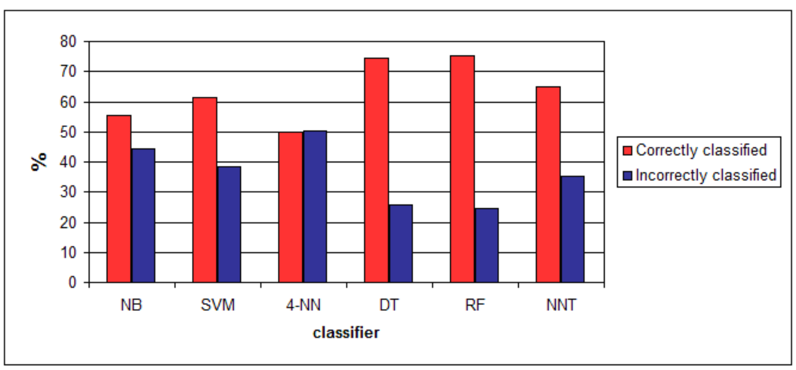


Figure 3.4 – Instance Classification

To find more details about the models and their classification errors, we compare the efficiency of DT and RF models. The efficiency of these models is evaluated by measuring the statistics analysis data about classification, as presented in Table 3.3.

Table 3.3 – Average weighted for RF and DT models

Model	TP	FP	Precision	Recall	F-Measure
RF	0.753	0.078	0.752	0.753	0.752
DT	0.743	0.084	0.748	0.743	0.745

We consider five statistical metrics to compare the performance of DT and RF models, which are: True Positive (TP), False Positive (FP), Precision, Recall and F-measure.

- **TP (True Positive)**: occurs when a statistical test rejects a true hypothesis. The best value for this measure is 1.
- **FP (False Positive)**: a false value means rejecting the hypothesis. Its value should be close to 0, which means the model works well.
- **Precision**: is the probability when a (randomly selected) retrieved result is relevant
  Precision = TP/ (TP+FP)
- **Recall**: is the probability when a (randomly selected) relevant document is retrieved in a search
  Recall = TP/ (TP+FN)
- **F-measure**: is a measure of a test accuracy, where an F1 score reaches its best value at 1 and in worst case its value is 0
  F-measure = 2 * (Precision * Recall) / (Precision + Recall)

The results of a classification can be negative or positive. If the results of the test correspond to reality, then one considers that a correct decision has been made. However, if the result of the test does not correspond to reality, then an error has occurred. According to these metrics, we conclude in Table 3.3 that RF is slightly more suitable than the DT model for QoS/QoE correlation.

3.4.2 User Profile Analysis

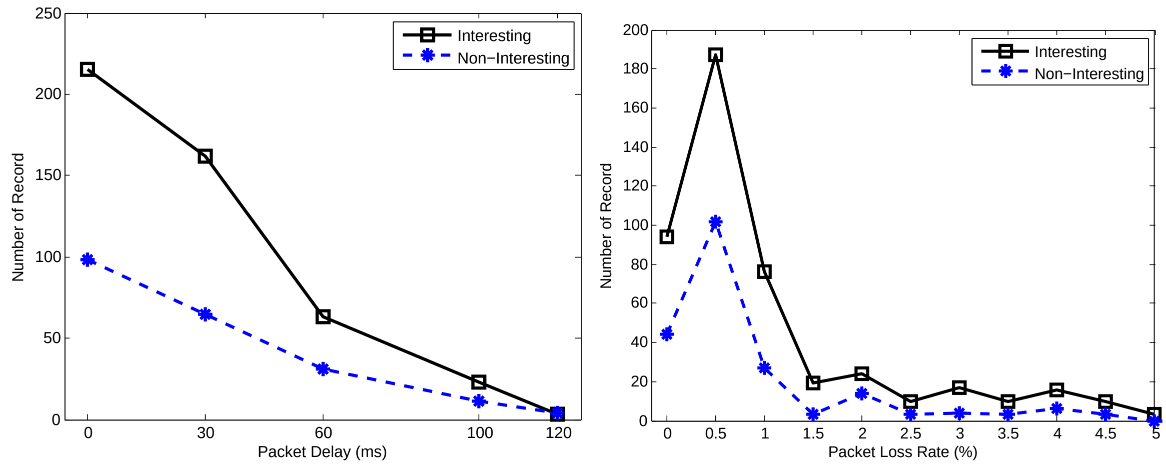
The subjectively collected dataset is also used to analysis the users' profile that provide vital information and can help the service providers in managing their resources efficiently, by analyzing users' behavior and expectation. The comprehensive study of users' profile provides significant insights on all metrics that influence the QoE (network parameters and video characteristics). Wireless and wired networks have different infrastructure aspects (reliability, availability, etc.) but the analysis and evaluations of users' profile are equally important for both networks. Our analytical study provides an opportunity to network service providers to obtain high user satisfaction by providing a service level that matches customers' usage patterns and expectations. Two cases are considered that provide significant information based on user's profile and other parameters.

Case 1. Interesting and Non-Interesting Video Content

In the first case, we consider the videos' content that relate the user's interest and non-interest. By considering the user's interest into the video content, we observe how the QoS parameters influence the user's interest. In this scenario, we only consider the dataset that has the MOS score equal or more than 3 because users are quite satisfied on these scores. Figure 3.5a compares the impact of delay on interesting and non-interesting video contents. It can be seen clearly that when delay is very low (0 ms), then a large number of users show high satisfaction in the video content with high MOS score. When the value of delay is increased (more than 30 ms) then number of user to watch the video's content are starting to decrease quickly. As a take-out, this results shows that it is necessary for network service provider to keep delay under 30 ms for video streaming. By considering this delay threshold, the network service provider still gets high user's satisfaction with efficient utilization of network resources. Figure 3.5b, represents the influence of packet loss rate on interesting and non-interesting video content. It is important to notice that the number of dataset records that are categorized as "non-interesting" are much fewer than the records categorized as "interesting". The results show that when network operators target a high user satisfaction then they must provide a low packet loss rate (less than at least 1%).

Case 2. Frequency, HD and Non-HD Video Content

In this case, we are considering the three important parameters that represent the behaviour and expectation of end users while watching the video streaming. We are analyzing the



(a) Impact of Delay (b) Impact of Packet Loss Rate

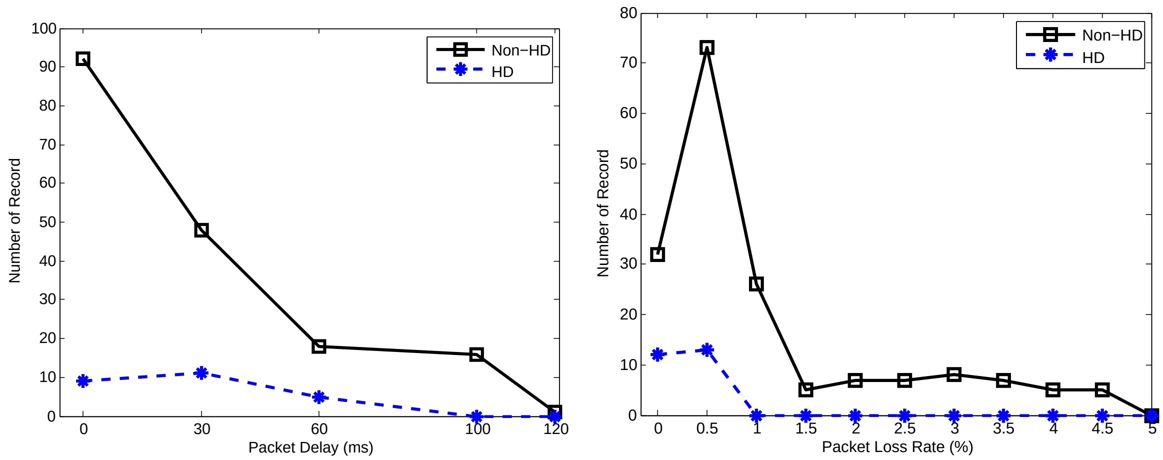
Figure 3.5 – Interesting and Non-Interesting Video Content

datasets that relate to the frequency of watching HD and Non-HD video content; again, we consider only the records that have MOS score equal or greater than 3. Regarding QoS parameters, we base our analysis on delays and loss rates.

Figure 3.6a, compares the impact of delay when users rarely watch Non-HD and HD video content. In case of Non-HD video content, Figure 3.6a, illustrates that the occasional watcher of video streaming are less sensitive for delay, but in case of HD video streaming, the users are more sensitive to delay. A small number of viewers are recorded for HD videos against the rarely video viewers. It is necessary for a network service provider that HD video streaming should have a low delay to achieve a high user satisfaction.

Figure 3.6b, compares the impact of packet loss rate when users rarely watch Non-HD and HD video content. In case of Non-HD video, many users tolerate a packet loss rate until 1%, but the increase in packet loss rate decrease the number of users watching the video streaming. Whereas, in case of HD video streaming, users are more sensitive to packet loss rate and do not tolerate a packet loss rate greater than 0.5 %.

Figure 3.7a, shows the impact of delay when users weekly watch Non-HD and HD video contents. The results clearly show that users are less sensitive to delay as compared to users who rarely watch Non-HD video streaming. The results depict that a few number



(a) Impact of Delay

(b) Impact of Packet Loss Rate

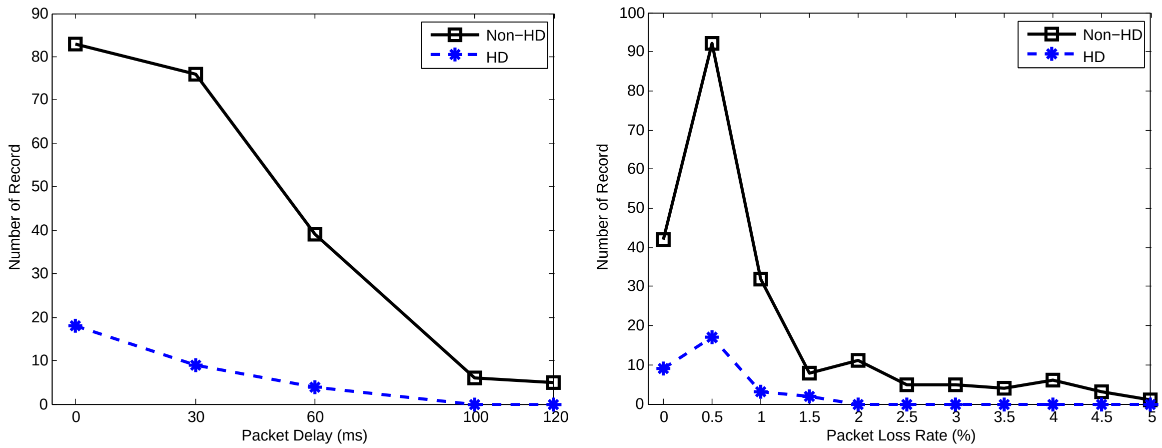
Figure 3.6 – User Rarely Watch the HD and Non-HD Video Content

of users watched the HD video streaming like users' rarely watching video. The users who weekly watch Non-HD and HD video streaming have more tolerance than the users who watch rarely video.

Figure 3.7b illustrates the impact of packet loss rate when users weekly watch Non-HD and HD video content. Non-HD videos content have the largest number of viewer's record, whereas HD videos content has the smaller number of viewers. In case of Non-HD videos, when packet loss rate is lower than 0.5%, then a large number of users watch videos' content, but it decreases when packet loss exceeds 1%. On the other hand, weekly watchers of HD videos are sensitive to packet loss rate. The results indicate that network service provider must optimize its network in order to keep the packet loss rate equal or less than 1%, for getting higher users' satisfaction.

Figure 3.8a, compares the impact of delay when users daily watch Non-HD and HD video content. It is noticeable that a large number of user's record fall within this category. In both cases, the results clearly show that daily videos watchers (to some extent) are less sensitive to delay as compare to users who watch the video streaming rarely or on a weekly basis.

Figure 3.8b, depicts the impact of packet loss rate when users daily watch Non-HD



(a) Impact of Delay

(b) Impact of Packet Loss Rate

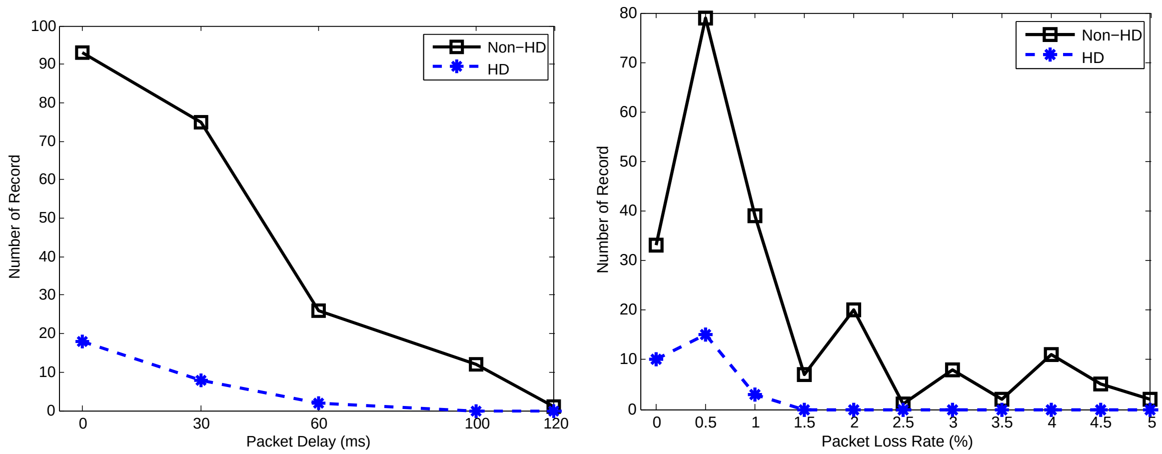
Figure 3.7 – User Weekly Watch the HD and Non-HD Video Content

and HD video content. A large number of viewer's record falls within this category. The results clearly show that users are more tolerant to packet loss rate as compared to users who rarely or weekly watch videos. By contrast, HD video's viewers have less tolerance for packet loss rate, and the small numbers of records are found in results as like the videos' watchers on the weekly basis.

It is observed when users show interest in videos' content then their tolerance more than non-interesting videos' content. However, in case of HD video content, users are more sensitive in the delay and packet loss, while for Non-HD videos' content the users have more tolerance levels.

3.4.3 Crowdsourcing Method

Two subjective testing approaches can be used for assessing the QoE of video service: controlled environment, and crowdsourcing environment approach. The crowdsourcing emerges as an efficient method that performs the subjective testing in the real world (the user's own environment). In crowdsourcing, users can participate remotely from all around the world by using their own devices.



(a) Impact of Delay

(b) Impact of Packet Loss Rate

Figure 3.8 – User Daily Watch the HD and Non-HD Video Content

Crowdsourcing Framework

The living laboratory is a new concept that is used in different researches by focusing on the user experience. It tries to bring the laboratory to the volunteers in its realistic context. The goal of our proposed work is assessing the QoE in real-time by building the larger dataset. This objective is achieved by developing the tool that uses the Internet as our simulation platform, and provides the opportunity of remote participation of users. The crowdsourcing framework tool is based on two parts; first, it detects the video on the website, and when the video ends, it ask for user about the perceived quality of video streaming. The second part measures and stores the real time factors that influence the users' QoE, e.g. network QoS parameters, terminal device characteristics, etc.

The framework records the degree of users' satisfaction, in the feedback form while using the video services on the Internet. The feedback form is shown in Figure 3.12. The proposed framework tool is tested by using the YouTube web portal, because it has the largest cloud network for video delivery, and it is considered as one of the most prominent videos streaming website. According to [64], in May 2010, 14.6 billion videos per day were served by the YouTube. The framework detects the presence of a YouTube video on a Web page, and automatically adds a button on which the user is asked to click, whenever he/she is unhappy while viewing the video. The plugin tool also stores the QoE values, which are used to build a large dataset of heterogeneous users, devices and situations. Figure 3.9 shows the framework structure in which remote users participate via an IP network (Internet).

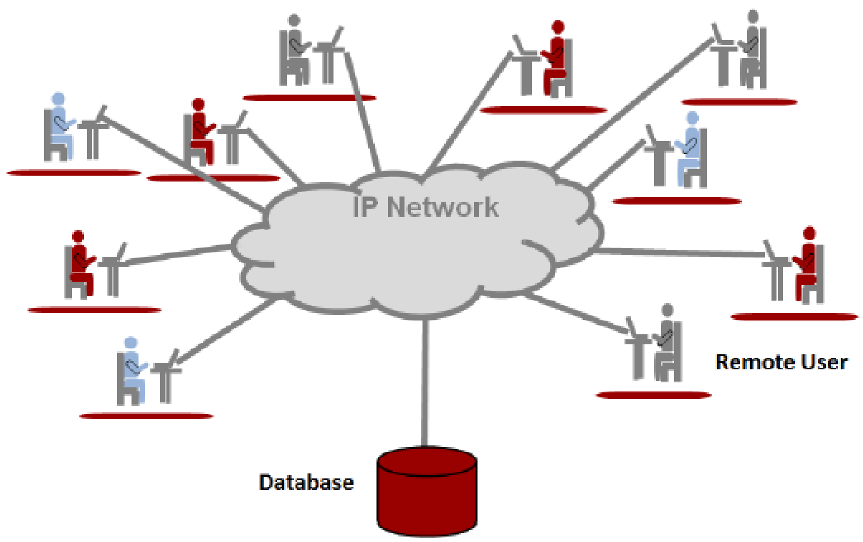


Figure 3.9 – Crowdsourcing Framework

The framework setup contains the following items:

- A Firefox plug-in is developed and installed on end users' devices to run the real-time experiment. In particular, the plug-in detects the presence of a video in a Web page, and automatically adds a button, on which the user can click whenever a user is unhappy about the video quality, as shown in Figure 3.11 .
- A large number of remote volunteers are invited to watch video sequences online, on their machines. Users can watch any video on the YouTube platform.
- Each video can have different characteristics and experience various, realistic QoS parameters.
- In viewing each video, the terminal properties and data on system processing are measured and recorded in a local database.

- During the video, and in the end of it, users rate the quality of video (MOS) according to their perception.
- All feedback informations are stored in the database for future analysis of QoE parameters.

Framework Architecture

Figure 3.10 shows the architecture of our framework. It is based on two major modules: Firefox extension and Java application. Initially, all collected informations will be stored in a local database at the user terminal device. Later, these collected datasets will be transferred to a remote server.

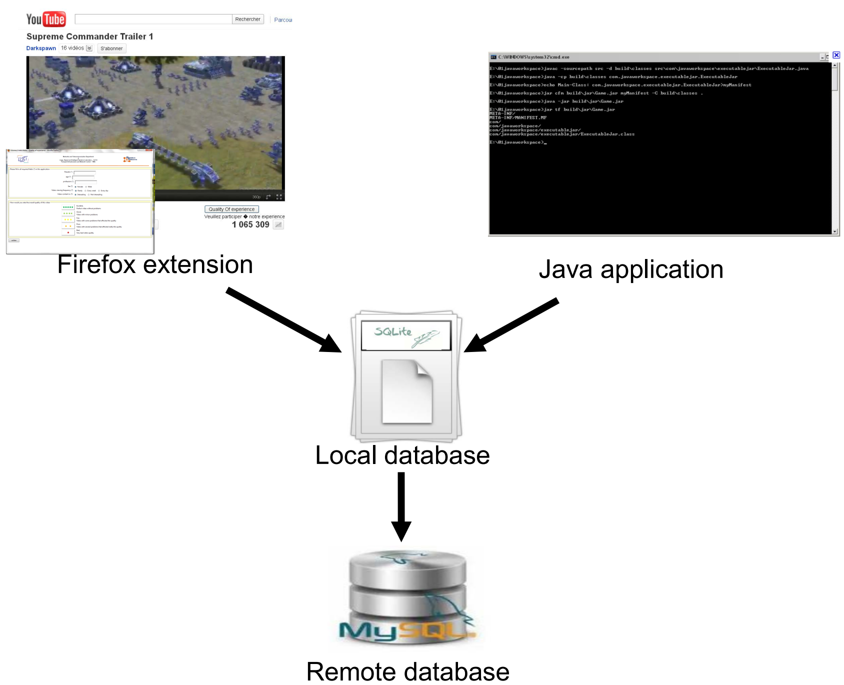


Figure 3.10 – Crowdsourcing Framework Architecture

Firefox Extension

The FireFox extension is developed in Javascript, which is a prototype-based an object-oriented scripting language, that is widely used in web development. It represents a complement to XML language in a Firefox extension, in order to enhance, enrich and improve the graphic interface of an application. The main functions of our Firefox extension are followings,

- On web page loading, it analyzes the page and if a YouTube web page (e.g. YouTube) is found, then it insert the button at the bottom of the online video, as shown in Figure 3.11.

- Add the "QoE Feedback" menu item under the "Tools" menu in the Firefox menu.

- When the user clicks the button, a feedback form will open, in order to take the feedback from the user, and store the information in the local database, as shown in Figure 3.12.

- It also stores the information related to video duration, video ID, video content type, operating system version, and screen resolution.

Figure 3.11 – Framework Implementation

In the subjective approach, the most common way is to ask the user opinion about the video streaming quality, and other relevant questions for analysis the user's QoE. In our framework, the user's feedback form is used for this purpose, and it is shown in Figure 3.12. It contains the following fields; Name, Age, Profession, Sex (male or female), Video viewing frequency (rarely, every week and daily), Video content (Interesting, Non-Interesting), and User quality experience (MOS).

We present the framework test only on YouTube website, but our Firefox extension is also compatible with DailyMotion and TF1 (french live video streaming content provider). In the future, we plan to make the plugin compatible with a large number of video streaming websites. We also plan to make it work on different platforms and streaming protocols (e.g. DASH, HLS), in order to capture the real user experience of perceived video quality.

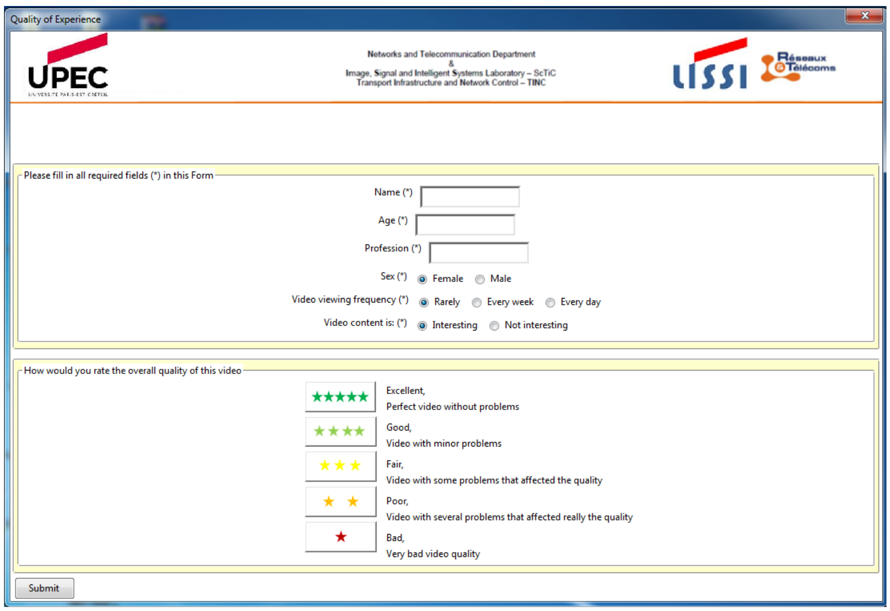


Figure 3.12 – User Feedback Form

Java Application

In parallel to the Firefox extension, we developed an application in the Java language that runs as a background process for storing the important information while a user is viewing an online video. The main advantage of this application is that it works for any video streaming website, if it uses the TCP protocol as a transmission layer protocol. It monitors and collects all the information by periodically (5 seconds) checking the status of the terminal device and examining the packets flow related to video streaming. This module application monitors the real time packets exchanged between the video server and the user, while viewing the video streaming. It extracts the required information by analysing the packets without storing them, in order to compute the network performance statistics of QoS (packet loss, delay, jitter and throughput) during the video flow.

The application measures and stores the different characteristic of user terminal device, e.g. CPU model specification, Vendor name (e.g. Intel), Speed, and number of CPU in the terminal device. This part of our framework tool carefully monitors the system performance behavior in terms of memory and CPU usage, while viewing the online video. During the video flow, the CPU's usage measures in Percentage unit that represent the share of CPU power used in terms of following parameters; User processes, System processes, Idle, Wait, Nice, Interrupt, and Combined usage (User+System). The application also measures the memory's usage in Mega Byte (MB), that represents the amount of memory used by the system, and how much is free. Initially, it stores all the information in the local database.

In the future, we plan to add more functionalities in the framework for investigating the influence of more parameters on user perceived quality. In case of video streaming service, the following parameters could be monitored and stored into the local database; resolution, codec, type of content, stalling time, user buffering behavior in terms of rebuffering event, required minimum data in the buffer before resuming the playback.

In the end of crowdsourcing test, when all parameters are extracted from the two modules (Java application and Firefox extension), the collected datasets are transferred from the user's terminal to a distant server for investigating the user's QoE.

3.5 Conclusion

In this chapter, two different approaches are discussed to gather datasets for assessing the QoE of video service, and analyse the impact of different parameters. These approaches are controlled, and crowdsourcing environment approach. A testbed experiment is setup to measure the influence of different parameters on the user perceived QoE, while watching the video service. The impact of different parameters (QoS parameters, video characteristic, device type, etc.) on user perception is recorded in the form of MOS value.

The collected dataset is used to investigate the correlation between QoS and video QoE. Six ML classifiers are used to classify the collected dataset. In case of mean absolute error rate, it is observed that Decision Tree (DT) has a good performance as compared to all other algorithms. An instance classification test is also performed to select the best model, and results clearly show that performance of RF and DT are approximately at the same level. Finally, to evaluate the efficiency of DT and RF, a statistical analysis of classification is done, and results show that RF performs slightly better than DT.

The dataset allows us to study the impact of different QoS parameters on user's profile, in order to achieve a high user satisfaction while watching video streaming services. The comprehensive study of users' profile in the perspective of QoS parameters, gives useful information for network service providers to understand the behaviour and expectation of end users. The analysis shows that interesting videos' content have more tolerance than non-interesting videos' content. Similarly, the users for HD videos' content are more sensitive in the delay and packet loss, while for Non-HD videos' content the users have more tolerance levels. Based on users' profile analysis, the network service provider can efficiently utilize their resources to improve user satisfaction.

In case of crowdsourcing, a new application tool is proposed that can be used to investigate the users' QoE in real-time environment. After watching the video, the tool takes the user's feedback by automatically opening feedback form. The user can also open and record the feedback, whenever the user wants to express his opinion of video quality by clicking the feedback button at the bottom of the video display screen. The tool can monitor and store the real time performance parameters of QoS (packet loss, delay, jitter and throughput). Instead of QoS networks, the tool also measures the real time performance characteristics of the end user device in terms of system memory, performance capacity, CPU usage and other parameters.

This chapter tackles the problem of assessing the QoE for video streaming by considering the influence of different parameters based on subjectively collected dataset. Our collected dataset points out the useful information about the video quality, which is a crucial step towards developing an adaptive video streaming method that changes the video quality based on network parameters and client device's properties. In the next chapter, we consider the three influential QoS parameters (bandwidth, buffer, dropped frame rate) that have a significant impact on the user's QoE for HTTP based video streaming. A client-side HTTP based rate adaptive method is proposed, that selects the most suitable video quality based on three QoS parameters.

