Université du Québec

Institut national de la recherche scientifique

Énergie, Matériaux et Télécommunications

Contributions to QoE management of live IP media

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy

By

Ahmad Vakili

Evaluation Jury

External Examiner	Prof. S. Valaee, University of Toronto
External Examiner	Dr. J