Chapter 4

Regulating QoE for Adaptive Video Streaming

In the previous chapter, different methodologies are described to assess user's QoE for video streaming by considering the influence of different parameters. This chapter extends the investigation of user's QoE in the perspective of three important parameters (Bandwidth, Buffer, and dropped frame rate). This chapter focuses on an adaptive method that can efficiently manage the video streaming traffic according to different parameters in order to regulate the user's QoE.

4.1 Introduction

Video streaming is a main and growing contributor in the Internet traffic. This growth comes with deep changes in the technologies that are employed for delivering video content to end-users over the Internet. According to Cisco forecast report, all forms of video (TV, Video on demand [VoD], Internet and P2P) will represent 80% to 90% of global consumer traffic by 2017 [49].

Traditionally, cable and IPTV services provide video service over a managed network as they use the multicast transport, where the required bandwidth is available for maximizing the user Quality of Experience (QoE) (defined in [85]). However, in the age of multimedia technology, a large number of video-enabled electronic devices are made available, with the capacity to support the highest quality video playback. These devices include Personal Computers (PCs), laptop, Smart phones, Tablets, gaming consoles, and Internet-enabled Televisions, etc. Generally, these devices access the video streaming services through unmanaged networks, e.g. Local Area Network (LAN), Wifi hot spots, 3G/4G wireless networks etc.

Internet-based video, also known as Over-the-Top (OTT) services, can be divided into three different categories, such as user-generated content (e.g. DailyMotion, YouTube), professional generated content (e.g. commercial), and movie sales to viewer over the Internet [8]. The content service providers make sure that video contents are available on the Internet in order to gain larger viewer-ship. Generally, video contents are delivered through a Content Delivery Network (CDN), and different CDN architecture are used to improve the performance of the system, reduce network load and enhance the end user perceived QoE. In general content is stored on the servers that scattered all around the world. The CDN algorithm tries to select servers that are close to the client in order to ensure a high-bandwidth video stream. Famous CDN providers include YouTube, Akamai, Netflix, Hulu, etc. The CDN provider uses different mechanisms to select the suitable server to serve the end-user, because it is an important factor that influences user perceived quality of video service. In [113], authors proposed a server selection method that select the server based on the load information of replica servers, while in [38] proposed method uses the minimum Round Trip Time (RTT) from client in order to pick out the suitable server. In [101], authors proposed a QoE-based server selection method that choose the appropriate server by considering the perceived QoE from each candidate server.

Furthermore, the demand of end-users to view the video contents any time on any device over any access network, create new challenges for network operators and CDN providers to deliver the video content on different devices with maximum end-user QoE. Facing distinct network technologies and time-varying network conditions, requires a video rate adaptive method that considers not only network characteristics, but also end user's device's properties to provide the highest quality video streaming to the end-users. To overcome this problem, leading companies Adobe, Microsoft, Apple, and MPEG/3GPP have developed the HTTP based adaptive streaming technologies (see Appendix A) that adapts the video service, according to client and network properties. The adaptive method efficiently shares network resources (bandwidth) among the users, and dynamically contributes in network resource management with high user's perceived QoE.

HTTP video streaming has the advantage that it easily traverses NAT's and firewalls, unlike other media transport protocols such as RTP/RTSP. In HTTP adaptive streaming, the source video content (either a stored file or live stream) is broken into file segments, called fragments, chunks or segments, using the desired format, which contains video codec, audio codec, encryption protocol, etc. Generally, the segment length is between 2-10 seconds of the stream. The segment file consists either in a multiplexing container that mixes the data from different tracks (video, audio, subtitles, etc.), or it can be a single track. The stream is divided into chunks at boundaries of video Group of Picture (GOP), identified by an IDR frame. The IDR is such a frame that can be decoded independently, without looking for other frames, and each chunk does not depend on previous and successive chunks. The file segments are hosted on a regular HTTP server (e.g. Apache server). The client adaptive player requests the appropriate video segment to the server, based on the network parameters and its machine processing state.

Accurate bandwidth estimation is an important task, as it regulates the user's buffer and influences the user perceived Quality of Service (QoS). Generally, bandwidth is estimated by using different information provided by the TCP protocol (e.g. Ack, RTT, etc.). In our proposed method, the video fragment size and download duration are used as the key parameters to estimate the client's bandwidth. The performance of rate adaptive methods are significantly affected by the bandwidth's oscillation. It is necessary not only to estimate the bandwidth but also handle an instantaneous fluctuation of bandwidth in an efficient way. The proposed method can estimate, and manage the bandwidth fluctuation that regulates the user's buffer and copes with a sudden drop of bandwidth.

In this chapter, a client-based rate adaptive method *BBF* is proposed that dynamically selects the appropriate video quality according to network conditions and user's device properties. The network bandwidth significantly affects the video service, as it directly reduces the client buffering that may result in pausing or stalling during video streaming. The buffer length plays a vital role to reduce the influence of dynamic change in bandwidth. The proposed BBF method efficiently deals with sudden dropping in network bandwidth by using new bandwidth metric, and reduces its impact on the buffer level of the end user. The dropped frame rate (fps) is another influential factor that has a negative impact on user's QoE. The BBF method considers three important QoS factors that regulates the user's QoE for video streaming over HTTP, which are: **B**andwidth, **B**uffer, and dropped **F**rame rate (BBF). This chapter is based on our contribution in two IEEE conference papers. [1] [2]

4.2 Adaptive Streaming Architecture

HTTP based adaptive streaming architecture mainly consists in three important components: client, delivery network, and server. Client based adaptive HTTP streaming primarily depends on the adaptive method used by the client player. The main goal of adaptive streaming method is to dynamically select the appropriate video segment based on client device properties and network conditions. Figure 4.1, illustrates an adaptive streaming architecture that is based on system model, describes in section 4.6. Generally, the main elements that regulate video streaming service at the client side consist in following components:

- Player buffer, stores the received video frames from the server.
- Decoder, decodes the received frames from the player buffer.
- Buffer regulator, controls the player buffer length in order to avoid buffer underflow/overflow condition.
- Bandwidth estimator, estimates the network bandwidth and requests the suitable segment to the server.

The client receives the video frames in its player buffer, that are later decoded to display the video stream to the user. The player buffer can contain different qualities of video frames, which influence the user perceived QoE. The decoding process of video frames mainly depends on the available system resources at the user, since some video frames can

[1] **M.Sajid Mushtaq**, Brice Augustin, and Abdelhamid Mellouk. *Regulating QoE for Adaptive Video Streaming using BBF Method*. In Proc. of IEEE International Conference on Communications (ICC), London, UK, June 10-14, 2015.

[2] **M.Sajid Mushtaq**, Brice Augustin, and Abdelhamid Mellouk. *HTTP Rata Adaptive Algorithm with High Bandwidth Utilization*. In Proc. of IFIP/IEEE International Conference on Network and Service Management (CNSM), Rio, Brazil, November 17-21, 2014.

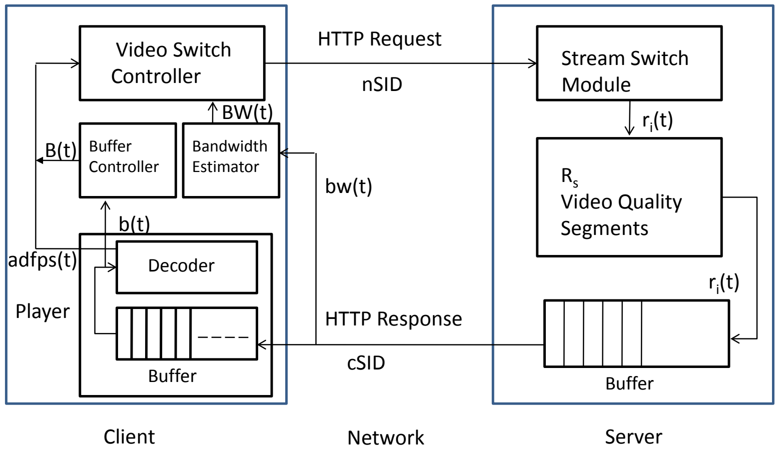


Figure 4.1 – Adaptive Streaming Architecture

be dropped due to insufficient local resources. In case of recorded video streaming, specially when a video has high quality or high-resolution, the decoder lag behind in decoding the required number of frames per second, because it does not has enough system CPU resources that cause the frames dropped. However, player buffer can also drop the video frames when the latency is too high, particularly in live video streaming services. The user's QoE decreases when the number of dropped frames increase, as they are not presented to the user for viewing. In [114], authors use the full-reference model (compare received data with reference data) to study the impact of video frame rate and resolution on user's QoE.

To understand the dynamics of video playback buffer, it is necessary to consider the relationship between available network bandwidth, and video rate in playback buffer as shown in Figure 4.2, where the buffer-size and buffer-filled length are measured in seconds. In [45], the authors proposed the buffer-based adaptive method that use the bandwidth and video rate relationship to avoid the re-buffering. Let consider if one second video is removed from the buffer and playback, then buffer is drained only for one second unit rate. However, when the player is paused then the buffer draining rate will be zero, in other way the buffer draining rate $d(t)$ can be 0 or 1. In this paper, the video segment duration is fixed to 4 seconds (i.e. 4 seconds per segments), and if the client requests the high video rate then it contains larger segment size (in bytes). When high video rate segment $R(t)$ is requested by the client and available bandwidth $B(t)$ is lower than the request video rate then the buffer is filled at the rate $B(t)/R(t) < 1$, and as the result the buffer decreases. If client continuously requests high video quality at a rate greater than network bandwidth, the buffer might be depleted. As a consequence, playback will freeze, and re-buffering event will occur, thus decreasing the client's QoE. However, if network bandwidth is always higher than the requested video rate, then client will never observe re-buffering events i.e. $B(t)/R(t) > 1$.

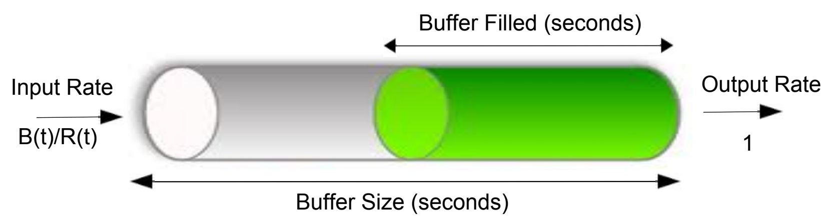


Figure 4.2 – Relationship between Bandwidth B(t) and Video rate R(t) in playback buffer

In adaptive streaming, the video is encoded into different bitrates. The player buffer length $q(t)$ [17],[18] can be modelled by using the following expression:

$$q(t) = \frac{B(t)}{R(t)} - d(t) \tag{4.1}$$

where $d(t)$ is the buffer draining rate, that can be modelled as given below:

$$d(t) = \begin{cases} 1 & \text{playing} \\ 0 & \text{paused} \end{cases} \tag{4.2}$$

where $B(t)$ represents the received rate while $R(t)$ represents the received video level. The player buffer filling rate represents the number of seconds video are stored in the buffer per second. The term $d(t)$ is the draining rate that illustrates the number of seconds video are played per second.

The video playback buffer directly depends on the video rate and available network bandwidth. In this perspective, it is mandatory that the main adaptive streaming controller at the client side consists of two sub-entities that regulate the video streaming service, i.e. buffer regulator and bandwidth estimator.

The buffer regulator tries to maintain the video buffer length within a certain bounded value. It primarily depends on the available network bandwidth: if the buffer draining rate is higher than the bandwidth then buffer will decrease, and empty buffer event occurs, leading to rebuffering stage. In [44], a buffer-based rate adaptation method is proposed that selects and downloads the appropriate video segment, that exclusively based on client video buffer length, and inconsiderate the available system capacity (bandwidth) at the client side. The bandwidth estimator measures the available network bandwidth at the client side. It determines the maximum client capacity to download the video stream rate. Generally, the bandwidth estimator predicts the available bandwidth based on past transmission history. HTTP adaptive video streaming mostly uses TCP as a transport protocol, and the behaviour of TCP during network congestion drastically influence the video quality. The adaptive streaming method should be robust to handle dynamic network conditions. The design of an adaptive streaming method is based on two controller; first select the appropriate video segment that matches the measured available bandwidth, other control the video playback buffer length by using the idle time length between the downloading of two video segments. The general behaviour of these two controllers in adaptive video streaming can be observed in [19], [43], [69].

The delivery network can belong to a private organization that manages its own network for video service (e.g. video conference) or simply open public network (Internet). The adaptive video streaming service uses the public Internet as its underlying delivery network, that is an unmanaged network. The Internet is a collection of diverse networks all over the world and it is constantly changing. The adaptive video streaming method consider the time varying characteristics of Internet to optimize the received video quality for improving the user's perceived QoE. Generally, over the Internet, the video streaming technologies send the video content from the server to the client using the standard delivery HTTP protocol over Transmission Control Protocol (TCP).

The server-side contains a streaming switch mechanism module that selects the proper video quality based on the request received from the streaming switching controller at the client. The server contains different video segments, and each segment has a specific playback duration, normally between 2 to 10 seconds. In case of recorded video, the client initially downloads a file that contains the information about available different video representation or profile at the server, i.e. manifest file. An XML based manifest or SMIL [11] file contains the information about the available video profiles. The client main controller has the full authority that regulates the video streaming and server side just following the order from the client controller.

4.3 Video Encoding

In adaptive video streaming, there are some critical elements in video encoding that should be taken into account for video quality stored at the server. The performance of an adaptive streaming method can significantly affect when the important factors are not considered during the encoding process. The keyframe is a main contribution factor that affects the performance of adaptive streaming method. The BBF method uses the system implementation that is based on Adobe Flash platform, and videos are encoded using the H.264 codec, which contains three type of frames

- I-frames: They are also known as keyframes, that are entirely self-referential without requiring data from other frames. In compression point of view, they are least efficient as compared to other frames (P and B).
- P-frames: They are "predicted" frames. The encoder produce a P-frame by considering only the previous I-frames or P-frames. They are more efficient than I-frames but less than B-frames in terms of compression.
- B-frames: They are bi-directional predicted frames. When encoder produces a B-frame, it considers both the forward and backward frames.

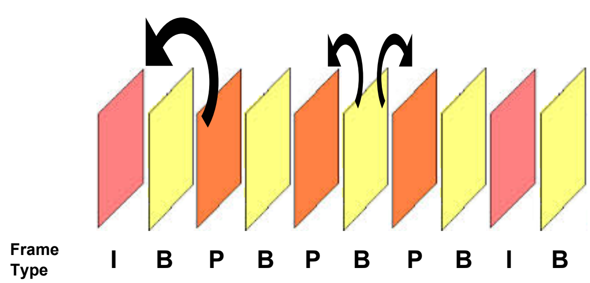


Figure 4.3 – H.264 Frame

The video contents are encoded according to Adobe recommendation [21] by using the Big Buck Bunny video file (YUV format). In case of H.264 codec, the IDR and non-IDR I-frames are considered in different perspective. Instantaneous Decoding Refresh (IDR) are common I-frames that guarantee a reliable seeking, because it allows succeeding frames reference itself and the frames after it, i.e. closed Group of Pictures (GOP). However, a non-IDR I-frame can be considered as an intra-coded P-frames, that referenced by looking the preceding B-frames. The non-IDR I-frames have the advantage that they improve the picture quality, and smooth the P-to-I frame transition by reducing the I-frame flicker. The drawback of non-IDR I-frames is, the decoder has high startup time, and also it reduces the seeking precision.

The adaptive streaming method based on Flash platform only changes the video quality (bitrate) at IDR keyframe intervals, (from here onwards referred to as "keyframe"). The keyframe distance has vital impacts, e.g. seeking performance, decoder startup time (in network streaming), recovery time from network errors, and entire video quality. Generally, keyframe distance is between 2 to 10 seconds. In case of smaller distance (e.g. 2 seconds), the resulting video quality can change more quickly. The keyframe is larger than other frames (P and B frames), and it directly affects the video quality, as it follows the rule 2x rate. Let us consider a case, when keyframe interval changes from 1 to 2 seconds then it will result almost 2x the bitrate for quality improvement, and when changes from 2 to 3 seconds then it will give another 50% quality improvement, and so on. The keyframe distance in frames can be calculated from Equation 4.3, and results are formulated in Table 4.1

$$\text{Keyframe Distance} = \text{Frame Rate Frequency} * \text{Interval in seconds} \quad (4.3)$$

Table 4.1 illustrates mostly used frame rate frequency in terms of different keyframe interval. In case of 60Hz (60fps), when the key frame interval increases from 1 second to 2 seconds and keeps constant all other factors, then the data rate is nearly doubled. Similarly, reducing a half of keyframe interval(e.g. from 4 seconds to 2 seconds) will reduce the video quality by half.

Table 4.1 – Keyframe Distance

Frame Rate Frequency	Keyframe Interval (in seconds)									
	1	2	3	4	5	6	7	8	9	10
60Hz (60fps)	60	120	180	240	300	360	420	480	540	600
30Hz (30fps)	30	60	90	120	150	180	210	240	270	300
25Hz (25fps)	25	50	75	100	125	150	175	200	225	250
24Hz (24fps)	24	48	72	96	120	144	168	192	216	240

The smooth switching can be achieved amongst the different video qualities (bitrate) by keeping the same Sequence Parameter Set/Picture Parameter Set (SPS/PPS), Network Abstraction Level (NAL). Furthermore, the following important components should be considered:

- Bitrate as a variable component among all possible switching bitrate.
- Fixed frame size and same video duration across all switching bitrate.
- Avoid scaling down from the larger screen size to lower frame size, and vice versa.

4.4 Client Server Communication

HTTP video streaming service is based on the communication between client and server with the TCP/IP protocols commonly used on the Internet for transmitting web pages from servers to the client. A web page is a collection of objects that are downloaded by using persistent or non-persistent HTTP connection. In [100], authors use HTTP non-persistent connection, where each video segment is downloaded by using a separate connection. In our proposed system implementation, we use HTTP persistent connection. This has high performance especially when video streaming shares available bandwidth with TCP greedy flows [17]. Additionally, in [68] authors proved that HTTP persistent connection has significant performance improvement over non-persistent connection.

Initially, the client connects to the server via a web browser, and after successful connection the flash application (player) is loaded in the browser in order to start the video streaming service. When the client starts the video streaming, a GET HTTP request is sent to the server. This initial request point out the manifest file (F4M) that is stored on the server, and contains the information about video meta data (e.g. video name, encoding video quality rate, etc.). After parsing the manifest file, the client player has complete information about the URL of each video quality level, and it can request the specific video quality level via a HTTP GET command, based on the decision made by its video stream switching controller.

Our proposed client player is based on Adobe streaming technology, where the server stores the different video quality files for each available video. In Adobe technology, a video is logically segmented as compared to physically different segments of each video quality level, which are used by the Apply and DASH based HTTP adaptive streaming technologies. In Adobe adaptive streaming, the server stores each video quality level that are logically segmented (i.e. keyframe) but physically stored in a single file. Microsoft Smooth Streaming (MSS) technology use the same technique. The main advantage of this technique is to reduce the number of objects handled by the CDN.

The videos are encoded using the H.264 codec with Instantaneous Decoding Refresh (IDR) I-frames at 24 frames per second (fps). The stream is broken at Group of Pictures (GOP) boundaries that begin with IDR I-frames, and has length equal to 96, which means the distance between two I-frames (i.e. keyframes interval) is 4 seconds. The video quality level will change only at the IDR keyframe interval that can also has different profiles (e.g. resolution, 2D, etc.) for different devices.

When the client parses the manifest F4M file, it opens a TCP socket to send the HTTP GET request, pointing out specific video quality levels in the URL. The server sends the requested video quality level back to the client using the TCP protocol on the same socket, and this streaming procedure continues even during stream switching process by using same socket.

4.5 Rate Adaptive Algorithm

A rate adaptive algorithm is a method that changes the video quality based on network conditions, end user's device properties, and other characteristics. Generally, Internet video services run over unmanaged networks. Mostly, the video streaming technologies send the video content from the server to the client using the standard delivery HTTP protocol over TCP. HTTP has some advantages that enable universal access, availability of connection to many devices, reliability, mobile-fixed convergence, robustness and last but not the least reuse of existing delivery infrastructure for larger distribution of media services. The main drawback of transport service over the HTTP protocol is the lack of bitrate guarantees. This deficiency of HTTP can be solved by enabling the client to dynamically select the appropriate video quality/bitrate segment of the same video content according to varying network conditions. Based on network conditions, TCP parameters provide vital information to the client, and streaming is managed by a rate adaptive player at the client end.

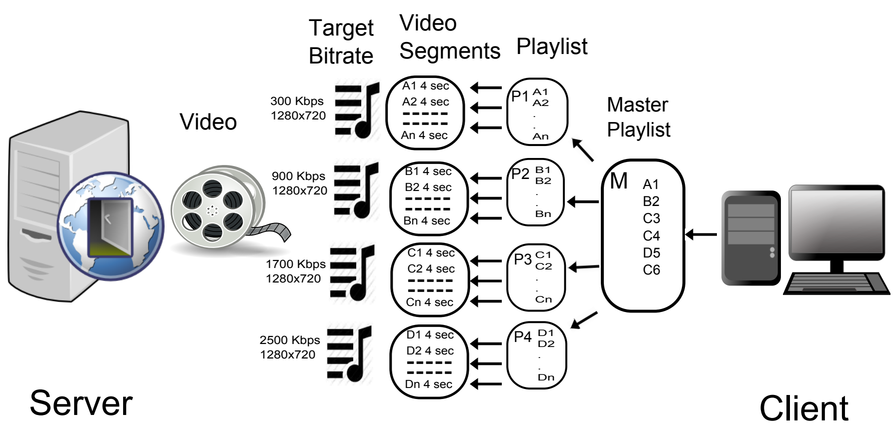


Figure 4.4 – Example: Adaptive Streaming

Figure 4.4 illustrates a simple behaviour of adaptive streaming in dynamic network conditions, and Figure 4.5 shows adaptive visual quality experience by the client. This example shows the rate adaptive streaming where only one video resolution is selected based on display property of a client device, but it is encoded with distinct target bitrates in order to conform with client or network conditions. It is observable that a video with different target bitrates has the same segment duration, and it will help the client to easily switch the next video segment, either lower or higher video quality, based on network condition. Each target video bitrate belongs to one playlist or profile, but the client gets the desired video segment from the different playlist, and makes its own playlist that is known as master playlist/profile. The master playlist contains different video segments based on the client device capabilities, network conditions, and preferences for optimal video quality experience as perceived by the end-user.

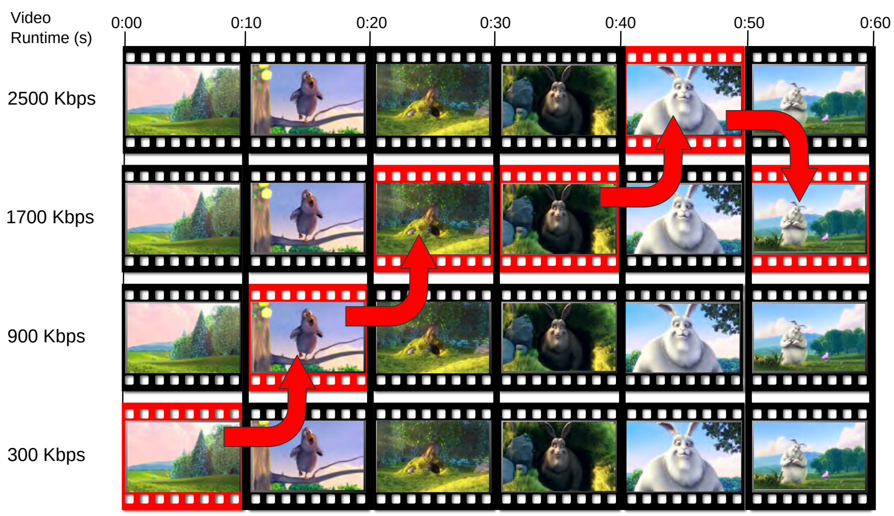


Figure 4.5 – Example: Adaptive Streaming Sequence

TCP parameters have a significant impact on the communication between the client and the server, especially in the transportation of adaptive video streaming. The analysis of TCP-based video streaming shows that TCP throughput should be double as compared to the video bitrate, which guarantees a smooth and good video streaming performance [106]. Adaptive video streaming endeavour to overcome this problem, and it adapts the video bitrate according to the available network bandwidth. The network bandwidth has direct influence on video quality selection, as the buffer is mainly affected by the network bandwidth. The buffer-based smooth adaptation method is discussed in [110], where the client-side buffer time is used as an important feedback parameter for avoiding buffer underflow/overflow.

4.6 System Model

We consider that x different video segments that are stored on the server. Each segment has a specific playback duration, and as a simplicity, we assume that all segments have the same duration. Generally, each segment has a duration between 2 to 10 seconds, and the proposed BBF method uses 4 seconds segment length. Each segment belongs to one video representation, in other words, one video is present in different set of representations (different profiles). The available representations for a given video are denoted by R. The number of available representations in R represent the distinct aspects of a video. They might contain different video qualities encoded at different bitrates, different resolutions, 2D or 3D video format. Normally, the recorded video representations are downloaded earlier by the client in the form of a manifest file, before it starts playing session. An XML-based manifest (F4M) or SMIL [11] file contains the necessary information about the available video profiles.

Let us consider user requests the video from a streaming server. A set of suitable video representations for a specific user is demoted as R_s. In case the user's device has a small screen with limited memory (e.g. smart phone), based on user device properties, a client specific video representation should not include the high resolution video, and similarly, it also does not take into account the high quality video that consumes more memory. It is useless to send unsuitable videos (e.g. high resolution) to devices that do not support it. In order to maximize user's QoE, an appropriate video representation should be selected based on device's properties and network conditions.

In this study work, a client player based on our proposed BBF method dynamically selects the appropriate video representation from R, and the client specific video representation R_s contains a finite set of representation. A video representation r belongs to R_s ($R_s = r_1, r_2, ...r_n$), where r_1 denotes the lowest video quality while r_n denotes the highest video quality representation. We identify the current video stream by $cSID$ that denotes any r_i representation belonging to R_s. Similarly the $nSID$ symbol denotes the next video stream identity that represents the r_{i+1} (possible higher quality) or r_{i-1} (possible lower quality) representation belonging to R_s. The adaptive method keeps monitoring the QoS parameters, because video quality switching is based on parameters related to video and network conditions.

The video playback starts immediately after completing the initial buffering requirement, i.e. there should be enough buffered video frame data in order to playback the video stream. Suppose that video is buffered for *Period*1 as shown in Figure 4.6, and it starts playing. The video has j number of period, and one period represents the playing duration of the same video quality. However, the adaptive player must takes a decision about the video quality of the next period before the end of the current period. In the adaptive video streaming method, it is required that during the video playback period available bandwidth, buffer, and dropped video frames should be monitored continuously in order to adapt the video quality according to time varying parameters for the next period. Let consider the *Period*1 and *Period*2 as shown in Figure 4.6. To make sure that there will not be an interruption for video quality Rs (client specific video) during the next playback time of *Period*2, we must instantaneously monitor the dynamic parameters (bandwidth, buffer, and drop video frame) at the client side. The playing duration of each period can be divided into n number of discrete time instants $(T_1, T_2, \ldots\ldots, T_n)$. It is not necessary that each playback Period has the same duration, e.g. in case of aggressive buffer mode, the Period duration becomes half ($Period_j/2$) of normal Period, as it is essential to monitor the dynamic parameters more frequently to avoid a buffer empty state. The period length has a significant role in estimating the QoS parameter (e.g. bandwidth) [99]. The general expression for calculating the average buffer length B for the specific time period is given in Equation 4.4.

$$B_j = \frac{\sum_{i=1}^{n_j} b_{i,j}}{\left(n_j\right)}, n_j = 1, 2, ...n \tag{4.4}$$

where $b_{i,j}$ is the measurement of instantaneous buffer for $Period\ j$ at time instance i. Let us consider the case for the next playback *Period*2; the instance buffer $b_{1,1}$ calculated at time T_{11} for *Period*1, and similarly next instance buffer $b_{2,1}$ represents the time T_{21}, and so on. In the BBF method, we set the instantaneous time to 150 milliseconds. The general expression to calculate the dropped frame is given in equation 4.5

$$dfps = \frac{(df - pdfps)}{ct - tpdfps} \tag{4.5}$$

where df is the number of video frames dropped in the current video playback session, and

$pdfps$ is a number of video frames dropped in the previous playback session. The current time is denoted by ct, while $tpdfps$ represents the time when $pdfps$ occurred. In recorded video streaming, when a downloaded video has a high-quality or high-resolution then the client might drop frames df because of insufficient system CPU resources to decode the required number of frames per second. In live streaming, the buffer drops video frames if the latency is too high. This property df specifies the number of frames that were dropped and not presented to the user for viewing. Initially, the dropped frame rate can be valid only if there are enough downloaded video data. In our case, the average dropped video frame rate ($adfps$) can be calculated from Equation 4.6 as follows

$$adpfs_j = \frac{\sum_{i=1}^{n_j} dfps_{i,j}}{\left(n_j\right)}, n_j = 1, 2, ...n \tag{4.6}$$

where $dfps$ represents the video dropped frame per second. Similarly, the average bandwidth (BW) is calculated from Equation 4.7 as follows

$$BW_j = \frac{\sum_{i=1}^{n_j} bw_{i,j}}{\left(n_j\right)}, n_j = 1, 2, ...n \tag{4.7}$$

where $bw_{i,j}$ is the measurement of instantaneous bandwidth for $Period\ j$ at time instance i, as explained earlier in case of buffer. The instantaneous bandwidth value bw is calculated by dividing the downloaded fragment size and download duration of that fragment. The weighting vector is used to calculate the bandwidth on the recent sample plus last downloaded sample. The BBF method uses the weighting vector $[7, 3]$ by considering the two fragments, where higher weight is assigned to the recent fragment sample. By exponentially averaging the bandwidth BW_j, the maximum bandwidth can be calculated by using the Equation 4.8

$$BW_{max(j)} = (\theta)BW_{max(j-1)} + (1-\theta)\frac{BW_j + BW_{j-1}}{2} \tag{4.8}$$

The estimated maximum bandwidth (BW_{max}) is used to regulate the client's buffer. The θ parameter is a weighting factor that finds out the last two bandwidth sample weight against the history of estimated bandwidth. We conducted experiments with different θ value, and observed that proposed BBF algorithm performs well when θ value is close to 1. The BBF method uses the $\theta = 0.8$.

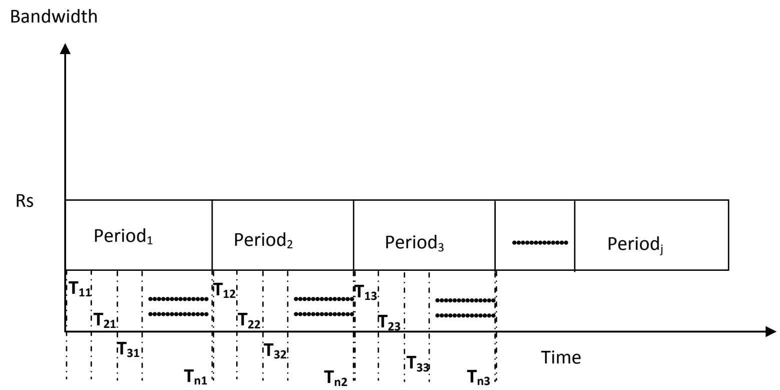


Figure 4.6 – Time Vs Bandwidth

4.7 Proposed BBF Method

The pseudo-code of our proposed BBF rate adaptive algorithm is presented in two sub-algorithms for simplicity and better understanding, but we refer them as a single algorithm. Algorithm 1 deals a case when certain conditions are fulfilled to switch down the current video quality, while Algorithm 2 considers a case when the video is switched up on a higher quality based on maximum bandwidth. The BBF algorithm dynamically selects an appropriate set of video representations R_s based on user device properties (e.g. screen resolution). In order to minimize the initial playback time, the algorithm selects the lowest video quality. It starts playing video as soon as the initial segments are downloaded, and buffer length (in seconds) reaches the start buffer length B_s. In case of quick start, B_s must be set to a low value, but it is necessary to set its value to be high enough, so it will be easy to compute the maximum bandwidth available for the stream. When a stream begins to play then the algorithm considers the preferred buffer length B_p, instead of B_s. The B_p is the length of buffer (in seconds), after a stream begins playing. The value of B_p should be higher than B_s. The value of B_p represents the preferable buffer length, and it does not illustrate the current buffer length B while playing the video streaming.

The maximum bandwidth capacity available for video stream is represented by BW_{max} that is calculated from Equation 4.7. It represents a client bandwidth, not a server bandwidth and its value changes according to network conditions where client is currently exposed. The currently playing video stream is identified by $cSID$ that denotes any r_i ($i.e.i = 1, 2, ...n$) representation belongs to R_s, similarly the symbol $nSID$ denotes the possible next video stream identity that represents the r_{i+1} (possible high quality) or r_{i-1} (possible low quality) representation belongs to R_s.

The BBF algorithm also monitors the video stream in terms of number of frames per second (fps). In such a circumstances when an average video dropped frame per second ($adfps$) is higher (more than 10%) then it becomes necessary to make a decision in order to adopt lower video quality, as it influences the end user perceived video quality. In [114], the authors study the impact of video frame rate and resolution on QoE by using the full-reference measurement method.

Two more buffers are considered in BBF algorithm, i.e. current buffer time B_c and buffer time B_t. Initially, B_c is equal to B_s, but later it contains the same value as B_p, and in the end of video streaming, B_c will be empty. On the other hand, B_t specifies how long to buffer a video data before starting to display the stream. In order to avoid distortion when streaming pre-recorded (not live) video content, the rate adaptive video player uses an input buffer (here is B_c) for pre-recorded content that queues the media data and plays the media properly. The BBF algorithm also takes into account the worst case scenario when the buffer is in underflow condition. In order to avoid buffer underflow condition that causes the video streaming interruption in form of stalling or pausing, an aggressive buffer length B_a is introduced. In a case, when user buffer length B is less than B_a then a video stream switches to the lowest possible bitrate in order to avoid the buffer from emptying, because an empty buffer can cause a pause or stutter in video streaming. However, shifting to lower possible video quality, it is necessary to check the QoS parameters more frequently for maximizing the user QoE.

Table 4.2, contains the information about all symbols or abbreviations used in the BBF algorithm. The proposed BBF algorithm considers three main parameters, i.e. B, BW_{max}, and $adfps$ in order to switch for lower or higher video quality. However, when the conditions for switching down to lower video bitrate do not fulfil (i.e. Algorithm 1) then the algorithm considers the other condition to shift-up the video bitrate (i.e. Algorithm 2). The BBF algorithm adapts the video streaming by taking into account the following conditions.

Switch down to lower video

- When available maximum bandwidth BW_{max} is lower than the current video stream bitrate $cSRB$.
- When client buffer length B is less than current buffer time B_c.
- Dropped frame per second $adfps$ is greater than 10%.
- Aggressive mode, when client buffer length B is less than aggressive buffer length B_a.

Switch-up to high video bitrate

- When available maximum bandwidth BW_{max} is higher than the current video stream bitrate $cSBR$, but only if find a good buffer level (i.e. $B > B_c$).

4.8 Experimental Setup

The experiential setup contains three important elements; a video streaming server, a video enabled client machine, and network emulator. The network emulator tools are used to emulate the real-time networks, and two mostly used tools are DummyNet [24], and built-in linux NetEm [82]. We use the NetEm as a network emulator to evaluate the proposed BBF algorithm. The experimental setup is shown in Figure 4.7, where traffic flows between the client and the server via network emulator. The client sends the video request message via a HTTP GET command to the video server by using the IP networks (LAN) and in response, the requested video is sent to the client. The server stores multiple copies of single video, but in different video quality (bit-rates). The video content "Big Buck Bunny" is stored on the Apache streaming server, and it has duration almost 10 minutes that is suitable for evaluating the BBF method. The server contains the video contents that are encoded at 10 different video bitrates as given in Table 4.3. When $adfps \geq 20\%$, BBF method lock the video quality for 15 second in order to avoid move again to the quality that causes the decrease in video quality.

Algorithm 1: Rate Adaptive Algorithm Switch down

Input: A finite set $R_s = \{r_1, r_2, \ldots, r_n\}$ of client specific video

Output: Select appropriate video ($nSID$) for end user

Result: Video quality switched down

Conditions to switch down video quality
if $B < B_p$ **or** $BW_{max} < cSBR$ **or** $fps > 0$ *and* $adfps$ *>0.10* **then**
    if $B < B_p$ **or** $BW_{max} < cSBR$ **then**
        $i \leftarrow$lenght of R_s
        while $i \geq 0$ **do**
            if $BW_{max} > R_s(i)$ **then**
                $nSID \leftarrow i$
                break
            $i \leftarrow i - 1$
        if $nSID < cSID$ **then**
            if $BW_{max} < cSBR$ **then**
                Switch down due to less bandwidth
            else
                if $B < B_c$ **then**
                    Switch down due to buffer
        if $B > B_c$ **and** $B_c! = B_p$ **then**
            $B_c \leftarrow B_p$
            $B_t \leftarrow B_c$
    else
        Switching down as adfps is greater than 10%
        if $adfps >= 10\%$ *and* $adfps < 14\%$ **then**
            $nSID \leftarrow cSID - 1$
        if $adfps >= 14\%$ *and* $adfps <= 20\%$ **then**
            $nSID \leftarrow cSID - 2$
        if $adfps > 20\%$ **then**
            $nSID \leftarrow 0$
    if $B < B_a$ **then**
        Switch down to lowest quality to avoid interruption

Algorithm 2: Rate Adaptive Algorithm Switch-up

```
Input: A finite set R_s = {r_1, r_2, ..., r_n} of client specific video
Output: Select appropriate video (nSID) for end user
Result: Video quality switched up
Conditions to switch up video quality
if B < B_p or BW_max < cSBR or fps > 0 and adfps >0.10 then
Run Algorithm 1 "Rate Adaptive Algorithm Switch down"
else
    Switch Up on Maximum Bandwidth
    nSID ← 0
    i ←lenght of R_s
    while i ≥ 0 do
        if BW_max > R_s(i) then
            nSID ← i
            break
        i ← i − 1
    if nSID < cSID then
        nSID ← cSID
    else
        if nSID > cSID then
            switch-up only if find good buffer level
            if B < B_c then
                nSID ← cSID
```

Table 4.2 – Algorithm Abbreviation

Words	**Abbreviations**
Next Stream ID	nSID
Current Stream ID	cSID
Average Maximum Bandwidth	BW_{max}
Client Specific Video Representation	R_s
Average Buffer Length	B
Start Buffer Length	B_s
Preferred Buffer Length	B_p
Aggressive Buffer Length	B_a
Current Buffer Time	B_c
Current Stream Bit-rate	cSBR
Buffer Time	B_t
Current Time	c_t
Current Frame Per Second	fps
Dropped Frame	df
Average Dropped Frame Per Second	adfps
Dropped Frame Per Second	dfps
Previous Dropped Frame Per Second	pdfps
Time Previous Dropped Frame	tpdfps

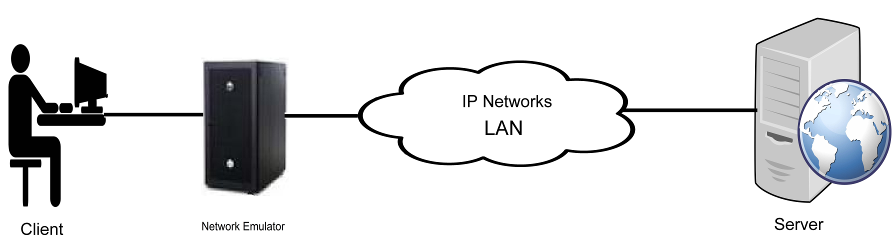


Figure 4.7 – Experimental Setup

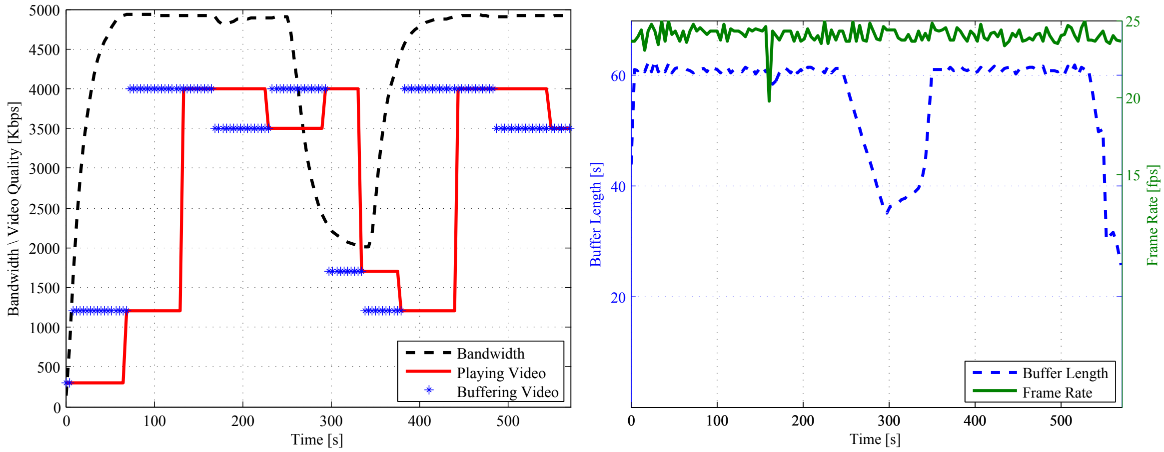
Table 4.3 – Video Content Quality

Videos	**Bitrate (kbps)**
1	300
2	600
3	900
4	1200
5	1700
6	2100
7	2500
8	3000
9	3500
10	4000

4.9 Results

The BBF rate adaptive method is evaluated in a controlled environment in the form of a testbed, where available network bandwidth and user buffer fluctuates. Their impact on end user's perceived quality is observed while watching the video streaming. The evaluation is done by using the wired Local Area Network (LAN), where the network emulator (NetEm)[82] tool is used to control the network bandwidth between the client and the server. Initially, the BBF-based player is evaluated in terms of different buffer length, which illustrates the importance of different buffer length for selecting the suitable video quality in dynamic network conditions. The evaluation condition is the same for all cases and three buffer lengths (60, 30, and 15 seconds) are provided. Later, the proposed method is compared to Adobe's OSMF streaming method.

Figure 4.8 shows the behaviour of the client's player in terms of bandwidth, buffer, and dropped frame rate, when the buffer length is set to 60 seconds. Initially, the BBF player starts buffering and playing the lowest video quality for reducing the start-up delay. In the meantime, it estimates the available bandwidth, and starts buffering next possible video

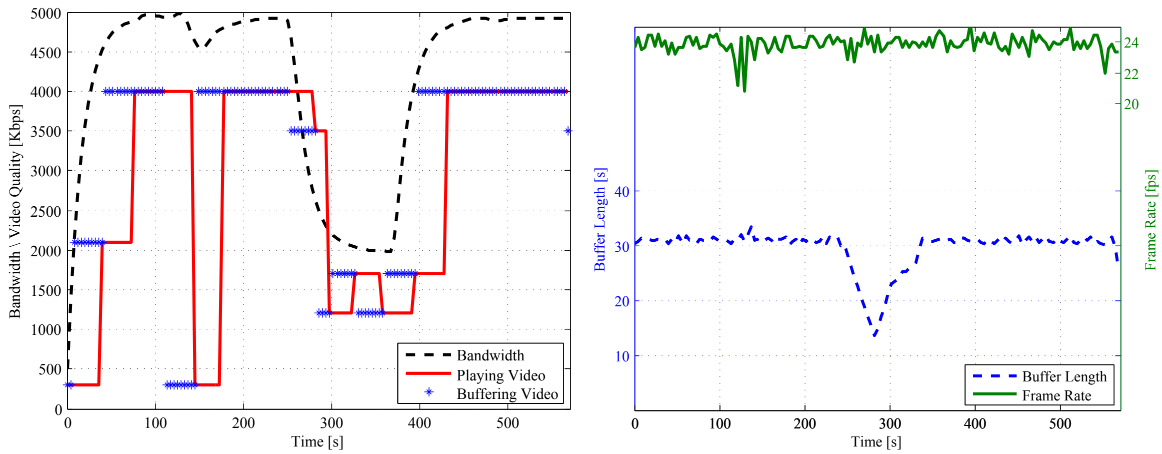


(a) Bandwidth and Video Quality (b) Buffer Length and Frame Rate

Figure 4.8 – Client Video Adaptive when Buffer=60

quality. Figure 4.8b depicts the buffer length, and frame rate behaviour that influences the selection of video quality as shown in Figure 4.8a. When dropped frame rate exceeds 10% (at t=174 sec), and when the buffer length is lower than 60 seconds (at 270, 340, 488, 540 sec.), video stream is shifted down to lower quality. The player performance is evaluated when the bandwidth is reduced to 2000 Kbps (2 Mbps) at 250 seconds, which is half of maximum available video's quality (4000 Kbps). The dropping off bandwidth also drags down the buffer level which causes the video shifting to lower quality (at 270 sec.) in order to avoid jerking or pausing in video streaming. Additionally, the drop of bandwidth also forces the video to switch down to lower video bitrate (at 300 sec.) The bandwidth increases back to 5000 Kbps (higher than maximum video quality) at 350 seconds, and client player successfully shifts-up to the suitable video quality by considering the bandwidth and buffer level.

Figure 4.9 shows the result of the BBF method when buffer length is 30 seconds, and it considers three QoS factors (i.e. bandwidth, buffer, and dropped frame rate) to select the suitable video quality index. Initially, the player starts streaming lowest video quality (300 Kbps), then it switches to 2100, and later to 4000 Kbps in a belligerent way based on bandwidth and buffer. It switches back to lowest video quality '300 Kbps' (at 113 seconds), when dropped frame rate is 21% as shown in Figure 4.9b. In case of sudden drop in



(a) Bandwidth and Video Quality

(b) Buffer Length and Frame Rate

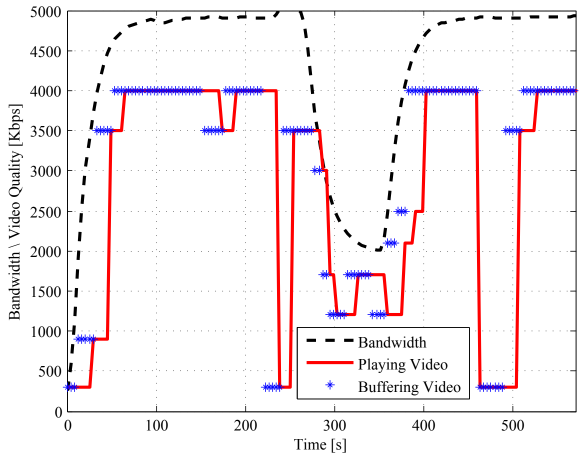
Figure 4.9 – Client Video Adaptive when Buffer=30

network bandwidth, forces the decreasing in buffer level which causes the video switching down to next lower video quality based on bandwidth and buffer length. When the bandwidth reaches 2000 Kbps, video quality shifts are totally based on buffer length. Later, the bandwidth increases back to 5000 Kbps (at 350 seconds), afterwards video quality switch up to highest quality index, i.e. 4000 Kbps.

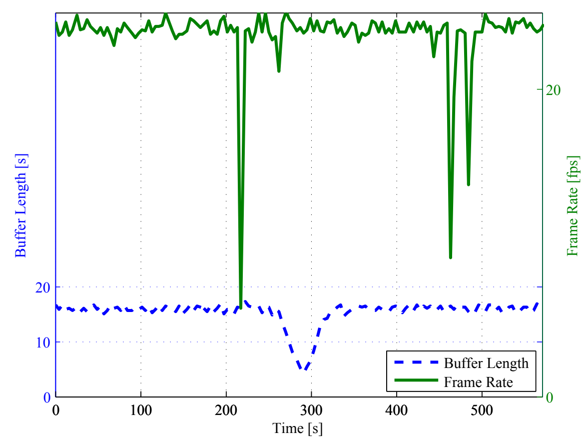
Figure 4.10 represents the performance of BBF rate adaptive method, when buffer length is set to 15 seconds. The two sharp drops in video quality (from 4000 Kbps to 300 Kbps) occurs due to high dropped frame rate (more than 20%) at 220 and 468 seconds. When the lock timer (15 seconds) expires, the video switches back to the highest possible level by considering the bandwidth and buffer level. The impact of sudden drops in bandwidth starts at 265 seconds, which causes the reduction in video quality. When bandwidth reaches 2000 Kbps, then video switch down occurs because of buffer length. The bandwidth increases back to more than 4000 Kbps, which results to a switch up of video quality in an aggressive way by considering the available bandwidth.

It is observed that a larger buffer length is less affected by time varying properties of the network, but it does not efficiently use network resources, and reduces user's QoE.

The performance of BBF method is compared with Adobe OSMF adaptive streaming method. The evaluation is based on the behaviour of adaptive streaming method during the



(a) Bandwidth and Video Quality



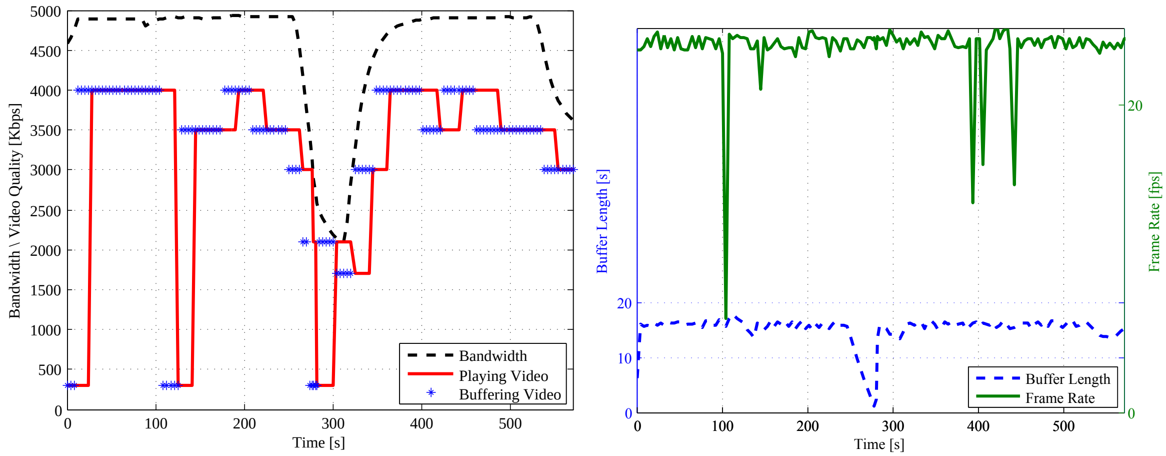
(b) Buffer Length and Frame Rate

Figure 4.10 – Client Video Adaptive when Buffer=15

sudden decrease in bandwidth, and dropped frame rate. The network bandwidth is reduced to half of maximum available video bitrate, when highest video quality is playing, and how the adaptive method efficiently deals with the scenario. Similarly, the influence of buffer level and dropped frame rate are observed on both adaptive method.

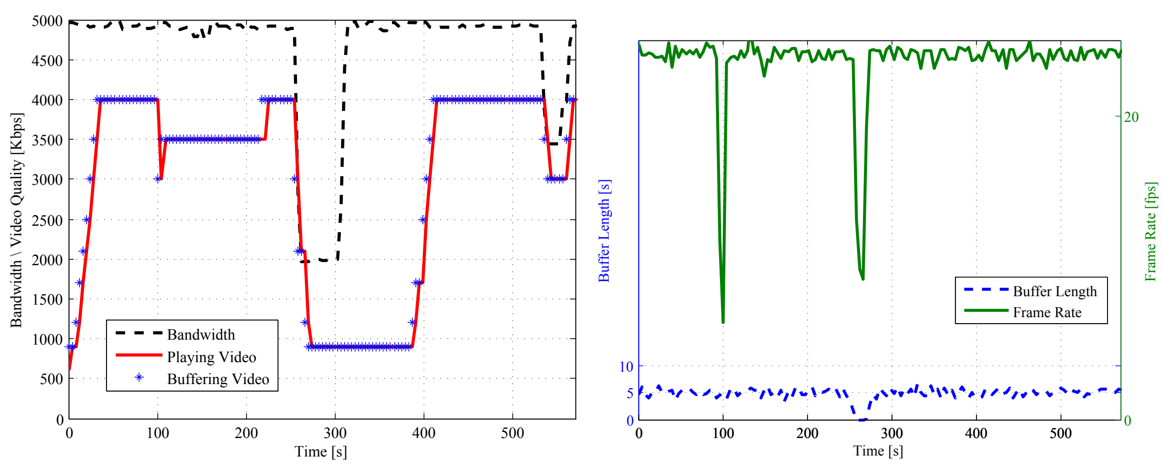
Figure 4.11 shows the performance of BBF method, and Figure 4.12 represents the operation of Adobe's OSMF player in terms of bandwidth, buffer and dropped fame rate. Initially, the BBF method starts playing a lowest video quality (300 Kbps), meanwhile based on current bandwidth and buffer length it starts buffering next possible video stream index as illustrates in Figure 4.11a. When the buffer level is equal to or greater than 15 seconds then BBF method increases the video quality based on available bandwidth. The video quality increases purely based on bandwidth in the aggressive way, compared to step by step manner in OSMF as shown in Figure 4.12a.

When $adfps \geq 10\%$, then BBF method switches down by one video quality level, but switches down two quality level if $14\% \leq adfps < 20\%$. In other cases, it switches down to lower video quality (e.g. 300 Kbps) when $adfps \geq 20\%$. In Figure 4.11a the decrease in video quality to 300 Kbps at 109 seconds occurs due to dropping of frame rate by more than 40%, and BBF method lock the video quality (4000 Kbps) for few seconds (15 sec.) in order to avoid switching again to a quality that would cause the decrease of video quality.



(a) Bandwidth and Video Quality (b) Buffer Length and Frame Rate

Figure 4.11 – BBF Video Adaptive Method



(a) Bandwidth and Video Quality (b) Buffer Length and Frame Rate

Figure 4.12 – OSMF Video Adaptive Method

Later, we observe that the video quality switch-up to 3500 Kbps instead of 4000 Kbps as available bandwidth is higher than highest video quality, but current buffer length not allows to move the highest video quality as shown in Figure 4.11b at 130 sec. In Figure 4.12a, it is observed that the OSMF player switch down two quality level (3000 Kbps), but sudden move up to next level (3500 Kbps) as it has small buffer length (5 seconds), and it locks the video quality index for 2 minutes, which causes the decrease in video quality. The small buffer length can react quickly to changing in network condition, but in case of sudden drop in bandwidth may cause the buffer to flash empty, which leads to pausing, stalling, and jerking in video streaming, which reduces user's QoE.

We reduce the available bandwidth to 2000 Kbps to observe the response of BBF and OSMF player. We observe that BBF method successfully manages to handle the dropping of bandwidth. It switches down the video quality step by step according to bandwidth, and buffer level. The bandwidth forces the buffer level to decrease quickly as shown in Figure 4.11b at 255 sec. BBF method supervises the situation, and based on buffer length (less than 4 sec.) it aggressively shifts the video quality to the lowest level to avoid the pausing, jerking and stalling in video streaming. On the other hand, OSMF player is unable to handle the sudden drops of bandwidth, as its buffer flash empty. That also causes high dropped frame rate, which blocks the video quality to switch up for 2 minutes despite high bandwidth. The user observes the pausing, stalling, and jerking in video streaming, which badly minimize user's QoE. In case of OSMF, we notice that when the video quality locks for a longer period then it does not efficiently utilize the bandwidth as shown in Figure 4.12a, during period 300 to 400 seconds.

In Figure 4.11a, the video switches down by one quality level (at 397 and 446 sec) due to a drop of 10% frame rate, and last decline in video quality occurs due to buffer level at 543 sec. In case of OSMF, Figure 4.12a shows that the video quality changes only because of a bandwidth drop at 530 seconds, as there is no drop of buffer level and frame rate shown in Figure 4.12b.

4.10 Conclusion

This chapter discussed HTTP rate adaptive video streaming services over the TCP protocol. It points out the role of several components in an adaptive video streaming architecture. The video encoding is an essential step that influences the performance of the whole adaptive streaming system. The key elements in video encoding are presented, and we highlight their impact on the adaptive video streaming service. The basic client-server communication in an adaptive video streaming system is described, and client downloads the manifest file to know the available different video representations on the server. The client only requests the appropriate video representation according to available network and device characteristics. The working behaviour of rate adaptive method is presented, and we show that the client makes its own playlist of different video quality based on the network properties and device status.

The BBF method is proposed that considers the three main QoS parameters in order to adapt the video quality. The system model is presented that used by the proposed BBF method. The system model illustrates the working behaviour of the BBF method, and how it computes the different metrics that are used in the decision process of selecting the suitable video quality. The proposed client-side rate adaptive BBF method, adapts the video quality based on dynamic network Bandwidth, user's Buffer status, and dropped Frame rate. The BBF is evaluated with different buffer length, and it is observed that a longer buffer length is less affected with dynamic bandwidth, but it is also not efficiently utilized the network resources. The BBF is evaluated and compared with Adobe's OSMF streaming method. The results show that BBF successfully manages situation as compared to OSMF, in terms of sudden drop of bandwidth, and dropped frame rate when the client system does not have enough resources to decode the frames. Additionally, BBF method optimizes the user's QoE by avoiding the stalling, and pausing during video playback.

The next chapter describes the methods to measure the user's perceived QoE for VoIP multimedia traffic. We propose a new downlink scheduling algorithm for Long Term Evolution-Advanced (LTE-A) network, that allocates the radio resources to end user by measuring the in-speech user's QoE, and other parameters of VoIP traffic.

Chapter 5

QoE Based Power Efficient LTE Downlink Scheduler

The previous chapter discussed about the role of different parameters to regulate the user's QoE for HTTP based adaptive video streaming services. The proposed adaptive BBF method considered the QoS parameters to adapt the video quality. The communication world moves towards an all-IP world, where all services will be IP-based along with essential features and functions. The current Fourth Generation (4G) wireless Long Term Evolution-Advanced (LTE-A) system and future 5G networks will also follow the same all-IP trends. Despite ever increasing video traffic in the IP world, the VoIP is still considered as a main revenue stream in the future wireless communication networks. The powerful mobile devices have capabilities to support VoIP service in the wireless networks. It is difficult to measure subjectively user's QoE for in-service speech quality. The 4G standard of LTE-A wireless system has adopted the Discontinuous Reception (DRX) method to extend and optimize the UE battery life, while there is no standard scheduling method to distribute the radio resources among the UE. This chapter presents a downlink scheduler, i.e. Quality of Experience (QoE) Power Efficient Method (QEPEM) for LTE-A, which efficiently allocates the radio resources and optimizes the use of UE power using the DRX mechanism. The QEPEM uses the E-Model to measures the user's QoE for in-speech VoIP multimedia traffic at the user side. Later, each user feedback, its perceived quality to the Evolving

NodeB (eNodeB), where QEPEM downlink scheduler for LTE-A network decides to allocate the radio resources to the end user based on distinct parameters (e.g. DRX status, channel quality, etc.). This chapter also investigates how the different duration of DRX Light and Deep Sleep cycle influences the QoS and QoE of end users, using VoIP over the LTE-A. The QEPEM is evaluated with the traditional methods, in terms of System Throughput, Fairness Index, Packet Loss Rate, and Packet Delay. Our proposed QEPEM method reduces the packet delay, packet loss, and increases the fairness and UE's power saving with high user's satisfaction. This chapter is based on our contribution from two journal articles. [1] [2].

5.1 Introduction

The tremendous growth in consumer electronic devices with enhanced capabilities, along with the improved capacities of wireless networks have led to a vast growth in multimedia services. The new trends in the electronic market have developed a large variety of smart mobile devices (e.g. iPhone, iPad, Android, ...) which are powerful enough to support a wide range of multimedia traffic. Meanwhile, there is an increasing demand for high-speed data services; 3rd Generation Partnership Project (3GPP) introduced the modern radio access technology, LTE and LTE-Advanced (henceforth refered as LTE). The LTE has the capability to provide larger bandwidth and low latencies on a wireless network in order to fulfill the demand of User Equipments (UEs) with acceptable Quality of Service (QoS); and working on future mobile systems (5G) to provide more freedom in terms of capacity, connectivity, supports the diverse set of services, applications and UEs along with efficient power utilization. In parallel to advanced network technology, a large number of data applications are also developed for smart mobile devices, which motivates users to access the LTE network more frequently [26].

[1] **M.Sajid Mushtaq**, Abdelhamid Mellouk, Brice Augustin, and Scott Fowler. QoE Power-Efficient Multimedia Delivery Method for LTE-A, IEEE System Journal, 2015.

[2] **M.Sajid Mushtaq**, Scott Fowler, Abdelhamid Mellouk, and Brice Augustin. *QoE/QoS-aware LTE downlink scheduler for VoIP with power saving*. In Elsevier International Journal of Networks and Computer Applications (JNCA); DOI: 10.1016/j.jnca.2014.02.01.

Initially, 3GPP improves LTE wireless system by considering the important performance parameters, such as high capacity, lower latencies and offering emerging multimedia service (e.g. VoIP, HD video streaming, multi-player interactive gaming and real-time video). It is necessary to manage these performance parameters in an efficient manner. A key performance parameter on the UE electronics device is power, because emerging multimedia services require computationally complex circuitry that drains the UE battery power quickly, as data transmission bandwidth is limited by the battery capacity [93].

Voice over IP (VoIP) is a popular low cost service for voice calls over IP networks. The success of VoIP is mainly influenced by user satisfaction, in the context of quality of calls as compared to conventional fixed telephone services. Initially, the implementation of VoIP services was unable to handle the unpredictable behaviour of IP networks, which badly affected the growth of early VoIP services, because its traffic streams are both delay and loss sensitive. It is a main challenge for VoIP services to provide the same QoS as a conventional telephone network, i.e. reliable and with a QoS guarantee.

The bearer quality is managed as a single quality plan in conventional networks, while in Next Generation Networks (NGNs), it is also necessary to manage end-users QoE. In a wireless system, the unpredictable air interface behaves differently for each UE. In these circumstances, it is necessary to monitor the QoE in the network on a call-by-call basis [86].

The main challenge in any wireless system is to optimize the power consumption at the UE. The Discontinuous Reception (DRX) method is not a novel approach in LTE [91], because the existing cellular communication systems (e.g. GSM, UMTS) use it to optimize the power consumption at the UE. In Universal Mobile Telecommunications System (UMTS), the DRX method uses two cycles, i.e. Inactivity for UE wakeup and DRX cycle for sleep. The main difference between LTE and early DRX method is that UE can switch to the sleep state even if the traffic buffer is not empty [35]. In LTE, the DRX states (e.g. Inactivity) depend on the scheduling, because it increases the UE's active time by reinitializing the Inactive cycle. The idea is to optimize the UE's battery life, so that it does not run out of power too quickly.

To save the power at UE, the LTE specification uses the DRX method along with Light Sleep and Deep Sleep methods. In DRX Light Sleep method, the UE enters into sleep mode for a shorter period of time. The UE consumes less power in the method than in normal active operational mode, because UE does not switch-off its receiver completely. Meantime, UE's receiver switches between active and sleep mode periodically to receive the scheduled packets. In a case, when the UE does not receive the packet for a long period the UE goes into the DRX Deep Sleep mode, and turned off its receiver completely. The DRX Deep Sleep mode has longer duration than the DRX Light Sleep mode, and does not consume any power. The multimedia traffic directly influences by DRX Sleep mode, because as increased power saving will result in more packet delays or packet loss. Thus it is required to optimize the DRX parameters for maximum power saving without degrading network performance that directly influences the service quality experienced by the user, especially for real-time multimedia services (e.g. VoIP, video streaming). In this context, our proposed scheduling method plays an important role that considers the DRX parameters in its scheduling decision for best network performance and maximum user's QoE. Quality of Experience (QoE) is a new concept that evaluate the quality of service by considering the users' perception.

Many network researchers are now working on this concept, and trying to integrate it in network decisions to ensure a high customer satisfaction with minimum network resources. The proposed QEPEM algorithm takes the scheduling decision by considering the user satisfaction factor. Generally, QoE is considered as a subjective measure of user satisfaction of a given service. According to [85], the standard definition of QoE is: a measure of the overall acceptability of an application or service, as perceived subjectively by the end-user.

We have discussed in chapter 3, there are two methods can be used to evaluate the quality of multimedia services: the subjective and the objective method. The subjective method is proposed by the International Telecommunication Union (ITUT) Rec. P.800 [33] which is mostly used to find out users' perception of the quality of speech. The Mean Opinion Score (MOS) is an example of a subjective measurement method in which users rate the voice quality by giving five different point score from 5 to 1, where 5 is the best and 1 is the worst quality. On the other hand, the objective method uses different models of human expectations and tries to estimate the performance of speech service in an automated manner, without human intervention. It is very difficult to measure subjectively the MOS of in-service speech quality because MOS is a numerical average value of a large number of user's opinion. Therefore, objective speech quality measurement methods are developed to make a good estimation of MOS. The E-model [77] and Perception Evaluation of Speech Quality (PESQ) [27] are objective methods for measuring the MOS scores. PESQ cannot be used to monitor the QoE for real-time calls, because it uses a reference signal and compares it to the real time degraded signal for calculating the MOS score. Therefore, we have used the E-model computational method to calculate the MOS score of conversation quality by using the latency (delay) and packet loss rate with the help of the transmission rating factor (R-factor) [77].

In this chapter, we propose a downlink scheduling method called QEPEM for LTE networks that uses an opportunistic approach to calculate the priorities of UEs based on user perception (QoE), and other important parameters for assigning the radio resources among UEs. The main objective is to enhance the user satisfaction by monitoring the MOS score of each UE. The priorities of UEs are calculated by considering the following parameters: MOS, channel condition, channel condition, average throughput, UE buffer status, UE DRX status, and Guaranteed Bit Rate (GBR) or non-GBR traffic. The performance of the QoE Scheme is compared with two traditional scheduling schemes, which are Proportional Fair (PF), and Best Channel Quality Indicator (BCQI). Two traditional methods are selected because they perform well in some QoS metrics according to the network conditions, as these are discussed in later section. The performance assessment is done for loss and delay sensitive VoIP multimedia traffic, and its impact on QoE is evaluated with the help of LTE System Level simulator.

5.2 An Overview of LTE

The increasing demand of high speed data services such as conversational voice, video and online gaming; the 3GPP introduced the new radio access technology LTE. The radio network architecture proposed by the 3GPP LTE consists of evolved NodeB (eNodeB) which provides a link between UE and core network. The eNodeB is responsible for the major Radio Resource Management (RRM) functions such as packet scheduling. The UE is connected with eNodeB via Uu interface. The eNodeB is connected to core network (MME/S-GW) via S1 interface, and each eNodeB is interconnected via X2 interface as shown in 5.1. The Mobility Management Entity (MME) is an important part of LTE architecture, which is responsible for paging and UE mobility in idle mode within the network. The Serving Gateway (S-GW) node is responsible to route user data packets and handles other user requests, e.g. handover. The MME and S-GW are part of the core network.

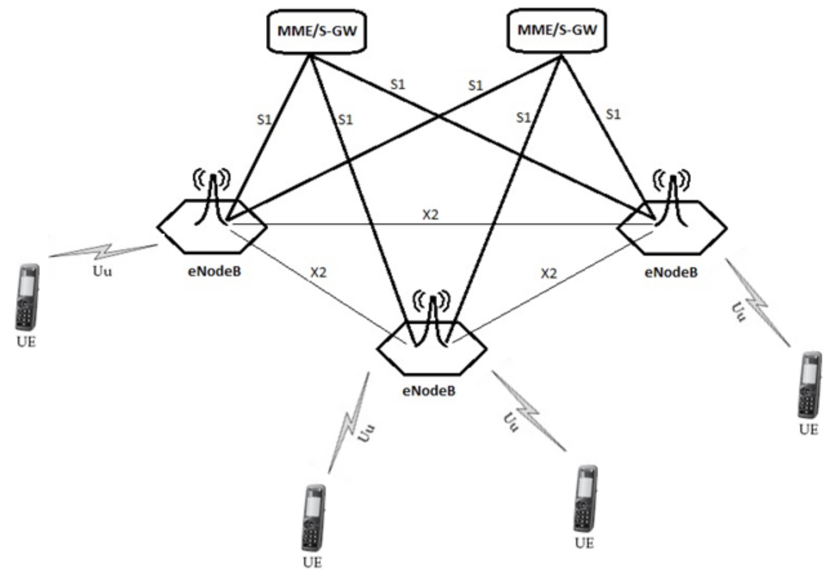


Figure 5.1 – LTE Architecture

LTE uses Orthogonal Frequency Division Multiple Access (OFDMA) as a radio interface which divides the bandwidth into subcarrier and assigns to the users depending on their current demand of service. Each subcarrier carries data at low rates, but at the same time uses multiple subcarriers to provide high data rates [92].

There are some advantages of OFDM as compared to other techniques. Firstly, OFDM uses the multiple carrier transmission techniques which makes the symbol time substantially larger than channel delay spread. Consequently, the effect of Inter Symbol Interference (ISI) reduces significantly. In other words, against the multi-path interference (frequency selective fading) the OFDM provides high robustness with less complexity. Secondly, the use of Fast Fourier Transform (FFT) processing; the OFDM allows low-complexity implementation. Thirdly, OFDM offers the complete freedom to the scheduler by using the frequency access technique (OFDMA). Lastly, it provides the spectrum flexibility which helps for smooth evolution from all the existing radio access technologies toward LTE.

Each downlink frame in LTE consists of 10 ms duration and contains 10 sub-frames. Each sub-frame has a duration of 1 ms, which is known as Transmission Time Interval (TTI), consists of two time slots and each time slot has a duration of 0.5 ms [23].

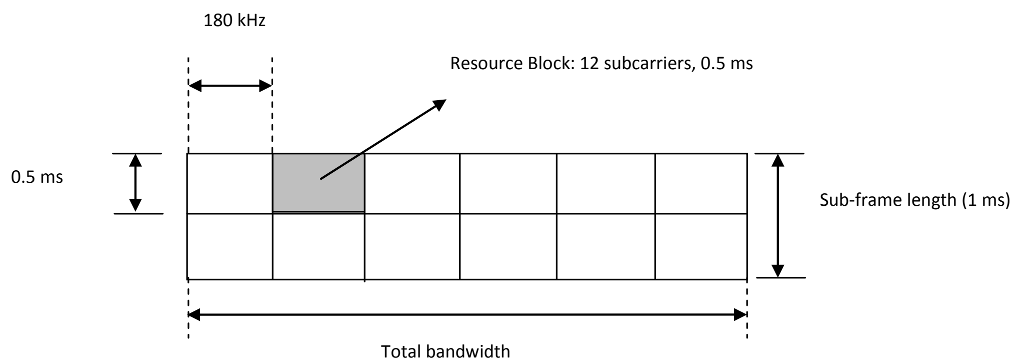


Figure 5.2 – LTE Frame Structure in Frequency Domain

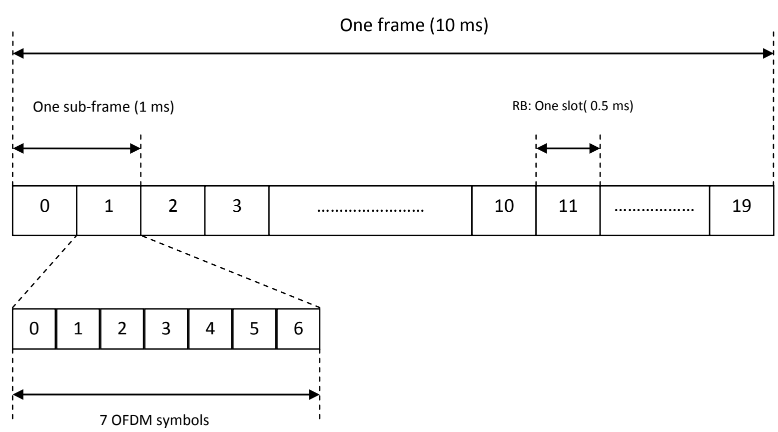


Figure 5.3 – LTE Frame Structure in Time Domain

The radio resources available for users are called Resource Blocks (RBs) which are defined in frequency as well as the time domain. In frequency domain, one RB is a collection of 12 contiguous subcarriers and each RB consisting of 180 kHz bandwidth (12 subcarriers; each subcarrier is 15 kHz) as shown in 5.2, while in the time domain, each RB is defined as 0.5 ms time slot and each time slot carries 7 OFDM symbols as shown in 5.3. Two consecutive time domain RBs make a TTI which is equal to one sub-frame of 1 ms duration. Each UE reports its channel condition to its corresponding eNodeB on every TTI, which includes received Signal to Noise Ratio (SNR) of each subcarrier at the user side. These feedback reports also consist of other radio parameter status perceived by the UE such as CQI, MOS, Rank Indicator, and user buffer status.

5.3 E-Model

The E-model defined in the ITU-T Rec. G. 107 [77], is an analytical model of voice quality and it is used for the network planning purposes. In the E-model, the basic result is to calculate the $R - factor$, that measures the voice quality ranging from 100 to 0, where 100 is the best and 0 is the worst quality. The R-factor value is used to determine the MOS value, which is the arithmetic average of user opinion. The MOS value is obtained from $R - factor$ by using the equation (5.1) [103].

$$MOS = \begin{cases} 1 & R < 1 \\ 1 + 0.035R + R(R-60)(100-R)7.10^{-6} & 0 < R < 100 \\ 4.5 & R > 100 \end{cases} \tag{5.1}$$

The general correlation between $R - factor$, MOS scores and the quality of user experience with VoIP service is shown in Table 5.1. The high value of $R - factor$ gives the highest MOS score, and the user gets the best QoS with high satisfactory experience.

The $R - factor$ mainly depends on four parameters as shown in equation (5.2)

$$R = R_o - I_s - I_d - I_{ef} + A \tag{5.2}$$

where R_o represents the basic signal-to-noise ratio, which includes noise sources such as circuit and room noise, I_s is a combination of all impairments with voice signal, I_d is the impairment's factor caused by delay, I_{ef} is an effective equipment impairment factor associated with the losses as it is defined in [29], and A is the advantage factor. In [47], ITU-T provides the common values of impairment factors. After selecting the default values, we can obtain the reduced expression for the $R - factor$ in equation (5.3).

Table 5.1 – Correlation between R-Factor, MOS and User's Experience

R-Factor (lower limit)	MOS (lower limit)	User Experience
90	4.34	Excellent
80	4.03	Good
70	3.60	Fair
60	3.10	Poor
50	2.58	Bad

$$R = 94.2 - I_d - I_{ef} \tag{5.3}$$

Equation (5.3) clearly shows that $R - factor$ mainly depends on the end-to-end delay and total loss probability, which affect the VoIP call quality. The delay components (I_d) is provided in [77] and its influence on voice quality depends on a critical time value of 177.3 ms, which is the total delay budget for VoIP streams. The impact of this delay is modelled in [16], and it is given in equation (5.4)

$$I_d = 0.024d + 0.11(d - 177.3)H(d - 177.3) \tag{5.4}$$

where d is the one way delay (in milliseconds) and $H(x)$ is a step function as mentioned in equation (5.5)

$$H(x) = \begin{cases} 0 & \text{if x} < 0 \\ 1 & \text{if x} \geq 0 \end{cases} \tag{5.5}$$

The quality of a VoIP call also depends on loss impairment (I_{ef}), as it is clearly shown in equation (5.3). In order to find the expression for calculating the value of I_{ef}, we use the methods as proposed in [16], [22] and [94] that consider the overall packet loss rate as

$$I_{ef} = \gamma_1 + \gamma_2 ln(1 + \gamma_3 e) \tag{5.6}$$

where e is the total loss probability (including network and buffer) which has a value between 0 and 1, γ_1 represent the voice quality impairment factor caused by the encoder, while γ_2 and γ_3 represent the impact of loss on voice quality for a given codec. In case of a G.729-A codec, $\gamma_1 = 11$, $\gamma_2 = 40$ and $\gamma_3 = 10$, while for a G.711 codec, $\gamma_1 = 0$, $\gamma_2 = 30$ and $\gamma_3 = 15$ as presented in [16]. The final expression of $R - factor$ by using the G.729-A codec is given in equation (5.7)

$$R = 94.2 - 0.024d - 0.11(d - 177.3)H(d - 177.3) - 11 - 40ln(1 + 10e) \tag{5.7}$$

5.4 DRX Mechanism

The Discontinuous Reception (DRX) mechanism has already implemented on 2G (GSM) and 3G (UMTS) cellular networks. LTE specification has adopted DRX at the link level to save power and extend battery life of the UE. In LTE networks, the DRX mechanism can observe the Radio Resource Control (RRC) states between the UEs and eNodeB [93]. The RRC has two different states where DRX mechanism can be worked, i.e. *RRC_Idle* and *RRC_Connected*.

In *RRC_Idle* state, the UE is registered in the LTE network with specific unique identifier, but it does not has an active session with the eNodeB. In this state, the eNodeB can page the UE at any time for the different purpose (e.g. get location information), while UE can request an uplink channel by establishing a *RRC_Connected* state, so that it can receive and transmit data. In the *RRC_Connected* state, the DRX mode can enable during idle periods between the packet arrivals. In case there is no data packet the UE can go into DRX mode.

The LTE's DRX mechanism, the sleep/wakeup scheduling of each UE receiver could be described in terms of three periods (ON-Duration, Inactivity and Sleep Interval) as shown in Figure 5.4. The values of LTE's DRX parameter are defined in [93]. In this chapter; we are considering the following parameters:

- DRX cycle: It is a time interval between the start of two consecutive ON-Duration in which UE remains active. One DRX cycle consists of an ON-Duration and a Sleep

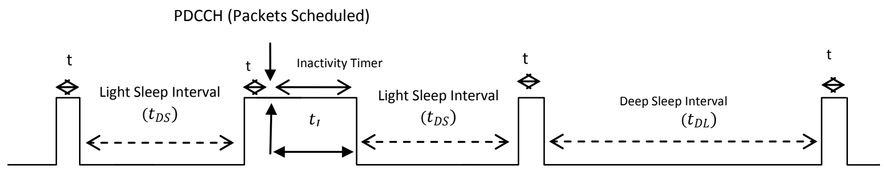


Figure 5.4 – LTE DRX Mechanism at UE

interval.

- ON-Duration (t): It is the time when the UE in the active state, and listens to the Physical Downlink Control Channel (PDCCH). If any data packet is scheduled, the UE starts its Inactivity Timer(t_I) otherwise, it continues its DRX Sleep cycle. In this work, we set the value of this timer to 1 ms.

- Inactivity Timer (t_I): During ON-Duration if a data packet is found through PDCCH, the UE starts its t_I and receives data packets. During t_I, if another PDCCH packet arrives, the Inactivity time restarts itself timer. When t_I expires, DRX cycle starts with a sleep interval. The value of t_I is set to 5 ms.

- Sleep Interval: It is a time interval during which the UE either in DRX Light Sleep t_{DS} mode (consume low power) or DRX Deep Sleep t_{DL} (consume no power) mode. In Deep Sleep mode, the duration of a sleep interval is longer than Light Sleep mode. We consider the following values of Light Sleep duration are 2, 5, 10, 16, 20 ms and for Deep Sleep duration 10, 20, 42, 64 and 80 ms according to [93].

In [48], a semi-Markovian model is presented to determine the numerical values of power saved by the UE in DRX mechanism as shown in Figure 5.5, which is also used by [112], [4] and [34]. This model shows that when the UE in the active state and downloading the data then it consumes 0.5 Watt/TTI. However, if the UE in Light Sleep mode, then it consumes 0.011 Watt/TTI, that means it saves 0.489Watt/TTI, but in the case of the Deep Sleep mode, the UE does not utilize any power (i.e. 0 Watt/TTI) that represents the full-power saving mode.

The impact of the Light Sleep Cycle and Deep Sleep Cycle on power saving can be observed with the help of Figure 5.6. The power saving behavior shown in Figure 5.6 is

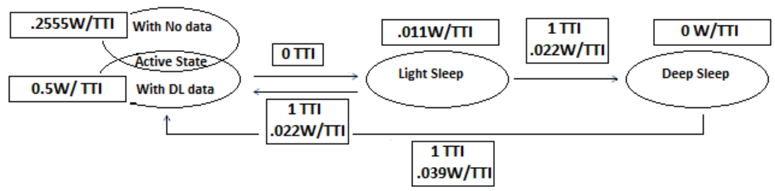


Figure 5.5 – Semi-Markovian Model for Power Consumption

increasing for both DRX Light Sleep Cycle and the Deep Sleep Cycle, that is due to the Sleep Cycles have longer duration and we have fixed the ON-Duration. The longer the DRX Cycles translate into more effective sleep time per cycle, resulting in better power saving.

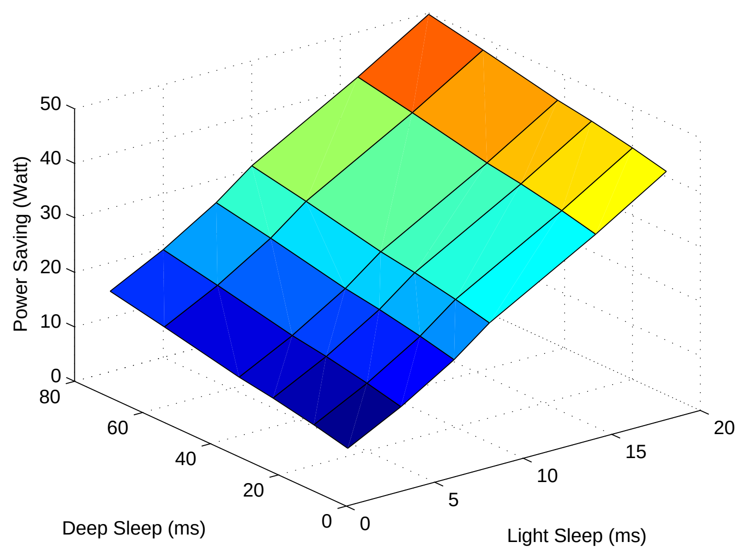


Figure 5.6 – Power Saving in Light and Deep Sleep Cycle

5.5 Methodology and Implementation

Mostly, the algorithms and procedures specified for any wireless network are implemented and tested in the simulation environment, and their performances are evaluated at the link level and system level. The link-level simulation environment considers only the link-related issues such as MIMO gain, channel coding and decoding modeling, physical layer modeling required for system-level testing, etc.... However, the system-level simulation's environment examines the problems that are related to system-level such as mobility handling, interference management and scheduling. The proposed work is implemented and tested in the LTE System level simulator which is developed in MATLAB [46]. This simulator investigates the network performance by considering the physical layer results obtained from the link level. The simulator is implemented with object-oriented programming, which provides greater flexibility to modify, test and implement new functionalities in the current simulator.

The main advantage in separating the link-level and system-level simulator is to reduce the complexity involved in each level. The link-level simulation is good in terms of developing the receiver scenarios, feedback techniques and coding methods, etc.... However, it is impractical for link-level simulations to consider the issues related to cell planning, scheduling and interference, which are part of system-level simulation. Similarly, it is impossible for system-level simulation to take care the whole radio links between the UEs and eNodeB, as it demands a large amount of computational power. The physical layer is implemented as a simple model in the system-level simulator, that acquires its significant properties with high accuracy but low complexity.

The LTE system-level simulator consists of two models, i.e. link measurement model and link performance model. The link measurement model measures the link quality information which are stored in trace files and later used for link adaptation and resource allocation method. The Signal to Interference and Noise Ratio (SINR) is a key parameter of the wireless communication system to measure its link quality. However, the link performance model uses the link adaptation strategy to find out the Block Error Ratio (BLER) with reduced complexity. The BLER is computed at the UE on the basis of resource allocation and Modulation and Coding Scheme (MCS). There are 15 different MCSs defined for LTE, which provide 15 Channel Quality Indicator (CQI) values as presented in Table 5.2. These CQI values use different coding rates between 1/13 and 1 according to different modulation schemes. The link performance model output are stored in trace files, that contain throughput and error rates, which are easily used to calculate their distributions.

Table 5.2 – 4-bit CQI Index and MCS [67]

CQI index	**Modulation**	**Effective Coding rate=** $\frac{c_r}{e_r}$ x 1024	**Spectral Efficiency=** $\frac{R_b}{B}$
0	out of range		
1	QPSK	78	0.1523
2	QPSK	120	0.2344
3	QPSK	193	0.3770
4	QPSK	308	0.6016
5	QPSK	449	0.8770
6	QPSK	602	1.1758
7	16QAM	378	1.4766
8	16QAM	490	1.9141
9	16QAM	616	2.4063
10	64QAM	466	2.7305
11	64QAM	567	3.3223
12	64QAM	666	3.9023
13	64QAM	772	4.5234
14	64QAM	873	5.1152
15	64QAM	948	5.5547

5.5.1 Traditional Algorithms

Generally, the main goals of packet scheduling algorithms in a wireless system are aimed to maximize the throughput and fairness among the users. Two traditional methods can be used in LTE network, because they perform well in some QoS metrics according to the network conditions.

1. The Best CQI (BCQI) algorithm chooses the users which report the highest downlink SNR values to corresponding eNodeB thus utilizes the radio resources efficiently among the users with good channel condition. On the other hand, the users experiencing bad channel conditions would never get resources. As a result, overall system throughput increases but it outcomes in starvation of resources for some users, especially the user far away from eNodeB. Thus BCQI algorithm performs well in terms of throughput but poor in terms of fairness among the users [92].

2. The Proportional Fair (PF) algorithm was proposed to achieve the high throughput and fair resources distribution among the UEs. It was originally developed to support

non-real-time traffic in Code Division Multiple Access High Data Rate (CDMA-HDR) systems. The scheduling strategies which are based on PF algorithm focus on trade-off between maximum average throughput and fairness.

5.5.2 Proposed QEPEM Method

The user's QoE is significantly influenced by the QoS parameters. However, there is always a trade-off between the QoS and power saving, because power saving mechanism badly affected the QoS such as delay. It is essential to have a method that considers the significant factors which influence the user's QoE for in-speech VoIP traffic. In this perspective, a new downlink scheduling method is proposed that efficiently utilizes the power, and keep balance between QoS and power consumption, while also consider their impact on the user's QoE. The proposed *QoE Power Efficient Method (QEPEM)* uses an opportunistic scheduling approach that calculates the priorities of UEs and assigns resources to them. Some scheduling schemes achieve multiuser diversity by using an opportunistic approach for assigning the resources to UEs by considering channel conditions. The high system throughputs can be achieved by assigning resources only to those UEs, who have a good channel condition; however, these techniques fail to fulfil fairness and the UEs QoS requirements. To deal with these problems, other parameters are required in order to balance between spectral efficiency and UE requirements. The QEPEM uses opportunistic scheduling approach, which is based on the six important scheduling dependencies that have the greater impact on QoS and Power saving mechanism, which are: MOS, Channel condition (CQI), Average throughput history, UE buffer status, GBR/non-GBR traffic, DRX status. The priority values for each Resource Block (RB) is estimated for every UE; the scheduler assigns RB to a UE whose priority value is the highest among all other UEs for that specific RB. The short description of each scheduling dependency is given below:

1. UE MOS: Each UE calculates its MOS score based on $R - factor$, that takes into account different factors like QoS parameters that include all kinds of delay (network, buffer, and codec), packet loss (network and UE's playout), and other UE impairment's factor. The scheduler gives high priority to those UEs whose QoE is

decreasing due to a large delay (approaching a predefined threshold) of data residing in the eNodeB buffer; a more waiting time in the buffer means a higher priority, which prevents packet loss and enhances QoE.

2. Channel condition: Scheduler estimates data rates and modulation scheme for each UE on every sub-band. Estimation is based on CQI reports sent by the UEs in the uplink, which include information about downlink SINR.

3. Average throughput: The averaged data rate experienced by each UE for a time window. By keeping track of the UE throughput history the scheduler will be able to give more resources to those UE which were lacking in the past to fulfill their requirements and as a result fairness among the UEs would also increase.

4. GBR/non-GBR: Schedulers require treating RT and NRT services separately. GBR is an important parameter for RT serviced UEs. If an UE experiences data rate lower than defined by the GBR, the scheduler must allocate more resources to that UE.

5. UE buffer status: Every UE has a finite buffer length (equal to 100 packets) for storing the received packets. Packet losses can occur due to the insufficient space in a buffer. In the proposed algorithm, buffer length at the UE is assumed to be limited and the scheduler gives high priority to the UEs who have more buffer space to avoid packet loss. Similarly, the UEs who have the fewer spare buffer would get low priority to minimize packet loss.

6. DRX status: DRX is an effective power saving technique to prolong UE battery life. There is a tradeoff between power conservation and QoS; more power savings result in higher transmission delays and packet losses. To address this issue, the proposed QEPEM algorithm considers DRX status to retain the delays within thresholds according to QCI characteristics of LTE.

5.5.3 Scheduler Architecture

The main entities involved in the downlink scheduling algorithm are shown in Figure 5.7, where eNodeB is shown on the left side with Layer 1 - Layer 3 and UEs shown on the right side. The information flows shown in the Figure 5.7 with solid lines are used both by the traditional and proposed scheduling algorithms. While information flows shown by dash lines are used only by the proposed QEPEM scheduling algorithm.

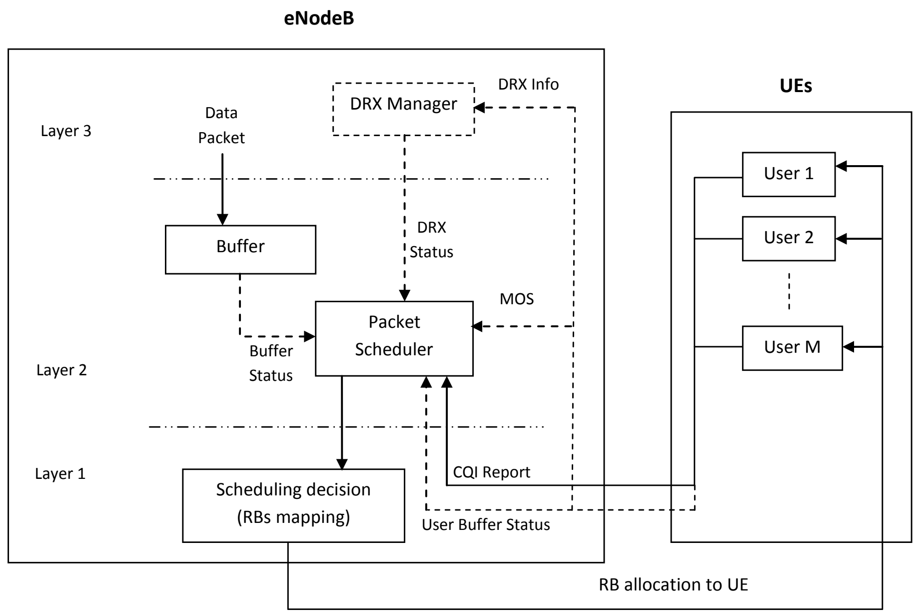


Figure 5.7 – Entities involved in downlink packet scheduler.

The proposed scheduler at Layer 2 acquires CQI reports from the UEs in order to estimate the channel conditions, while UEs' buffer statuses are also received to avoid packet loss because the receiver buffer at the UE is assumed to be limited. A set of buffers at the eNodeB stores the packets for each UE to be scheduled. The proposed scheduler attempts to minimize packet losses by prioritizing the UEs, who has the oldest packet in the eNodeB buffer. Each UE sends its MOS information to the packet scheduler which represents the user's perceived quality, and DRX information to the DRX manager which determines the remaining active and Sleep mode time for each UE. The DRX manager sends the DRX status to the packet scheduler. By considering six scheduling dependencies, the QEPEM scheduler assigns resources to the UEs through PDCCH. This allows the QEPEM scheduling algorithm to keep packets within delay bounds and effectively minimize packet delays, packet loss rate, and maximize the user's QoE.

5.5.4 Scheduling Algorithm

This section described our QEPEM method, that selects and assigns available radio RBs to UEs according to the priority matrix. The priority matrix is calculated by considering the six scheduling dependencies for each UE. The MOS score is calculated from $R - factor$ which considers all types of delay (network, buffer, and codec) and packet loss (network and UE's playout) factors, as a result the MOS score represents the overall effect of delay and packet loss. The priority values for each RB are estimated for every UE; the scheduler assigns RB to a UE whose priority value is the highest among all other UEs for that specific RB.

To calculate the priorities, the algorithm first estimates maximum achievable throughputs for every RB if assigned to UEs according to channel conditions reported by UEs. In order to balance between system throughput and fair resource distribution the proposed scheduler (henceforth is referred to as QEPEM) utilizes the property of Proportional Fair (PF) which is defined in [92].

$$fair\_factor_i = \frac{achievable\_throughput_{ij}}{average\_throughput_i} \tag{5.8}$$

$$R_i(t) = \left(1 - \frac{1}{t_c}\right) * R_i\,(t-1) + \frac{1}{t_c} * r_i\,(t-1) \tag{5.9}$$

Equation (5.8), $achievable\_throughput_{ij}$ represents a theoretical achievable throughput of RB_j if assigned to UE_i at Transmission Time Interval (TTI). In Equation (5.9), R_i represents the $average\_throughput_i$ of UE_i over a window t_c at every TTI and r_i is an achievable throughput of UE_i. The window size t_c is an important element which is used to calculate the average data rate experienced by each UE.

The priority function P_{ij} calculates priorities of $Non{-}RealTime(NRT)$ and $RealTime(RT)$ services from Equations (5.10) and (5.11) respectively. In this study, RT VoIP is used to evaluate the proposed QEPEM method; however, calculating the user's perception (MOS) for different NRT traffic can be considered in future work.

$$P_{ij} = MOS_i * \delta_i\,(fair\_factor_i)\,,\ \ i\ \ is\ \ NRT\ \ UE \tag{5.10}$$

$$P_{ij} \;\;=\;\; MOS_i * \delta_i\left(fair\_factor_i\left(\frac{GBR}{average\_throughput_i}\right)^{\varnothing}\right),\ \ i\ \ is\ \ RT\ \ UE \tag{5.11}$$

where $\varnothing$ is a tunable exponential factor for GBR and δ is a DRX status indicator for each UE. The P_{ij} is a priority matrix for each RB_j if assigned to UE_i while $fair\_factor_i$ in accordance to equations (5.8). GBR is the guaranteed bit rate requirement for GBR UEs. The tunable exponential factor $\varnothing$ can be used to adjust preferences of GBR UEs; if a UE would achieve lower than the average throughput required by GBR, the scheduler will increase the priority of that UE to fulfil the GBR requirement and vice versa. The MOS_i is a priority multiplier that increases the priority of UEs whose facing the degradation of service due to delay and packet loss rate, as higher priority to prevent packet loss. The GBR is irrelevant for NRT traffic because NRT traffic does not delay sensitive, and they do not require minimum data rates to guarantee.

The QEPEM is designed in conjunction with DRX mechanism as to fully exploit high bandwidth efficiency of LTE. The DRX manager at eNodeB shares DRX status with the UEs. On each TTI, the scheduler must consider only the UEs that are in active mode of operation then allocate resources for data transmission; this is achieved by including the DRX status in priority criteria. The DRX status δ defines the state of UE, when a UE is in-active mode $\delta = 1$. When a UE is in Sleep mode $\delta = 0$ makes that UE out of the scheduling competition. Thus the scheduler helps reducing resource wastage by considering only the UEs that are in active state.

5.6 Simulation setup

The simulation setup consists of LTE network that is operating at 2 GHz operating frequency, and 5 MHz system channel bandwidth. The eNodeB is considered to be static, which is serving 15 VoIP traffic UEs who are uniformly distributed within the sector and allowed to move randomly. These UEs can be considered as pedestrians moving with a speed of 5 km/h. The VoIP traffic model is used to simulate the IP based voice according to [32]. The VoIP traffic model is considered due to the major usage on the UEs. Additionally, fading models [15] and [30] are used to simulate realistic channel conditions. DRX Light and Deep Sleep mechanism are implemented on the UEs for saving power, on the other hand, each UE has a finite buffer length at eNodeB that buffered data when the UE in sleeping mode.

A longer Deep Sleep duration can cause the buffer overflow of UE at the eNodeB, because a number of packets being created would be much higher than packets being scheduled. In this work, DRX ON-Duration and In-Active parameters are set to 1 TTI and 5 TTIs, respectively to avoid the UE buffer overflow at eNodeB. The power saving effect on user's QoE is considered in the terms of QoS parameters that will be presented and discussed, which are Average System Throughput, Average Throughput Fairness Index, Packet Loss Rate (PLR) and Average Packet Delay. The three performance evaluation parameters are well known, however, the Fairness Index can be defined in terms of system resource allocation or throughput. Jain's equation is used to obtain a throughput fairness index. In [54], fairness index J for n UEs is defined as

$$J(x_1, x_2, \ldots, x_n) = \frac{(\sum_{i=1}^{n} x_i)^2}{n \sum_{i=1}^{n} x_i^2} \tag{5.12}$$

where x_i is the throughput for the i^{th} UE. The best case can give a maximum value of 1, which means all UEs achieved exactly the same throughput. When the difference between the UEs throughput increase then the value of Jain's equation decreases. The important simulation parameters are listed in Table 5.3 and the durations of Light and Deep Sleep mode cycle are selected according to 3GPP TS 36.331 version 8.8.0 Release 8.

5.7 Simulation Results

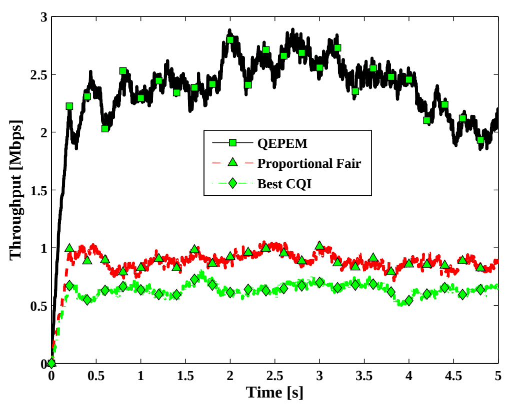
The performance of the proposed QEPEM method will be evaluated and compared with two traditional scheduling algorithms; Proportional Fair (PF), and Best CQI (BCQI) in power saving mode. The evaluation and comparison are done with the same simulation environment and parameters.

Table 5.3 – Main Simulation Parameters

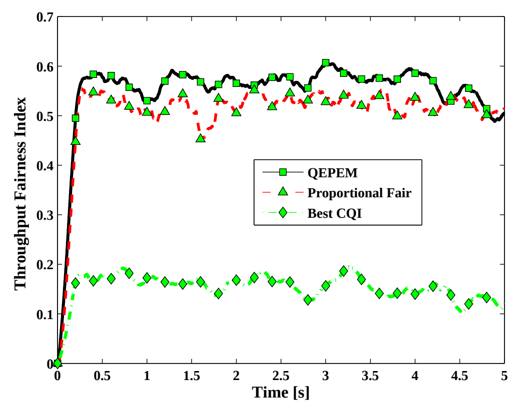
Parameters	Values
eNodeB radius	250 m
Number of sectors per eNodeB	3
Target area	Single sector
Number of UEs	15
eNodeB total TX power	20 W
Number of antennas (SISO)	1 TX, 1 RX
Fading models	Fast fading
UE Speed	5 km/h
Operating frequency band	2 GHz
System channel bandwidth	5 MHz
Number of RBs	25
$\varnothing$	2
GBR	25 kbps
CQI reporting	Every TTI
Traffic model	VoIP
VoIP packet generation interval	20 ms
VoIP delay threshold	100 ms
Power saving mechanism	DRX Light and Deep Sleep
DRX on duration	1 TTI
DRX In-Active duration	5 TTIs
DRX Light Sleep duration	2, 5, 10, 16, 20 (ms)
DRX Deep Sleep duration	10, 20, 40, 64, 80 (ms)

5.7.1 Performance Analysis with Fixed Deep Sleep 20 ms

The simulation setups are same for all the schedulers as given in Table 5.3, and performance are evaluated in the varying power saving environment DRX Light Sleep with fixed Deep Sleep mode of 20TTI (20ms). The DRX mechanism is applied on the UEs along with the fixed DRX ON-Duration of 1 TTI, while the In-Active duration set to 5TTI. The simulation executes for different Light Sleep parameters, and one result is given in Figure 5.8, while impacts of other parameters are summarized in Table 5.4.



(a) Average Throughput

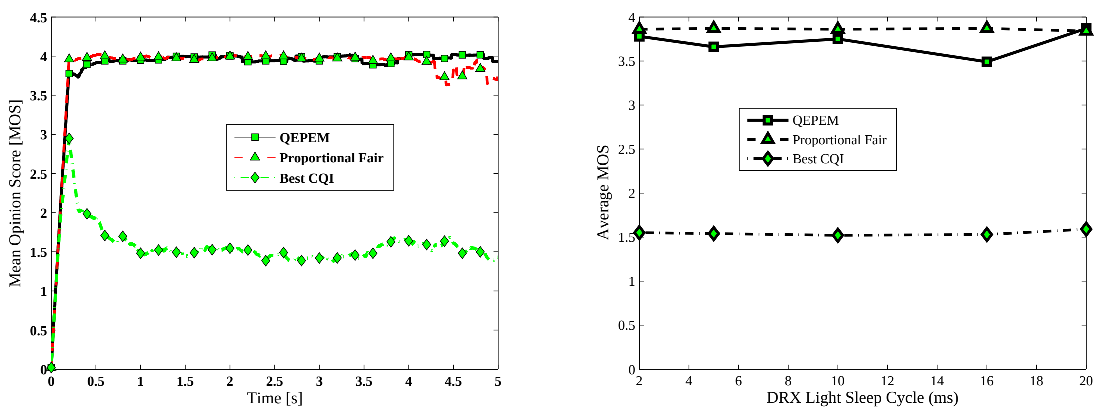


(b) Throughput Fairness Index

Figure 5.8 – Light Sleep = 20 ms, Fixed Deep Sleep = 20 ms

Figure 5.8a shows average system throughput when the simulation runs for 5000 TTI, which are equal to 5 seconds. The results are obtained, when the duration of DRX Light Sleep Cycle is set to 20ms (20 TTI) with a fixed duration of the DRX Deep Sleep Cycle, which is equal to 20 ms (20 TTI). The result shows that the throughput of the proposed QEPEM method is significantly higher as compared to all other schedulers. QEPEM uses the DRX information of each UE, in other words; QEPEM method considers the ON-Duration and In-active duration of all UEs during the scheduling decision. The traditional schedulers are designed to consider all the UEs that are connected at the time scheduling is performed. PF holds second position in terms of throughput because it also tries to balance the throughput with the resource fairness. BCQI performed the worst in this regard, because BCQI chooses only those UEs, which have the best channel conditions in the uplink through the CQI feedbacks.

Figure 5.8b, illustrates the Throughput Fairness Index according to Jain's equation. The result clearly shows that proposed QEPEM method performed the best as compared to all other scheduling schemes. QEPEM manages to achieve higher fairness, because it considers the channel conditions and UE's GBR requirements. It tries to allocate resources to those UEs which packets are residing in the eNodeB buffer for a longer time to avoid the packet lost, and improve the user's QoE. Similarly if the UEs is lacking in throughput according to their defined GBR requirement, then it again allocates more radio resources to those UEs. PF does not consider the sleeping state of UEs, but it tries to achieve fairness among them by considering the performance history of each UE. It follows the pattern of QEPEM method. The value of BCQI is close to the worst-case scenario as it allocates the resources only to those UEs which report good channel condition.



(a) Mean Opinion Score

(b) Average MOS Value

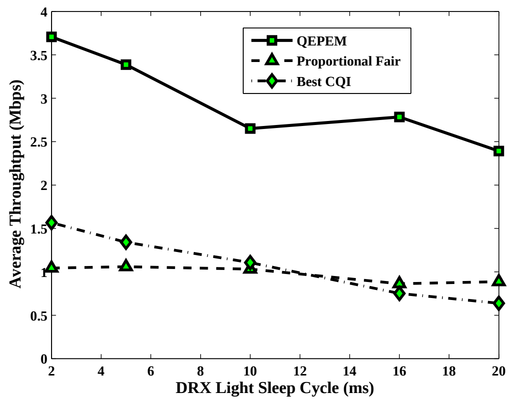
Figure 5.9 – Light Sleep = 20 ms, Fixed Deep Sleep = 20 ms

Figure 5.9 illustrates the performance of three schedulers in terms of user's perceived QoE, when DRX Light Sleep cycle has a duration of 20ms along with fixed Deep Sleep duration of 20ms. Figure 5.9a shows that the QEPEM and PF have almost the same performance; however, BCQI has worst performance. Similarly, Figure 5.9b shows that performance of PF is close to proposed QEPEM method, unless the Light Sleep has a duration of 16 ms. BCQI has bad performance, as it deals only the limited UEs that are reporting the same channel quality.

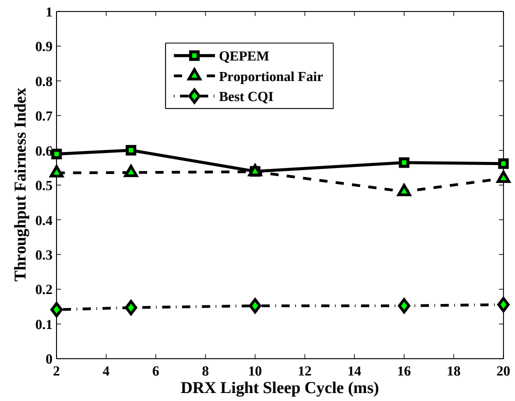
Table 5.4 – Schedulers Evaluation, Fixed Deep Sleep cycle 20 ms

Light	Scheduler	Throughput	F-Index	Delay	PLR	MOS
2	QEPEM	3.707	0.5894	9.8714	0	3.78
	PF	1.043	0.5350	17.6295	0.00059	3.86
	BCQI	1.566	0.1410	38.1408	0.4675	1.55
5	QEPEM	3.3865	0.6001	8.9632	0	3.66
	PF	1.0586	0.5362	17.3929	0.00043	3.87
	BCQI	1.3412	0.1470	33.8087	0.4494	1.55
10	QEPEM	2.6513	0.5393	10.5643	0.0024	3.75
	PF	1.0316	0.5385	17.8852	0	3.86
	BCQI	1.1071	0.1522	34.2655	0.4600	1.52
16	QEPEM	2.7837	0.5646	9.1295	0.0127	3.49
	PF	0.86255	0.4812	17.1979	0	3.87
	BCQI	0.7513	0.1523	36.6407	0.4634	1.53
20	QEPEM	2.3923	0.5617	10.7919	0.0013	3.87
	PF	0.8861	0.5191	19.4660	0	3.84
	BCQI	0.6382	0.1554	29.8075	0.4568	1.59

Table 5.4 summarizes the results of different Light Sleep Cycle with fixed Deep Sleep mode of 20 ms. The average values of distinctive performance parameters are given in terms of system throughput, throughput fairness index, packet delay, packet loss rate, and user's perception (MOS). The average value of packet delay shows that the QEPEM scheduler achieved the least delay followed by the PF scheduler, which has performed better than BCQI scheduler. The proposed QEPEM method performs best as compared to other methods in terms of Throughput, Fairness Index, and Delay, while in terms of PLR and MOS, QEPEM performs exceptionally than BCQI, but sometime its performance is close to PF. BCQI scheduler performed the worst in all cases, as it assigns radio resources to the limited UEs.



(a) Average Throughput

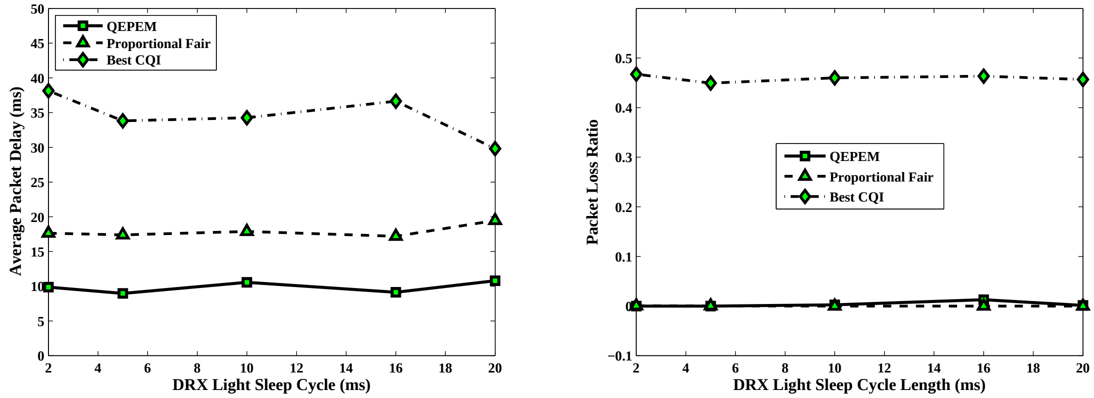


(b) Fairness Index

Figure 5.10 – Vary Light Sleep with Fixed Deep Sleep = 20 ms

Figure 5.10, shows the Average Throughput and Fairness Index for three scheduling method QEPEM, PF, and BCQI. The results show the impact of DRX Light Sleep duration along with fixed Deep Sleep duration equal to 20 ms. Figure 5.10a, depicts that the QEPEM performs best since it is designed to provide best fairness among the UEs by fulfilling the GBR UEs' requirements at the cost of lower system throughput. The results clearly represent that the QEPEM is least affected by the increase in sleep durations because it considers the DRX state of the UEs and user perception in order to maximize the QoE. The BCQI and PF scheduler performance degraded significantly when the system is working in power saving mode. The figure clearly shows that the QEPEM is performing in a superior way than the other schemes if duration of DRX sleep is increased. Figure 5.10b shows that QEPEM performs better as compared to other methods, while the performance of PF is close to proposed QEPEM. The BCQI performed the worst best in this case due to its resource distribution policy.

Figure 5.11 illustrates the effect of power saving on packet delay shown in Figure 5.11a, and packet loss rate presented in Figure 5.11b for the three scheduling methods. In case of VoIP communication, it is required that when a packet is created, it must reach the UE within 100 ms as per QCI characteristic of LTE networks, otherwise the packet will be discarded. It is observed that when the DRX Light Sleep duration increases, subsequently



(a) Average Packet Delay

(b) Average Packet Loss Rate

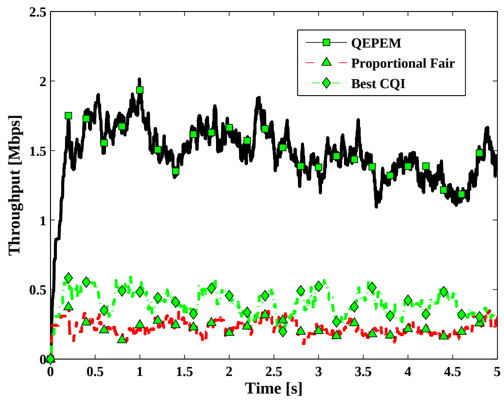
Figure 5.11 – Vary Light Sleep with Fixed Deep Sleep = 20 ms

packets start to have more delay, because the packet delay is directly proportional to the power being saved through the DRX sleep duration. Figure 5.11, depicts that QEPEM performed the best, and PF method came second in terms of packet delay and packet loss rate. The results show that both schedulers follow a linear pattern. QEPEM Scheme is designed to reduce the packet delays and losses while achieving the high throughput and fairness to improve the user's QoE. BCQI performs worst in terms of packet delay and packet loss rate, because it is designed to achieve maximum system throughput in normal operational mode, yet it disregards fairness and delay constraints.

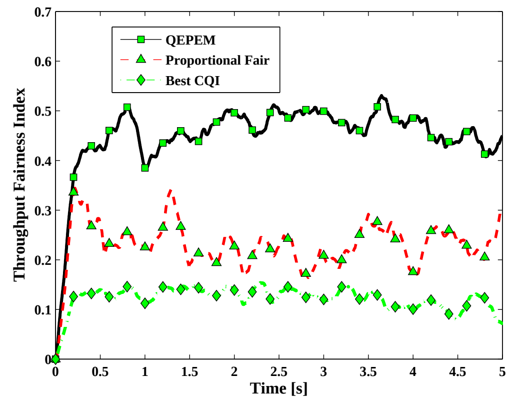
5.7.2 Performance Analysis with Fixed Light Sleep 10 ms

The impact of a power saving mechanism on user's QoE and QoS in the LTE networks will be evaluated by fixing the DRX Light Sleep Cycle to 10 ms and observing the effect of different DRX Deep Sleep Cycle duration. The impact of each Deep Sleep duration is evaluated, while the results are summarized in Table 5.5.

Figure 5.12 depicts the Average throughput and fairness index, when the DRX Light Sleep Cycle has a value of 10 ms with a DRX Deep Sleep Cycle duration set to 80ms. QEPEM has the best performance in terms of throughput and fairness, as compared to other scheduling schemes due to its efficiency of scheduling decision, which is based on



(a) Average Throughput



(b) Throughput Fairness Index

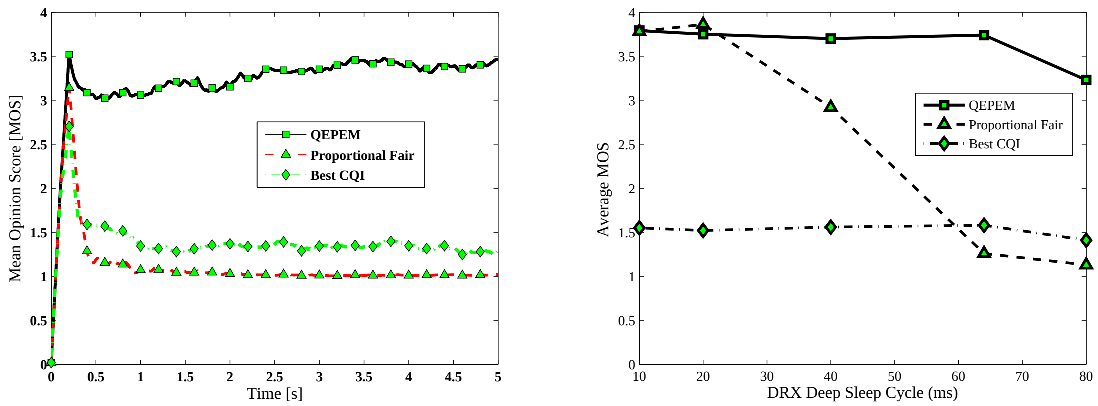
Figure 5.12 – Deep Sleep = 80 ms, Fixed Light Sleep = 10 ms

important parameters (e.g. DRX, MOS, GBR, etc.). In addition QEPEM, PF is performing better in contrast to traditional BCQI scheme. By increasing the duration of the Deep Sleep cycle, the Average throughput of all the scheduling schemes are reduced. Figure 5.12a shows that QEPEM again achieves the highest throughput than traditional schedulers, because it assigns the resources to those UEs that are in-active mode, which results to achieve high fairness index as shown in Figure 5.12b.

Figure 5.13 depicts the user's perceived QoE in the form of MOS values while using the three scheduling methods, when the DRX Deep Sleep Cycle has a value of 80 ms with a fixed DRX Light Sleep Cycle duration set to 10 ms. Figure 5.13a, clearly shows that QEPEM achieves a high user's satisfaction along with a large power saving at the UE. This is because QEPEM considers the user's perception and DRX status while making the scheduling decision. BCQI holds the second position, while PF has worst performance in this case scenario. Figure 5.13b represents the performance of three scheduling method using the Average MOS performance metric. It is observed that when the duration of the Deep Sleep cycle is increased then Average MOS of PF is significantly reduced. QEPEM method again achieves the highest user's satisfaction as compared to the other traditional methods. BCQI has nearly identical behaviour as it servers only the limited UEs that face almost the same network quality.

Table 5.5 – Schedulers Evaluation, Fixed Light Sleep cycle 10 ms

Deep	Scheduler	Throughput	F-Index	Delay	PLR	MOS
10	QEPEM	3.5172	0.5838	5.7893	0	3.79
	PF	1.3473	0.5643	8.6391	0	3.78
	BCQI	1.2000	0.1549	32.2102	0.4103	1.55
20	QEPEM	2.6513	0.5393	10.5643	0.0024	3.75
	PF	1.0316	0.5385	17.8852	0	3.86
	BCQI	1.1071	0.1522	34.2655	0.4600	1.52
40	QEPEM	2.2174	0.5178	19.9674	0.0098	3.70
	PF	0.52386	0.4211	38.2932	0.0517	2.92
	BCQI	0.74179	0.1431	42.2722	0.5346	1.56
64	QEPEM	1.9751	0.4815	30.1797	0.0125	3.47
	PF	0.30250	0.2918	49.9616	0.3565	1.26
	BCQI	0.65989	0.1255	48.7677	0.6073	1.58
80	QEPEM	1.5037	0.4605	37.0972	0.0250	3.23
	PF	0.23113	0.2369	53.2352	0.4865	1.13
	BCQI	0.409880	0.1239	57.9616	0.6298	1.41



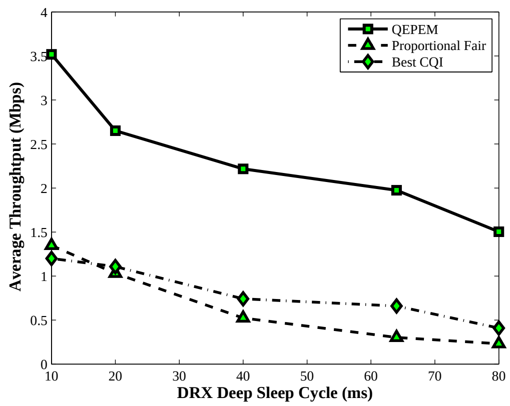
(a) Mean Opinion Score

(b) Average MOS Value

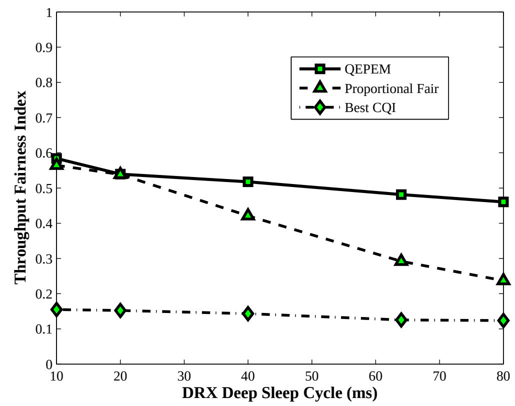
Figure 5.13 – Deep Sleep = 80 ms, Fixed Light Sleep = 10 ms

Table 5.5 sums up the performance of three schedulers QEPEM, PF, and BCQI in the forms of QoS parameters (throughput, fairness index, packet delay, and packet loss rate) that have high influence on the user's perceived QoE. When the duration of Deep Sleep Cycle increases, the performances of all schedulers are degraded. However, QEPEM has successfully managed the situation by considering the DRX and user's perception in its scheduling decision. QEPEM has the highest system throughput, fairness index, and least packet delay in comparison to the other schedulers, while in case of PLR and MOS, QEPEM has also better performance than PF unless the Deep Sleep has value 20 ms, where QEPEM performance is very close to PF. BCQI has the worst performance in all case scenarios, because it allocates the resources to fewer UEs by considering the channel quality.

Figure 5.14 illustrates the performance of QEPEM, PF, and BCQI in terms of QoS parameters, which are average throughput and fairness index. The system throughput is averaged over 5000 TTIs for each scheduler. The QEPEM performs better as compared to the other traditional schemes (PF and BCQI) in both performance parameters. In power saving mode, the performance of the PF, and BCQI degraded significantly in their respected order. The result clearly shows that the QEPEM is still performing better than the other schemes if the duration of the DRX Deep Sleep is increased. When the DRX Deep Sleep duration is increased continuously as shown in Figure 5.14, the QEPEM has the highest



(a) Average Throughput



(b) Fairness Index

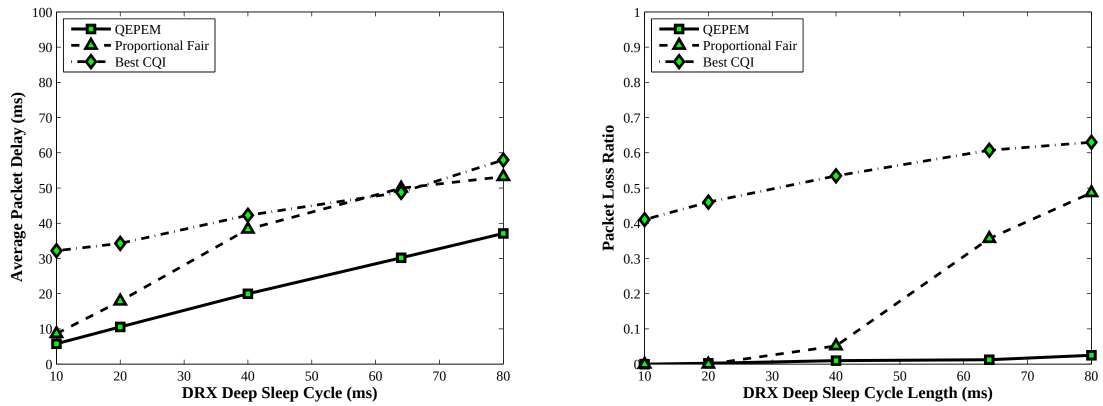
Figure 5.14 – Vary Deep Sleep with Fixed Light Sleep=10 ms

performance index as compared sto the other methods, but the PF experienced poor system throughput, as indicated by Figure 5.14a. Similarly, the performance of PF significantly degrades when the Deep Sleep duration exceeds more than 20 ms as shown in Figure 5.14b.

Figure 5.15 shows the performance of the three schedulers in terms of packet delay and loss rate. When the Deep Sleep duration increases, result packets start to get delayed, as the packet delay is directly proportional to the power being saved through the DRX Deep and the Light Sleep duration. The simulation results clearly show that QEPEM method performs best with less packet delay as indicated in Figure 5.15a, and low packet loss rate as shown in Figure 5.15b compared to other schedulers (PF and BCQI). The performance of PF is badly affected, as it has high packet loss rate when the duration of Deep Sleep increases from more than 40 ms. The BCQI has the worst performance in both performance metrics of packet delay and packet loss rate, due to its resource allocation policy.

5.8 Conclusion

This chapter discusses the general aspects of LTE wireless network. The main focus is to develop a downlink scheduling algorithm that manages the RT multimedia VoIP traffic by considering the distinct significant parameters. The resource allocation process mainly



(a) Average Packet Delay

(b) Average Packet Loss Rate

Figure 5.15 – Vary Deep Sleep with Fixed Light Sleep = 10 ms

depends on different scheduling parameters that play an important role in scheduling decision for achieving the desired QoS objective and high user satisfaction. In the proposed QEPEM algorithm, the main challenge is to acquire the user's perceived QoE for in-speech VoIP traffic, and it is possible by using the E-model. The E-model is an analytical model of voice quality, and it is used to find out the MOS value, which is the arithmetic average of user opinion.

The proposed QEPEM for LTE downlink scheduling uses the opportunistic scheduling approach for delay sensitive multimedia traffic (VoIP). It takes into account the six important scheduling dependencies that have the greater impact on QoS and QoE; which are user's MOS, Channel condition (CQI), Average throughput history, UE buffer status, GBR/non-GBR traffic, DRX status. The QEPEM method opts to enhance the QoE and provide better QoS by decreasing packet losses, improve fairness among UEs and meeting the QoS requirement of multimedia services. It has the capability to assure QoS in the power saving mode with high level of the users' satisfaction. The QEPEM method maximizes the user's QoE by using the user perception in its scheduling decision. The performance of QEPEM is compared with the traditional schemes according to different QoS attributes through simulations. From the simulation results, it is observed that PLR has more influence on QoE as compared to delay. The QEPEM method is evaluated in the power saving mode and the impact of the power saving on QoS and QoE is also examined. In the power saving environment, QEPEM performs remarkably better than the traditional schedulers with better user's experience, since it allocates resources efficiently and fairly among the UEs.

Chapter 6

Conclusions and Future Works

The communication system is always evolving that try to fulfil the increasing traffic demands, and provide good QoS to achieve high user satisfaction. The new concept of QoE consists of technical and non-technical aspects that directly/indirectly influence the user's perception, while QoS represents the network ability to provide service only from a technical aspect. Hence, QoE and QoS are different, but they are interdependent: because QoS is a key factor that has a high impact on the user perception. It is necessary to regard the QoS in order to study the QoE of different type of services. The service integrity in the perceptive of QoE can be defined in terms of QoS parameters such as jitter, delay, packet loss rate, throughput, etc. The accurate measurement of QoE, that is influenced by distinctive QoS parameters is not an easy task, but it is essential to develop an optimal method that considers the QoS for the best network performance and achieves high user's satisfaction.

In this dissertation, we describe the different methods that investigate a user's QoE in the view of technical and non-technical parameters using multimedia services (video and VoIP). Two approaches are discussed to gather the datasets for assessing the QoE of video service. The subjectively collected datasets is used to analysis the user's profile, that shed light on key factors, which help the network service providers to understand the behaviour and expectation of end-user. The datasets point out the role of different video quality along with QoS parameters that influence the user's perception. It motivates us to develop an adaptive video streaming method that changes the video quality based on network parameters and user device's properties. In the future communication network, the resources and power optimization are the key challenges, because multimedia services are resources hungry and consume more power. In this context, we also propose a scheduling method that allocates the resources to the end-user based on user's QoE, and optimize the power efficiency of user' device for LTE-A. In the following section, we summarize our main contribution, and present the future works.

6.1 Summary of Contributions

The objective of this dissertation is to investigate the concept of QoE for multimedia services through the analysis of technical and non-technical parameters, and quantify the performance of offered services, as well as their impact on end-users. We summarize our contribution as follows:

1. We present two subjective methods which are used to gather the datasets for assessing QoE of video service, and analyse the impact of different parameters. These methods are based on controlled, and un-controlled environmental approaches. In controlled approach, a testbed experiment is setup to measure the influence of different parameters on the user perceived QoE, while watching the video service. The impact of different parameters (QoS parameters, video characteristic, device type, etc.) on user perception is recorded in the form of MOS value. The subjective collected dataset is used to investigate the correlation between QoS and video QoE. Six ML classifiers are used to classify the collected dataset. In case of mean absolute error rate, it is observed that Decision Tree (DT) has a good performance as compared to all other algorithms. An instance classification test is also performed to select the best model, and results clearly show that performance of RF, and DT are approximately at the same level. Finally, to evaluate the efficiency of DT and RF, a statistical analysis of classification is done, and results show that RF performs slightly better than DT.

2. The datasets is also used to investigate the impact of different QoS parameters on user's profile, and comprehensive study of users' profile gives useful information for network service providers to understand the behaviour and expectation of end users. The analysis shows that interesting videos' content has more tolerance than

non-interesting videos' content. Similarly, the users for HD videos' content are more sensitive in the delay and packet loss, while for Non-HD videos' content, the users have more tolerance levels. Based on users' profile analysis, the network service provider can efficiently utilize their resources to improve user satisfaction.

3. In un-controlled environment, a crowdsourcing application tool is developed that can be used to investigate the users' QoE in real-time environment. The application tool uses the feedback form to subjectively record the user's perception. It can monitor and store the real time performance parameters of QoS (packet loss, delay, jitter and throughput). Instead of QoS networks, the tool also measures the real time performance characteristics of the end user device in terms of system memory, performance capacity, CPU usage and other parameters.

4. The client-side HTTP rate adaptive BBF method is proposed that adapts the video quality based on three main QoS parameters, such as dynamic network bandwidth, user's buffer status, and dropped frame rate. The BBF is evaluated with different buffer length, and it is observed that a longer buffer length is less affected with dynamic bandwidth, but it is also not efficiently utilized the network resources. The BBF is evaluated and compared with Adobe's OSMF streaming method. It is observed that BBF successfully manages situation as compared to OSMF, in terms of sudden drop of bandwidth, and dropped frame rate when the client system does not have enough resources to decode the frames. Additionally, BBF method optimizes the user's QoE by avoiding the stalling, and pausing during video playback.

5. The downlink scheduling algorithm $QEPEM$ is proposed for delay sensitive traffic (VoIP). The QEPEM method endeavours to enhance the QoE and provide better QoS by decreasing packet losses, improve fairness among the UE and considering the QoS requirement of multimedia service. It can assure QoS in the power saving environment with high users' satisfaction. The QEPEM method maximizes the user's QoE by using the user perception in its scheduling decision, and its performance is compared with the traditional schemes according to different QoS attributes through simulations. It is observed that packet loss rate has more influence on QoE as compared to delay. The QEPEM method is evaluated in the power saving mode and the impact

of the power saving on QoS and QoE is also examined. In the power saving environment, the QEPEM method performance is remarkably better than the traditional schedulers with better user's experience because it allocates resources efficiently and fairly among the UEs.

6.2 Future Research Directions

This dissertation addressed the challenges to investigate user QoE for multimedia services, and high light the impacts of different parameters on user perception. Several future research directions and open issues can be derived from our work. Some of our future research directions are followings.

1. The analysis of user's profile under different scenarios can provide key information to the network service provider that helps to understand the user behaviour and expectation. We shall analysis the users' profile with the influence of different factors and parameters, e.g. terminal types (HD TV, 10" tablets, smart mobile device, LCD screen), during travelling (car, bus, train, etc.) and we also apply the statistical analysis techniques.

2. The crowdsourcing is considering a key technique that is used to evaluate and measure the service quality in real environment where a user exposes its perceived perception. We shall extend the functionality of our proposed crowdsourcing application tool that will be added to the Firefox extension and Java application. It will help to analysis the impact of other parameters on user's QoE in the real time environment.

3. Internet is a collection of diverse network with different access techniques, that forced the network service providers to develop a solution for the unpredictable network characteristics. The rate adaptive video streaming method is developed to solve the problem by considering different parameters on the client side. In this context, we proposed the HTTP-based rate adaptive video streaming method *BBF*, that adapts the video quality by considering the three important QoS parameters, which are Bandwidth, Buffer, and dropped Frame rate observed on the client side. In the future work, we shall extend the proposed BBF method to optimize its performance by

measuring the real time user's QoE, and select the appropriate video quality based on user perceived QoE. The complete adaptive streaming model will be developed and evaluated.

4. The power saving method has direct influence on QoS of multimedia services, because more power saved will increase the packet delay that may cause packet loss and minimize user's perceived QoE. To overcome this problem, we shall optimize the DRX parameters that maximize the user's perception along power saving without getting more packet delay. The proposed QEPEM will be evaluated with other traffic models, e.g. video, ftp, and gaming and measure the impact of these traffic models along with the power saving mechanism on the user satisfaction. In case of mobility, the effects of UE handover between eNodeB will also observe, and we shall extend $QEPEM$ for the future mobile communication network, because user perception and efficient power utilization are key challenges in NGN.

Chapter 7

Version française abrégée

Introduction

Les services multimédia émergents deviennent un contributeur majeur dans un trafic IP en croissance permanente. Ces dernières années, nous avons été les témoins d'une croissance extrême des services multimédia, en particulier les services de diffusion vidéo en ligne, qui sont majoritaires dans le trafic Internet global. Selon les prévisions de Cisco, le trafic vidéo atteindra 69% du trafic total dans l'Internet en 2017, alors qu'il est déjà de 57% en 2012. Ce pourcentage ne tient pas compte des vidéos échangées à travers le système pair-à-pair. Cependant, si on fait la somme de toutes les formes de vidéo (télévision, vidéo à la demande, pair-à-pair), ce trafic représentera 80 à 90% du trafic global en 2017. En général, les opérateurs réseaux utilisent différentes méthodes pour améliorer la qualité de service (QoS) de bout-en-bout, mais ces méthodes se sont avérées insuffisantes, voire inappropriées dans certains cas, pour répondre à la demande de l'utilisateur final et garantir un certain niveau de qualité des services qui lui sont vendus. Par conséquent, les fournisseurs de service ont changé leur stratégie en se recentrant sur l'utilisateur en faisant de lui, non pas un simple spectateur, mais un véritable acteur de la chaîne de mesure de la qualité. Néanmoins, il est très difficile pour un fournisseur de service réseau de garantir une grande satisfaction utilisateur dans des réseaux divers avec de multiples technologies d'accès. Les systèmes de communication sans fil utilisent différentes technologies d'accès allant des standards IEEE (WLAN, réseaux locaux sans fil) au réseau cellulaire large-bande de quatrième génération (4G). Les prévisions de Cisco indiquent que le trafic mobile global va être multiplié par 11 d'ici 2018. Le trafic multimédia sera le contributeur majeur dans les communications sans fil. Un défi important de la cinquième génération (5G) sera de fournir ces services de manière efficace de façon à tenir compte des attentes des utilisateurs en termes de qualité. Pour palier ce problème, l'informatique dans le cloud est considérée comme un atout fondamental de l'architecture cellulaire 5G, en fournissant une plate-forme informatique puissante pour accepter des services vidéo ultra haute définition (télévision sur IP en direct, video 2D/3D, vidéo à la demande, jeux interactifs, ...) capables de satisfaire les utilisateurs. Le cloud améliore l'expérience utilisateur en permettant la gestion de ces services dans des centres de données distants. Grâce à cette tendance, un grand nombre de centres de données ont émergé, aidant au développement d'une pléthore de services internet. Dans le cloud, beaucoup d'applications et de services sont fournis aux utilisateurs de manière distante. Par conséquent, une qualité de service supérieure aux standards habituels est indispensable. Le concept de Qualité de l'Expérience (QoE) a attiré l'attention récemment, à la fois dans les réseaux filaires et sans fil, et particulièrement dans les réseaux du futur (ex : 5G). Son objectif principal est de considérer non seulement la QoS, mais aussi d'améliorer l'estimation de la qualité perçue par l'utilisateur. En réalité, le but d'un fournisseur de services réseau est d'offrir une bonne expérience utilisateur en utilisant le minimum de ressources réseau. Il est essentiel pour eux de considérer l'impact de chaque paramètre réseau sur la perception utilisateur, puisque leurs affaires dépendent largement de la satisfaction utilisateur. Dans ce contexte, il est nécessaire de comprendre les exigences du client/utilisateur en termes de qualité, et cet objectif est défini sous le terme de QoE. A ce titre, la communauté scientifique, en lien avec les fournisseurs de services réseau, a abordé ces problèmes en s'intéressant en particulier au développement de mécanismes permettant de mesurer la qualité perçue dans le cas de services multimédia. La QoE représente la qualité réelle telle que perçue par l'utilisateur lorsqu'il visionne une vidéo ou utilise un autre service. La QoE est donc définie comme « la mesure de l'acceptabilité globale d'une application ou service perçue subjectivement par l'utilisateur final. ». Par ailleurs, la croissance spectaculaire des périphériques électroniques (tablettes, smartphones, ...) avec des capacités décuplées, ajouté aux capacités des nouveaux réseaux sans fil, ont conduit à une forte croissance des services multimédia. D'un autre côté, les nouvelles tendances sur le marché des appareils électroniques ont permis le développement d'une palette impressionnante de périphériques mobiles intelligents et connectés, possédant suffisamment de puissance de calcul pour permettre le développement d'une large gamme de trafic multimédia. Parallèlement, il existe une demande croissante pour des services de données à haut débit. Le projet 3GPP (3rd Generation Partnership Project) a proposé une nouvelle technologie d'accès radio, le LTE et LTE-Advanced, qui a la capacité de fournir une plus grande bande passante et des latences faibles dans un réseau sans fil, dans le but de répondre à la demande des équipements utilisateurs avec une qualité de service acceptable. Un grand nombre d'applications de données sont aussi développées pour les appareils mobiles intelligents, ce qui encourage les utilisateurs à utiliser le réseau LTE de plus en plus souvent. A ce titre, la Voix sur IP (VoIP) et le streaming vidéo sont des services multimédia fondamentaux, qui sont utilisés de manière extrêmement courante. La VoIP est un service à bas coût très populaire qui permet d'appeler en utilisant le réseau IP. Le succès de la VoIP est principalement influencé par la satisfaction utilisateur, dans le contexte de la qualité des conversations téléphoniques, comparée aux services de téléphonie fixe conventionnels. Le défi principal pour les services VoIP est de fournir la même QoS qu'un réseau téléphonique conventionnel, c'est à dire la fiabilité et une certaine garantie de QoS. Dans les réseaux conventionnels, la qualité est gérée comme un unique plan de qualité, alors que dans les réseau de nouvelle génération (NGN), il est aussi nécessaire de prendre en compte la QoE des utilisateurs. Dans un système sans fil, le médium présente des comportements imprévisibles et différents pour chaque équipement. Dans ces circonstances, il est nécessaire de surveiller la QoE dans le réseau, pour chaque appel individuel, et non pas dans leur globalité. Nous nous plaçons ici dans le contexte du trafic VoIP dans un ordonnanceur LTE qui alloue des ressources radio en se basant sur la QoE des utilisateurs.

Sur un tout autre plan, l'explosion du trafic de streaming vidéo a engendré de profonds changements dans les technologies qui sont utilisées pour délivrer du contenu vidéo aux utilisateurs finaux sur Internet. Pour satisfaire le haut niveau d'exigence des utilisateurs, il est nécessaire d'analyser attentivement les services de streaming vidéo dans le but de comprendre le degré d'influence des paramètres (techniques et non techniques) sur la satisfaction utilisateur. Parmi ces facteurs, on trouve les paramètres réseau, représentés par la QoS. Délai, gigue et perte de paquet en sont les principaux paramètres, et ont une influence importante sur la satisfaction (ou insatisfaction) de l'utilisateur. En plus des paramètres réseau, d'autres facteurs externes ont un impact significatif sur la qualité perçue, en particulier la qualité de l'encodage de la vidéo, le type de terminal utilisé, ainsi que les facteurs relatifs à l'utilisateur lui-même. En général, on utilise deux méthodes pour mesurer la qualité de services multimédia : la méthode subjective et la méthode objective. La méthode subjective est proposée par l'union des télécommunications internationale (ITU-T), et est utilisée pour déterminer la perception utilisateur de la qualité d'un streaming vidéo. Le score MOS, Mean Opinion Score, est un exemple d'une méthode subjective dans laquelle les utilisateurs notent la qualité d'une vidéo en utilisant un score de 1 à 5, où 5 est la meilleure et 1 la pire qualité. Cependant, la méthode objective utilise différents modèles des attentes humaines et tente d'estimer les performances d'un service vidéo d'une façon automatisée, sans intervention humaine. Ces méthodes, subjectives et objectives, ont leur importance relative dans la mesure de la QoE. Elles sont complémentaires. Il est néanmoins très difficile de mesurer subjectivement le MOS d'une conversation donnée, car le MOS est une moyenne d'un grand nombre d'opinions utilisateurs. Par conséquent, de nombreuses méthodes de mesure de la qualité de la voix sont développées pour affiner l'estimation du MOS. L'E-model et la méthode PESQ (Perception Evaluation of Speech Quality) sont des méthodes objectives pour mesurer des scores MOS. PESQ ne peut pas être utilisée pour mesurer la QoE d'appels en temps réel car il a besoin du signal de référence pour le comparer au signal dégradé et calculer un score MOS. Dans cette thèse, nous avons utilisé l'E-model pour calculer le score MOS de conversations téléphoniques en utilisant la latence et le taux de perte de paquets, à l'aide de la métrique R-factor.

Dans cette thèse, organisée en six chapitres présentées ci-dessous, nous nous intéressons sur le plan scientifique à la commande du réseau en intégrant à la fois des aspects qualitatifs (perception du niveau de satisfaction de l'usager) et quantitatifs (mesure de paramètres réseau) dans l'objectif de développer des mécanismes capables, à la fois, de s'adapter à la variabilité des mesures collectées et d'améliorer la qualité de perception. Pour ce faire, nous avons étudié le cas de deux services multimédia populaires : le streaming vidéo, et la voix sur IP (VoIP).

Chapitre 2 – Etat de l'art

Ce chapitre apporte une synthèse des travaux actuels en rapport avec cette thèse. Le chapitre est divisé en trois sections différentes qui correspondent chacune à une contribution de la thèse. L'analyse de la QoE n'est pas une tâche aisée, car l'ensemble des paramètres qui influencent directement ou indirectement la qualité perçue par l'utilisateur doivent être pris en compte. Il existe différentes méthodes pour corréler les paramètres de QoS réseau avec la QoE ressentie par l'utilisateur. La plupart de ces méthodes sont basées sur des expérimentations sur des plates-formes d'essai présentant différents équipements, protocoles et outils. Les jeux de données collectés sont analysés pour observer l'influence des divers paramètres sur la qualité perçue. Nous construisons aussi un profil des utilisateurs à partir des résultats de ce banc d'essai. De plus, des approches de streaming vidéo adaptatif sont évaluées via ce testbed, en mesurant les performances des trois éléments clés (client, serveur et réseau). Enfin, nous discutons des méthodes d'ordonnancement qui permettent d'allouer les ressources radio aux équipements utilisateur (UE) en se basant sur plusieurs critères. Le rôle des méthodes d'économie d'énergie est aussi discuté dans le contexte de plusieurs systèmes sans fil, ainsi que leur impact sur les performances du système.

Chapitre 3 – Méthodologies pour l'évaluation subjective de la QoE du streaming vidéo

Dans ce chapitre, nous discutons deux approches utilisées pour collecter des jeux de données subjectifs pour l'évaluation de la QoE d'un utilisateur utilisant un service vidéo. Il s'agit de l'approche d'environnement contrôlé, et non-contrôlé. Dans un environnement contrôlé, un banc d'essai en laboratoire est implémenté pour collecter les données en fonction de variations contrôlées de multiples paramètres (paramètres QoS, caractéristique de la vidéo, type de terminal, ...). Ces résultats sont stockés sous la forme d'un score MOS. Nous utilisons ce jeu de données pour analyser la corrélation entre la QoS et la QoE, à partir de six classificateurs issus de l'apprentissage machine. Le jeu de données contient les profils des utilisateurs, que nous utilisons aussi pour investiguer l'impact des différents paramètres sur la perception utilisateur. Parallèlement, nous avons développé un environnement non contrôlé basé sur le crowdsourcing. Cet outil collecte les opinions des utilisateurs sur la qualité dans leur propre environnement (terminal, navigateur Web, vidéo visionnée). Pendant le visionnage, l'outil enregistre les performances réseau en temps réel dans une base de données SQL locale. De plus, il mesure et enregistre les performances en temps réel du terminal utilisateur, en termes de mémoire utilisée, utilisation du CPU, du réseau.

Chapitre 4 – Régulation de la QoE pour le streaming adaptatif

Ce chapitre décrit un système complet de vidéo adaptative et souligne les éléments jouant un rôle prépondérant dans la régulation du streaming vidéo côté client. Nous discutons de l'architecture utilisée, consistant essentiellement en trois éléments : le client, le réseau de distribution et le serveur. Nous proposons un nouvel algorithme adaptatif qui sélectionne dynamiquement les segments les mieux adaptés en fonction des conditions du réseau et des paramètres du client. La méthode proposée, appelée BBF (Bandwidth, Buffer and dropped Frame rate), tient compte des trois paramètres suivants pour réguler un streaming vidéo sur HTTP : la bande passante, la taille de la mémoire tampon et le taux de trames perdues. Le BBF est évalué en utilisant différentes tailles de buffer, et les résultats montrent qu'un buffer large est moins affecté par des variations de débit, mais il ne permet pas d'utiliser au mieux les ressources du réseau. Les performances de BBF sont comparées à la méthode de streaming OSMF d'Adobe, et les résultats montrent que notre méthode traite correctement les situations de chute brutale du débit de la vidéo et l'augmentation des pertes de paquets quand le client n'a plus suffisamment de ressources pour décoder les trames. Dans le cas d'un buffer de petite taille, le BBF passe automatiquement à une qualité vidéo moins élevée et optimise la QoE utilisateur en évitant le blocage et la mise en pause de la vidéo.

Chapitre 5 – Ordonnanceur LTE basé sur la QoE et l'économie d'énergie

Ce chapitre présente une vue générale des réseaux sans fil LTE-A. Nous nous focalisons sur le mécanisme d'ordonnancement dans le sens descendant. En effet, celui-ci est plus important car il convoie beaucoup plus de trafic que le sens montant. Nous proposons un algorithme d'ordonnancement qui tient compte de la QoE pour des trafics sensibles au délai (type VoIP). L'architecture générale d'un ordonnanceur LTE est présentée, ainsi que les principaux éléments qui interfèrent dans ce mécanisme. Les performances d'un nouvel ordonnanceur, appelé QEPEM (QoE Power Efficient Method) sont présentées. Le but est de développer un ordonnanceur qui alloue les ressources radio aux utilisateurs en se basant sur la QoE perçue par cet utilisateur, en relation avec l'utilisation de méthode d'économie d'énergie (DRX, Discontinuous Reception). La performance de QEPEM est évaluée et comparée à des méthodes d'ordonnancement traditionnelles, à savoir Proportional Fair (PF) et le Best Channel Quality Indicator (BCQI). La méthode QEPEM a pour but d'améliorer la QoE et fournir une meilleure QoS en diminuant la perte de paquets, améliorer l'équité parmi les différents utilisateurs tout en satisfaisant aux exigences des services multimédia. Les résultats montrent que QEPEM offre des performances supérieures aux ordonnanceurs traditionnels ainsi qu'une meilleure QoE, en allouant équitablement les ressources parmi les utilisateurs.

Chapitre 6 – Conclusion et travaux à venir

Ce chapitre conclut ce travail de thèse et propose quelques pistes d'amélioration pour des travaux futurs. Le chapitre résume les résultats et défis rencontrés pour mesurer et maintenir une certaine QoE utilisateur dans les services multimédia, notamment en tenant compte des différents paramètres qui influent sur la perception utilisateur. Plusieurs directions de recherche sont proposées, ainsi que de nouvelles problématiques issues de nos travaux, qu'il reste à résoudre.

Appendix A

HTTP-based Adaptive Video Streaming

A.1 Introduction

This appendix discusses some background information related to HTTP-based video streaming. Generally, video streaming services run over either managed network or unmanaged network. In a managed network, the video services use the multicast transport and try to maintain the required QoS characteristics, such as cable and IPTV services. However, it is a challenging task to achieve the certain QoS features, when video services run over an unmanaged network. The main video streaming technologies that run over unmanaged networks are Adobe Flash, Apple QuickTime, Microsoft Windows Media, and in addition to the emerging adaptive video streaming technologies consist in Adobe's HTTP Dynamic Streaming (HDS), Apple's HTTP Live Streaming (HLS), Microsoft's Smooth Streaming (MSS), and MPEG's Dynamic Adaptive Streaming over HTTP (DASH). These streaming technologies send the video content to end-user using the unicast connection. This appendix discusses the briefly video streaming technologies over unmanaged networks, and we focus our discussion on the Adobe's HDS adaptive video streaming technology.

A.2 Media Streaming

The media streaming content is transmitted among the different end-user over the IP networks by using distinct methods. Generally, a selection of appropriate method is based on the type of media content and underlying network conditions, because it needs certain level of QoS features such as low packet loss, jitter, delay, and efficient transmission. The media streaming protocol defines the structure of packets and transmission method. Nowadays, many protocols are implemented for efficient media streaming, and we can classify them into two categories: push-based and pull-based protocols [8].

A.2.1 Push-based Media Streaming Protocols

In push-based media streaming protocols, when the server and the client connection is established then server pushes media content (packets) to the client, until a client ends the session. It is a server driven approach, where a server maintains the session and listens messages from the client to change the session-state. The well-known session control protocols used in push-based media streaming is Real Time Streaming Protocol (RTSP). Generally, push-based protocols use Real-time Transport Protocol (RTP) along with User Datagram Protocol (UDP). In RTP/UDP, the client/server communication relies on application-level implementation as compare to underlying transport protocol [8], where RTP performs best for low delay and best-effort transmission. In conventional push-based method, the server encodes the media content according to client consumption capacity, and maintains the certain buffer level to avoid buffer underflow by switching to lower bitrate stream.

A.2.2 Pull-based Media Streaming Protocols

In pull-based media streaming protocols, client performs a key role, and makes a decision for requesting the appropriate content from the media server. Therefore, the server only active just to respond the client's requests, otherwise it is in an idle state. The client requests the media streaming content based on device properties and network bandwidth. HTTP is a main protocol for Internet download, and it is also principal protocol for pull-based media delivery. Progressive download method is an example of pull-based protocol, that widely uses for downloading the media streaming on IP based networks. In pull-based streaming protocols, the client avoids the buffer underflow by using the bitrate adaptation method, where a client requests the suitable media segment according to device states, and available network bandwidth.

A.3 Video Streaming Method

Video streaming over HTTP is highly dominant due to the availability of Internet support on many devices, and it easily traverses NATs and firewalls, unlike other media transport protocols such as RTP/RTSP. Both progressive download and adaptive streaming methods use the HTTP as a primary protocol to transport the media content to the client. HTTP-based servers are possibly more scalable than push-based streaming servers, because it maintains minimum state information on the server side. Video streaming over HTTP is easier and cheaper to move data closer to network users, and the video file segment is just like a normal Web object. It also provides opportunity to CDNs increasing their scalability of content distribution [8].

A.3.1 Progressive Download

Earlier, HTTP-based video streaming application used the progressive download method (HTTP over TCP) and thanks to its simplicity this method becomes very popular for viewing online video contents. In progressive download, the client requests the video content to the server via an HTTP-based command, and it begins quickly pulling content from the server until it does not completely download the video. The client player starts playing the video, when a desired minimum buffer level is fill-up, and it continues playback the video without any interruption, until a sufficient client buffer level is filled. The buffer underflow can occur when the playback rate is more than the download rate due to insufficient network bandwidth.

This method has some limitation that degrades the QoE, because it lacks the rich features of video streaming, e.g. trick modes such as fast-forward seek/play, rewind, and often freezing or rebuffering due to the shortage of bandwidth. The new emerging approach for adaptive streaming not only replaces the progressive download, but it also covers the shortcoming features. The adaptive streaming is a pull-based media streaming approach that consists in progressive download and a streaming method [8].

A.3.2 Adaptive Streaming

The innovation in the HTTP video streaming was started by Move Networks, it is called Adaptive Streaming. This adaptive streaming increase the quality and resolution of video content according to the handling capability of the user device, throughout the data network. The adaptive streaming server maintains different copies of the same video content that vary in a bitrate, and client can switch to high-quality content according to its available bandwidth. There are a number of adaptive video streaming methods are available, but these are not penetrated very well in the market, which are 3GPP's Adaptation HTTP Streaming (AHS) release 9 specification, HTTP Adaptive Streaming (HAS) from Open TV Forum.

A.4 Adaptive Video Delivery Components

The adaptive video streaming have some new functionalities that must be added in the networks, and service providers must implement the fundamental CDN components. The most important components in HTTP adaptive video streaming are shown in Figure A.1 which are following: Transcoder/Encoder, Packager (also called fragmenter, segmenter and chunking), and CDN.

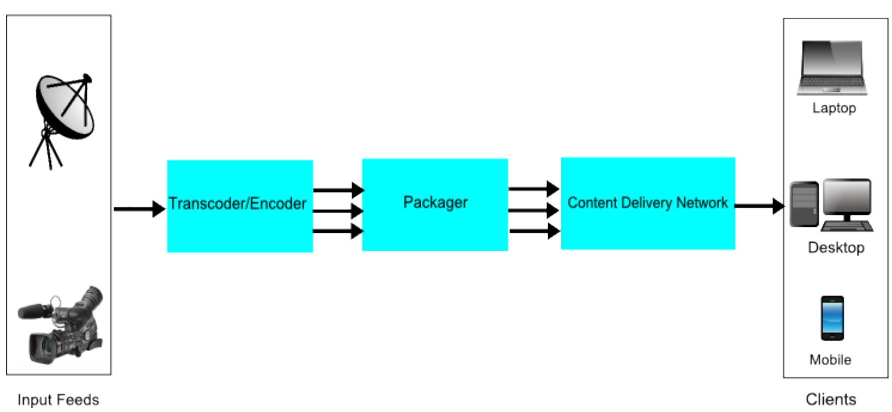


Figure A.1 – Adaptive Video Delivery Components

Transcoder/Encoder

The main function of transcoder/encoder is to prepare the media file for the packager. It takes the incoming baseband or IP digital video, and converts it into a multi-stream output profile of different bit-rates and resolutions that are suitable for the end-user device. The transcoder/encoder provides different profiles for each input video, because QoE of an end user mainly depends on a number of profiles. A large number of profiles resulting to support more devices and a better QoE, but it requires more space on the server.

Packager

The adaptive streaming uses the state-less protocol (HTTP), where the video file is broken into small pieces of HTTP files i.e. fragments, segments or chunks. The process of fragmentation, segmentation or chunking can be done in the transcoder or it can be processed to the Packager component. Generally, each segment lasts between 2 to 10 seconds. It supports the live streaming and also on-demand video. Packager is the central main component of the adaptive streaming system, which takes the output from the transcoder and converts the video for delivery according to the protocol. The video segment is delivered either through HTTP pull or HTTP PUT/POST command. Packager has an encryption capability, and it encrypts each outgoing segment in the compatible format for the delivery protocol. It also works with a third-party key management system that manages and distributes the key to end users. The generation of the manifest or playlist is a key function of this component.

Content Delivery Network (CDN)

A Content Delivery Network (CDN) is based on generic HTTP server/caches for streaming the video contents over HTTP, and it requires specialized servers at each node. It is very important that CDN should have the ability to handle the large number of segments, and similarly, support substantial number of video contents.

A.5 HTTP-based Adaptive Video Streaming Methods

The evolution of the adaptive video streaming leads to a new set of standards from well-known organizations, i.e., Adobe, Microsoft, Apple, and MPEG. These standards are widely adopted as they increase user's QoE by streaming the video service over HTTP, but in an adaptive manner, according to network conditions and device characteristics. The HTTP adaptive streaming technologies provided by these organizations are Adobe's HDS, Microsoft's MSS, Apple's HLS, and 3GPP/MPEG's DASH.

A.5.1 Adobe HTTP Dynamic Streaming (HDS)

Adobe HTTP Dynamic Streaming (HDS) uses the MP4 fragment format (F4F) for both live and on-demand media streaming. It was developed after the MSS and HLS standard. It uses the same structure that adjusted the video quality for improving the user's QoE by considering the client network speed and processing power, using the standard HTTP protocol infrastructures. The HDS provides the best viewers' streaming experience to a large number of end devices and platforms that support Adobe Flash software. There are two tools developed by Adobe for preparing the media streams into a fragmented format: the File Packager used to prepare on-demand media and Live Packager used to prepare live RTMP streams. These two packagers are used to generate MP4 fragment files (F4F), an XML-based manifest file (F4M) and optionally provide content protection.

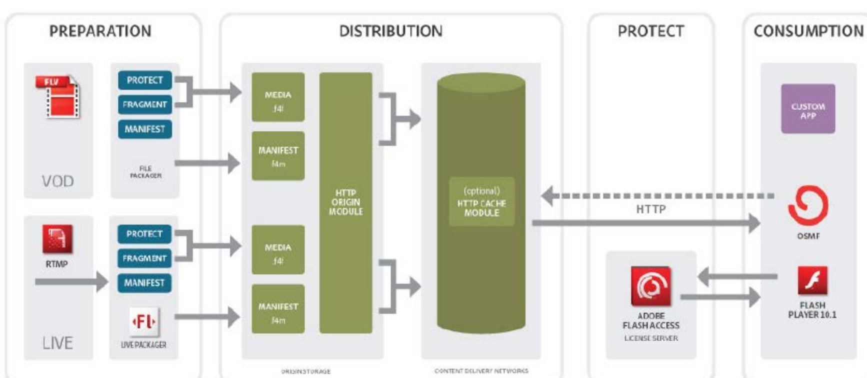


Figure A.2 – Preparation, Distribution, Protetionc and Consumption of HDS [39]

A.5.2 Microsoft Smooth Streaming (MSS)

In 2008, Smooth Streaming was announced by Microsoft as a part of its Silverlight architecture. It has core properties of adaptive video streaming. Video content is broken into small segments, delivered over HTTP, and multiple bit-rates that allow an end-user to dynamically and seamlessly switch from one bit-rate to another, based on the network condition, to increase its QoE. The resulting user experience is reliable and offers a consistent playback without stutter, buffering or congestion, in other words, Smooth. The MSS uses the ISO/IEC 14496-12 ISO Base Media File Format specification, also known as the MP4 file specification. MP4 is a lightweight container format with fewer overheads, and it is used to deliver a series of segments for smooth video streaming. The Smooth Streaming consists of two formats; the disk file format and the wire format. Normally, a full-length video is stored as a single file on the disk that is encoded with the specific bit rate, and during the streaming, it is transferred as sequences of small fragments (segments or chunks). The disk file format defines the structure of continuous files on the disk and on the other hand; the wire format defines the structure of each segment/chunk that is transferred from the server to the client. The file format of MSS is shown in Figure A.3. The file structure starts with file-level metadata $'moov'$ that represents the file, while the fragment boxes describe the fragment level metadata ($'moof'$) and the media data ($'mdat'$). The file structure ends with a $mfra$ index which helps in seeking within the file.

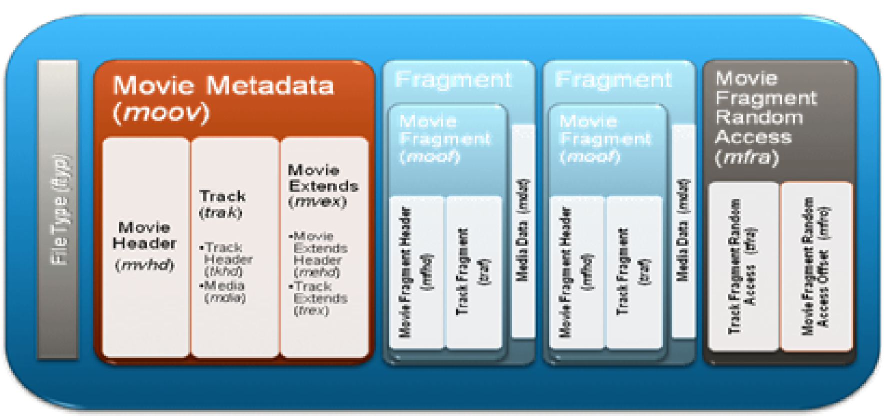


Figure A.3 – MSS File Format [80]

The Web server searches through the MP4 file to find a video fragment that is requested by the client player. The requested fragment of the file is sent to the client over the wire, hence the name 'wire format'. The format of fragment is shown in Figure A.4.

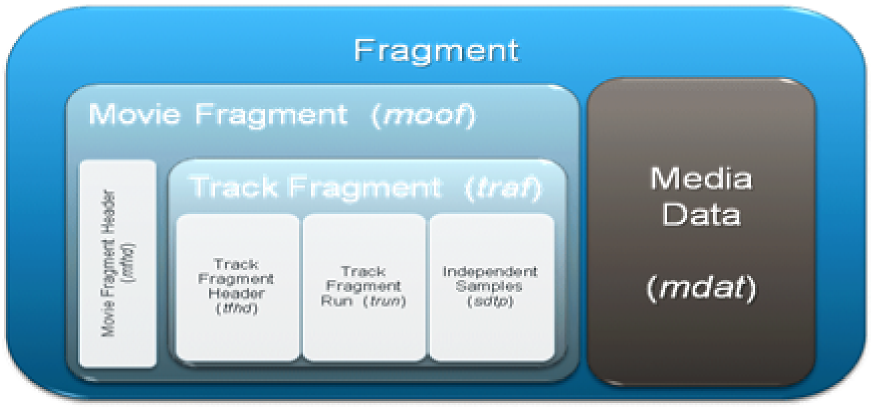


Figure A.4 – MSS Fragment Format [80]

A.5.3 Apple HTTP Live Streaming (HLS)

Apple chose the MOV file format as its adaptive streaming technology, unlike the well-known ISO MPEG file format. It allows to send the audio and video over HTTP from a simple Web server for playing on different kinds of IOS-based end devices, such as iPod, iPad, iPhone, Apple TV and desktop Mac OS X computers. The Safari web browser is a client software that plays HTTP Live streams using the tag. In HLS, the adaptive transport of video streaming is achieved by sending sequences of small files of video/audio that generally last 10 seconds, known as media segment files. Apple provides a free tool to generate the media segment and playlists (manifest file) for on-demand and live streams. The basic configuration architecture of HLS is shown in Figure A.5. The server components (media encoder and segmenter) have the responsibility to take the input from the source media, encode them into the MPEG-2 Transport Stream (TS), and split them into a series of TS files that encapsulate both audio and video in a format that is suitable for delivery to an end-user device. The web server is the main part of the distribution component, that accepts and responds to the client requests. The client software is responsible for generating the appropriate media segment request, download and reassemble them so that the media stream can playback in a continuous manner, to maintain a high user QoE.

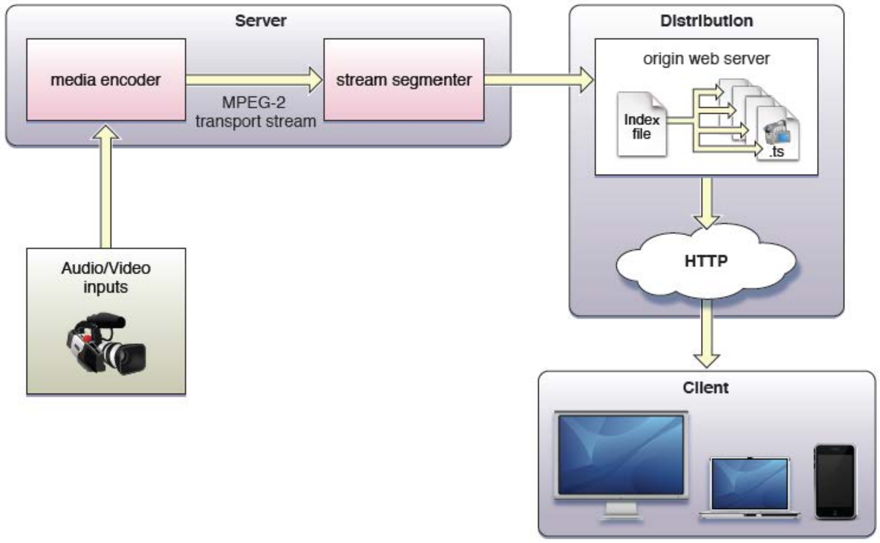


Figure A.5 – HLS Basic Configuration Architecture [40]

A.5.4 MPEG-Dynamic Adaptive Streaming over HTTP (DASH)

The Moving Picture Expert Group (MPEG) has developed many multimedia standards, including MPEG-2, MPEG-4, MPEG-7, MPEG-21. Recently, the group developed a standard for streaming multimedia over the Internet (HTTP). This standard is known as MPEG-DASH or simply DASH. The format used by the DASH standard is similar to HDS, MSS, and HLS, where the index files (manifest or playlist file) describe the order in which segments or chunks are downloaded and played for continuous media streaming. Figure A.6 shows a simple DASH streaming scenario between an HTTP server and the DASH client. In this figure, the multimedia content is captured and stored on a server and delivered to the client using HTTP. The server contains two content parts: the first one is the Media Presentation Description (MPD), which describes a manifest file about the available contents, including various alternative formats, URL addresses, and other characteristics; the second part is the segment part, which contains the actual multimedia bitstreams in the form of chunks, in single or multiple files.

To play the content, the DASH client first obtains the manifest or playlist file (i.e. MPD). The MPD can be delivered using HTTP or other transport's methods, e.g. email, thumb drive, broadcast. Initially, the DASH client parses the MPD, and it learns about the program timing, media-content availability, media types, resolutions, minimum and maximum

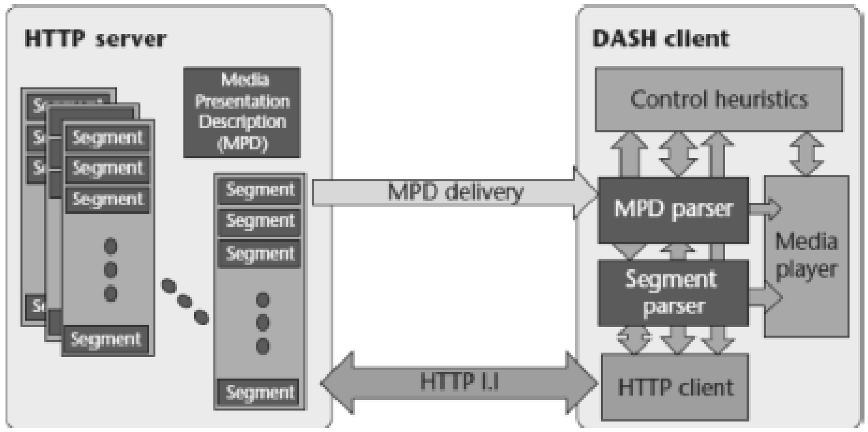


Figure A.6 – DASH Streaming Scenario [96]

bandwidth, and the existence of various encoded alternatives of multimedia components, accessibility features and required digital rights management (DRM), media-component locations on the network, and other content characteristics. After parsing the MPD, the DASH client selects the appropriate encoded segment and starts streaming the content by fetching the segments using HTTP GET requests.

The appropriate buffering handling allows network throughput variations, and the client continues fetching the successive segments and monitors the fluctuations in network bandwidth. Based on its measurement results, the client decides how to adapt according to the available bandwidth, and fetches the segments of different qualities (lower or higher bitrates) to avoid a buffer starvation [96]. Buffering plays a vital role for uninterrupted or smoothed streaming, which in turn improves the client's QoE. The DASH specification only defines the MPD and the segment formats. The delivery of the MPD and the media-encoding formats containing the segments, as well as the client behavior for fetching, adaptive heuristics, and playing content, are not considered in MPEG-DASH's scope [97].

Bibliography

[1] ISO/IEC 23009-1:2012. Part 1: Media presentation description and segment formats. In *Information technology – Dynamic adaptive streaming over HTTP (DASH)*, Jan. 2013.

[2] Florence Agboma and Antonio Liotta. Qoe-aware qos management. In *Proceedings of the 6th International Conference on Advances in Mobile Computing and Multimedia*, MoMM '08, pages 111–116. ACM, 2008.

[3] K. Aho, I. Repo, T. Nihtila, and T. Ristaniemi. Analysis of VoIP over HSDPA performance with discontinuous reception cycles. In *Sixth International Conference on Information Technology: New Generations (ITNG)*, pages 1190 –1194, April 2009.

[4] Kari Aho, Tero Henttonen, Jani Puttonen, Lars Dalsgaard, and Tapani Ristaniemi. User equipment energy efficiency versus lte network performance. *International Journal On Advances in Telecommunications*, 3(4):27 – 38, April 2011.

[5] Saamer Akhshabi, Lakshmi Anantakrishnan, Ali C. Begen, and Constantine Dovrolis. What happens when http adaptive streaming players compete for bandwidth? In *Proceedings of the 22nd International Workshop on Network and Operating System Support for Digital Audio and Video*, NOSSDAV '12, pages 9–14, 2012.

[6] Laurent Miclet Antoine Cornuéjols and Jean-Paul Haton. *Apprentissage artificiel - Concepts et algorithmes*. EYROLLES, 2010.

[7] Win Thanda Aung and Khin Hay Mar Saw Hla. Random forest classifier for multi-category classification of web pages. In *Services Computing Conference, 2009. APSCC 2009. IEEE Asia-Pacific*, pages 372–376, Dec 2009.

[8] AC. Begen, T. Akgul, and M. Baugher. Watching video over the web: Part 1: Streaming protocols. *IEEE Internet Computing*, 15(2):54–63, March 2011.

[9] Kian Chung Beh, S. Armour, and A. Doufexi. Joint time-frequency domain proportional fair scheduler with HARQ for 3GPP LTE systems. In *IEEE 68th Vehicular Technology Conference(VTC Fall)*, pages 1 – 5, Sept. 2008.

[10] Huang Bo, Tian Hui, Chen Lan, and Zhu Jianchi. DRX-aware scheduling method for delay-sensitive traffic. *IEEE Communications Letters*, 14(12):1113 – 1115, Dec. 2010.

[11] D.C.A. Bulterman. Smil 2.0 part 1: overview, concepts, and structure. *MultiMedia, IEEE*, 8:82–88, October 2001.

[12] Jili Chen, Kebin Huang, Feng Wang, and Huixia Wang. E-learning behavior analysis based on fuzzy clustering. In *3rd International Conference on Genetic and Evolutionary Computing, (WGEC '09).*, pages 863–866, 2009.

[13] Kuan-Ta Chen, Chi-Jui Chang, Chen-Chi Wu, Yu-Chun Chang, and Chin-Laung Lei. Quadrant of euphoria: a crowdsourcing platform for qoe assessment. *Network, IEEE*, 24(2):28–35, 2010.

[14] JungYul Choi, Min-Gon Kim, Hongkyu Jeong, and Hong-Shik Park. Power-saving mechanisms for energy efficient IEEE 802.16e/m. *Journal of Network and Computer Applications, Elsevier*, 35(6):1728 – 1739, 2012.

[15] H. Claussen. Efficient modelling of channel maps with correlated shadow fading in mobile radio systems. In *IEEE 16th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, volume 1, pages 512–516, 2005.

[16] R. G. Cole and J. H. Rosenbluth. Voice over ip performance monitoring. *SIGCOMM Computer Communication*, 31(2):9–24, April. 2001.

[17] L. De Cicco, V. Caldaralo, V. Palmisano, and S. Mascolo. Elastic: A client-side controller for dynamic adaptive streaming over http (dash). In *Packet Video Workshop (PV), 2013 20th International*, pages 1–8, Dec 2013.

[18] L. De Cicco, G. Cofano, and S. Mascolo. A hybrid model of the akamai adaptive streaming control system. In *IFAC World Congress*, August 2014.

[19] L. De Cicco and S. Mascolo. An adaptive video streaming control system: Modeling, validation, and performance evaluation. *IEEE/ACM Transactions on Networking*, 22(2):526–539, April 2014.

[20] O. Delgado and B. Jaumard. Joint admission control and resource allocation with GoS and QoS in LTE uplink. In *IEEE GLOBECOM Workshops (GC Wkshps)*, pages 829 –833, Dec. 2010.

[21] Adobe Media Server Developer. In *Recommendation, Video Endcoding for HTTP Dynamic Streaming on Flash Platform*, 2010.

[22] Lijing Ding and R.A. Goubran. Speech quality prediction in voip using the extended e-model. In *IEEE Global Telecommunications Conference,(GLOBECOM '03)*, volume 7, pages 3974–3978 vol.7, 2003.

[23] S.N. Donthi and N.B. Mehta. Performance analysis of subband-level channel quality indicator feedback scheme of lte. In *National Conference on Communications (NCC)*, pages 1–5, Jan 2010.

[24] DummyNet. Network emulation tool. In *Available [Online] "http://info.iet.unipi.it/ luigi/dummynet/"*.

[25] C. Eduardo, Z. Sherali, L. Mikołaj, C. Marilia, and M. Andreas. Recent advances in multimedia networking. *Multimedia Tools and Applications*, 54(3), 2011.

[26] 3GPP; LTE RAN enhancements for diverse data applications. In *RAN Plenary Contribution, RP-110410*, 2011.

[27] International Telecommunication Union (ITU-T); Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs. In *Recommendation P.862*, Feb. 2001.

[28] Yong Fan, P. Lunden, M. Kuusela, and M. Valkama. Efficient semi-persistent scheduling for VoIP on EUTRA downlink. In *IEEE 68th Vehicular Technology Conference(VTC Fall)*, pages 1 – 5, Sept. 2008.

[29] International Telecommunication Union (ITU-T); Methodology for derivation of equipment impairment factors from subjective listening-only tests. In *Recommendation P.833*, Nov. 2001.

[30] International Telecommunication Union (ITU-T): Guidelines for evaluation of radio transmission technologies for IMT-2000 , Tech. Rep. In *Recommendation ITU-R M.1225*, 1997.

[31] International Telecommunication Union-Telecommunication (ITU-T); Subjective Video Quality Assessment Methods for multimedia applications. In *Recommendation P.910*, 2008.

[32] 3GPP; LTE Physical Layer Framework for Performance Verification Radio Access Network (RAN). In *TSG-RAN1 no.48, R1-070674*, 2007.

[33] International Telecommunication Union (ITU-T); Methods for subjective determination of transmissiom quality. In *Recommendation P.800*, 1996.

[34] S. Fowler. Study on power saving based on radio frame in LTE wireless communication system using DRX. *IEEE Globecom Joint Workshop of SCPA and SaCoNAS*, Dec. 2011.

[35] S. Fowler, R.S. Bhamber, and A. Mellouk. Analysis of adjustable and fixed DRX mechanism for power saving in LTE/LTE-Advanced. In *IEEE International Conference on Communications (ICC)*, June 2012.

[36] H. French, Jie Lin, Tung Phan, and A.C. Dalal. Real time video qoe analysis of rtmp streams. In *Performance Computing and Communications Conference (IPCCC), IEEE 30th International*, pages 1–2, Nov 2011.

[37] B. Gardlo, M. Ries, T. Hossfeld, and R. Schatz. Microworkers vs. facebook: The impact of crowdsourcing platform choice on experimental results. In *Fourth International Workshop on Quality of Multimedia Experience (QoMEX)*, pages 35–36, 2012.

[38] Young-Tae Han, Min-Gon Kim, and Hong-Shik Park. A novel server selection method to achieve delay-based fairness in the server palm. *IEEE Communications Letters*, 13(11):868–870, November 2009.

[39] HTTP Dynamic Streaming (HDS). HDS) on the Adobe Flash Platform, Technical white paper., month = Nov., year=2013,. In *[Online] "http://www.adobe.com/httpdynamicstreaming/pdfs/httpdynamicstreaming_wp_ue.pdf"*.

[40] Apple HTTP Live Streaming (HLS). Overview hls. In *[Online] "https://developer.apple.com/library/mac/documentation/StreamingMediaGuide.html"*, Nov. 2013.

[41] T. Hosfeld, S. Biedermann, R. Schatz, A. Platzer, S. Egger, and M. Fiedler. The memory effect and its implications on web QoE modeling. In *23rd International Teletraffic Congress (ITC)*, pages 103 –110, Sept. 2011.

[42] T. Hossfeld, M. Seufert, M. Hirth, T. Zinner, P. Tran-Gia, and R. Schatz. Quantification of YouTube QoE via crowdsourcing. In *IEEE International Symposium on Multimedia (ISM)*, pages 494–499, 2011.

[43] Te-Yuan Huang, Nikhil Handigol, Brandon Heller, Nick McKeown, and Ramesh Johari. Confused, timid, and unstable: Picking a video streaming rate is hard. In *Proceedings of the ACM Conference on Internet Measurement Conference (IMC)*, pages 225–238, 2012.

[44] Te-Yuan Huang, Ramesh Johari, Nick McKeown, Matthew Trunnell, and M Watson. A buffer-based approach to rate adaptation: Evidence from a large video streaming service. In *Proceedings of the ACM Conference on Special Interest Group on Data Communication (SIGCOMM)*, August 2014.

[45] Te-Yuan Huang, Ramesh Johari, Nick McKeown, Matthew Trunnell, and Mark Watson. Using the buffer to avoid rebuffers: Evidence from a large video streaming service. In *arXiv:1401.2209,*, January 2014.

[46] Josep Colom Ikuno, Martin Wrulich, and Markus Rupp. System level simulation of LTE networks. In *IEEE 71st Vehicular Technology Conference (VTC Spring)*, Taipei, Taiwan, May 2010.

[47] International Telecommunication Union (ITU-T); Transmission impairments due to speech processing. In *Recommendation G.113*, Nov. 2007.

[48] 3GPP; DRX Parameters in LTE; TSG RAN WG2 LTE Contribution. In *Technical Specification, TS 36.300, N. R2-071285,*, 2007.

[49] Cisco System Inc. In *Cisco Visual Networking Index: Forecast and Methodology, 2013-2017*, 2013.

[50] Cisco System Inc. Cisco visual networking index: Global mobile data traffic forecast update 2013-2018. Dec. 2014.

[51] International Telecommunication Union-Telecommunication (ITU-T). Subjective video quality assessment methods for multimedia applications. In *Recommendation P.910*, 2008.

[52] International Telecommunication Union-Telecommunication (ITU-T). Methodology for the subjective assessment of the quality of television pictures. In *ITU-R Recommendation BT.500-12*, sep. 2009.

[53] M.J. Islam, Q.M.J. Wu, M. Ahmadi, and M.A. Sid-Ahmed. Investigating the performance of naive- bayes classifiers and k- nearest neighbor classifiers. In *Convergence Information Technology, 2007. International Conference on*, pages 1541–1546, Nov 2007.

[54] R. Jain. The art of computer systems performance analysis: techniques for experimental design, measurement, simulation and modeling. *New York, John Wiley & Sons*, 1991.

[55] Lucjan Janowski and Piotr Romaniak. QoE as a function of frame rate and resolution changes. In *Proceedings of the Third international conference on Future Multimedia Networking*, FMN'10, pages 34–45, 2010.

[56] Michael Jarschel, Daniel Schlosser, Sven Scheuring, and Tobias Hoßfeld. An evaluation of qoe in cloud gaming based on subjective tests. In *Proceedings of the 2011 Fifth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, IMIS '11, pages 330–335, Washington, DC, USA, 2011. IEEE Computer Society.

[57] S.C. Jha, A.T. Koç, and R. Vannithamby. Optimization of discontinuous reception (drx) for mobile internet applications over LTE. In *IEEE Vehicular Technology Conference (VTC Fall)*, Sept. 2012.

[58] Junchen Jiang, V. Sekar, and Hui Zhang. Improving fairness, efficiency, and stability in http-based adaptive video streaming with festive. *IEEE/ACM Transactions on Networking*, 22:326–340, Feb 2014.

[59] Sunggeun Jin and D. Qiao. Numerical analysis of the power saving in 3GPP LTE Advanced wireless networks. *IEEE Transactions on Vehicular Technology*, 61(4):1779–1785, 2012.

[60] A. Jurgelionis, J. Laulajainen, M. Hirvonen, and A.I. Wang. An empirical study of NetEm Network Emulation functionalities. In *Proceedings of 20th International Conference on Computer Communications and Networks (ICCCN)*, pages 1–6, 2011.

[61] Hyun Jong Kim, Dong Hyeon Lee, Jong Min Lee, Kyoung Hee Lee, Won Lyu, and Seong Gon Choi. The QoE evaluation method through the QoS-QoE correlation model. In *Fourth International Conference on Networked Computing and Advanced Information Management, (NCM '08).*, volume 2, pages 719 –725, Sept. 2008.

[62] Hyun Jong Kim, Dong Geun Yun, Hwa-Suk Kim, Kee Seong Cho, and Seong Gon Choi. QoE assessment model for video streaming service using QoS parameters in

wired-wireless network. In *14th International Conference on Advanced Communication Technology (ICACT)*, pages 459 –464, Feb. 2012.

[63] Charles Krasic, Jonathan Walpole, and Wu-chi Feng. Quality-adaptive media streaming by priority drop. In *Proceedings of the 13th International Workshop on Network and Operating Systems Support for Digital Audio and Video*, NOSSDAV '03, pages 112–121. ACM, 2003.

[64] D.K. Krishnappa, S. Khemmarat, and M. Zink. Planet youtube: Global, measurement-based performance analysis of viewer;'s experience watching user generated videos. In *IEEE 36th Conference on Local Computer Networks (LCN)*, pages 948–956, 2011.

[65] Robert Kuschnig, Ingo Kofler, and Hermann Hellwagner. An evaluation of tcp-based rate-control algorithms for adaptive internet streaming of h.264/svc. In *Proceedings of the First Annual ACM SIGMM Conference on Multimedia Systems*, MMSys '10, pages 157–168. ACM, 2010.

[66] L. S. Lam, J.Y.B. Lee, S.C. Liew, and W. Wang. A transparent rate adaptation algorithm for streaming video over the internet. In *18th International Conference on Advanced Information Networking and Applications (AINA)*, pages 346–351 Vol.1, 2004.

[67] 3GPP; Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures. In *Technical Specification, TS 36.213 V9.3.0*, 2010.

[68] Stefan Lederer, Christopher Müller, and Christian Timmerer. Dynamic adaptive streaming over http dataset. In *Proceedings of the 3rd Multimedia Systems Conference MMSys '12*, pages 89–94, New York, USA, 2012. ACM.

[69] Zhi Li, Xiaoqing Zhu, J. Gahm, Rong Pan, Hao Hu, AC. Begen, and D. Oran. Probe and adapt: Rate adaptation for http video streaming at scale. *IEEE Journal on Selected Areas in Communications*, 32(4):719–733, April 2014.

[70] Yan Lin and Guangxin Yue. Channel-adapted and buffer-aware packet scheduling in LTE wireless communication system. In *4th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM)*, pages 1 – 4, Oct. 2008.

[71] Chenghao Liu, Imed Bouazizi, and Moncef Gabbouj. Rate adaptation for adaptive HTTP streaming. In *Proceedings of the Second Annual ACM Conference on Multimedia Systems*, MMSys '11, pages 169–174, 2011.

[72] Zhaoming Lu, Yan Yang, Xiangming Wen, Ying Ju, and Wei Zheng. A cross-layer resource allocation scheme for ICIC in LTE-Advanced. *Journal of Network and Computer Applications, Elsevier*, 34(6):1861 – 1868, 2011.

[73] John D. McCarthy, M. Angela Sasse, and Dimitrios Miras. Sharp or smooth?: comparing the effects of quantization vs. frame rate for streamed video. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '04, pages 535–542, 2004.

[74] V. Menkovski, G. Exarchakos, and A. Liotta. Machine learning approach for quality of experience aware networks. In *Intelligent Networking and Collaborative Systems (INCOS), 2nd International Conference on*, pages 461–466, Nov 2010.

[75] Vlado Menkovski, Adetola Oredope, Antonio Liotta, and Antonio Cuadra Sánchez. Predicting quality of experience in multimedia streaming. In *Proceedings of the 7th International Conference on Advances in Mobile Computing and Multimedia*, MoMM '09, pages 52–59. ACM, 2009.

[76] K. Miller, E. Quacchio, G. Gennari, and A. Wolisz. Adaptation algorithm for adaptive streaming over http. In *19th International Packet Video Workshop (PV)*, pages 173–178, May 2012.

[77] International Telecommunication Union (ITU-T); The E model: a computational model for use in transmission planning. In *Recommendation G.107*, Dec. 2011.

[78] Ricky K. P. Mok, Xiapu Luo, Edmond W. W. Chan, and Rocky K. C. Chang. Qdash: A QoE-aware DASH system. In *Proceedings of the 3rd Multimedia Systems Conference*, MMSys '12, pages 11–22, 2012.

[79] R.K.P. Mok, E.W.W. Chan, and R.K.C. Chang. Measuring the quality of experience of HTTP video streaming. In *IFIP/IEEE International Symposium on Integrated Network Management (IM)*, pages 485 –492, May 2011.

[80] Microsoft Smooth Streaming (MSS). Iss smooth streaming transport protocol. In *Available [Online] "http://www.iis.net/learn/media/smooth-streaming"*, Nov. 2013.

[81] M.S. Mushtaq, A. Shahid, and S. Fowler. QoS-Aware LTE downlink scheduler for VoIP with power saving. In *IEEE 15th International Conference on Computational Science and Engineering (CSE)*, Dec. 2012.

[82] NetEM. Linux network emulation tool. In *Available [Online] "http://www.linuxfoundation.org/collaborate/workgroups/networking/netem"*.

[83] Netflix. In *Available [Online] "http://www.netflix.com"*, Jan. 2015.

[84] Move Networks. Hd adaptive video streaming. In *[Online] "http://www.movenetworkshd.com"*, Jan. 2015.

[85] International Telecommunication Union-Telecommunication (ITU-T); Amendment 1: Defination of Quality of Experience. In *Recommendation P.10/G.100*, Jan. 2007.

[86] ETSI; Quality of Service (QoS) measurement methodologies. Annex e, method for determining an equipment impairment factor using passive monitoring. In *TIPHONE TS101 329-5*, 2002.

[87] European Network on Quality of Experience in Multimedia Systems and Services (Qualinet). White paper on definitions of quality of experience *qoe* and related concepts. Mar. 2013.

[88] M. Pal and P.M. Mather. A comparison of decision tree and backpropagation neural network classifiers for land use classification. In *Geoscience and Remote Sensing*

Symposium, 2002. IGARSS '02. 2002 IEEE International, volume 1, pages 503–505 vol.1, 2002.

[89] Martin Prangl, I. Kofler, and H. Hellwagner. Towards qos improvements of tcp-based media delivery. In *Fourth International Conference on Networking and Services ICNS.*, pages 188–193, March 2008.

[90] 3GPP; Universal Mobile Telecommunications System (UMTS); User Equipment (UE) procedures in idle mode and procedures for cell reselection in connected mode. In *Technical Specification, TS 25.304 version 5.9.0 Release 5*, 2005.

[91] 3GPP; Medium Access Control(MAC) protocol specification. In *Technical Specification, TS 36.321 version 10.2.0 Release 10*, Mar. 2011.

[92] H.A.M. Ramli, R. Basukala, K. Sandrasegaran, and R. Patachaianand. Performance of well known packet scheduling algorithms in the downlink 3GPP LTE system. In *9th IEEE Malaysia International Conference on Communications (MICC)*, pages 815 – 820, Dec. 2009.

[93] 3GPP; Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC). In *Technical Specification, TS 36.331 version 11.0.0 Release 11*, 2012.

[94] S. Sengupta, M. Chatterjee, S. Ganguly, and R. Izmailov. Improving R-Score of VoIP Streams over WiMax. In *IEEE International Conference on Communications, (ICC '06)*, volume 2, 2006.

[95] M. Shirazipour, G. Charlot, G. Lefebvre, S. Krishnan, and S. Pierre. Conex based QoE feedback to enhance QoS. In *ACM workshop on Capacity sharing,*, CSWS '12, pages 27–32, Dec. 2012.

[96] I. Sodagar. The mpeg-dash standard for multimedia streaming over the internet. *MultiMedia, IEEE*, 18(4):62–67, April 2011.

[97] ISO/IEC 23009-1:2012 Information technology. In *Dynamic adaptive streaming over HTTP (DASH) – Part 1: Media presentation description and segment formats*, 2012.

[98] Jaime Teevan, Susan T. Dumais, and Eric Horvitz. Personalizing search via automated analysis of interests and activities. In *ACM Proceedings of the 28th annual international conference on Research and development in information retrieval*, SIGIR '05, pages 449–456, 2005.

[99] Truong Cong Thang, H.T. Le, H.X. Nguyen, A.T. Pham, Jung Won Kang, and Yong Man Ro. Adaptive video streaming over http with dynamic resource estimation. *Journal of Communications and Networks*, 15(6):635–644, Dec 2013.

[100] Guibin Tian and Yong Liu. Towards agile and smooth video adaptation in dynamic http streaming. In *Proceedings of the 8th International Conference on Emerging Networking Experiments and Technologies CoNEXT '12*, pages 109–120, New York, NY, USA, 2012. ACM.

[101] Hai Anh Tran, S. Hoceini, A. Mellouk, J. Perez, and S. Zeadally. Qoe-based server selection for content distribution networks. *IEEE Transactions on Computers*, 63(11):2803–2815, Nov 2014.

[102] Thu-Huong Truong, Tai-Hung Nguyen, and Huu-Thanh Nguyen. On relationship between quality of experience and quality of service metrics for ims-based iptv networks. In *IEEE International Conference on Computing and Communication Technologies, Research, Innovation, and Vision for the Future (RIVF)*, pages 1–6, Feb 2012.

[103] S. Uemura, N. Fukumoto, H. Yamada, and H. Nakamura. QoS/QoE measurement system implemented on cellular phone for NGN. In *5th IEEE Consumer Communications and Networking Conference (CCNC)*, pages 117 –121, Jan. 2008.

[104] M. Venkataraman and M. Chatterjee. Inferring video QoE in real time. *IEEE Network*, 25(1):4 –13, january-february 2011.

[105] 5G A Technology Vision. Huawei white paper, access on 18-dec-2014. In *Online: (http://www.huawei.com/5gwhitepaper/)*, 2014.

[106] Bing Wang, Jim Kurose, Prashant Shenoy, and Don Towsley. Multimedia streaming via tcp: An analytic performance study. *ACM Trans. Multimedia Comput. Commun. Appl.*, 4(2):16:1–16:22, May 2008.

[107] Shun-Ren Yang and Yi-Bing Lin. Modeling UMTS discontinuous reception mechanism. *IEEE Transactions on Wireless Communications*, 4(1):312 – 319, Jan. 2005.

[108] Gexiang Zhang, Weidong Jin, and Laizhao Hu. Radar emitter signal recognition based on support vector machines. In *Control, Automation, Robotics and Vision Conference, 2004. ICARCV 2004 8th*, volume 2, pages 826–831 Vol. 2, Dec 2004.

[109] Bing Zhou, Jingyuan Wang, Zixuan Zou, and Jiangtao Wen. Bandwidth estimation and rate adaptation in HTTP streaming. In *International Conference on Computing, Networking and Communications (ICNC)*, pages 734–738, Jan 2012.

[110] Chao Zhou, Chia-Wen Lin, Xinggong Zhang, and Zongming Guo. Buffer-based smooth rate adaptation for dynamic http streaming. In *Signal and Information Processing Association Annual Summit and Conference (APSIPA), Asia-Pacific*, pages 1–9, Oct 2013.

[111] Kaijie Zhou, Navid Nikaein, and Thrasyvoulos Spyropoulos. LTE/LTE-A discontinuous reception modeling for machine type communications. *IEEE Wireless Communications Letters*, 2:102–105, 2013.

[112] Lei Zhou, Haibo Xu, Hui Tian, Youjun Gao, Lei Du, and Lan Chen. Performance analysis of power saving mechanism with adjustable DRX Cycles in 3GPP LTE. In *IEEE 68th, Vehicular Technology Conference, (VTC Fall)*, pages 1–5, Sept. 2008.

[113] Xing Zhou, T. Dreibholz, and E.P. Rathgeb. A new server selection strategy for reliable server pooling in widely distributed environments. In *Second International Conference on the Digital Society*, pages 171–177, Feb 2008.

[114] T. Zinner, O. Hohlfeld, O. Abboud, and T. Hossfeld. Impact of frame rate and resolution on objective qoe metrics. In *Second International Workshop on Quality of Multimedia Experience (QoMEX)*, June 2010.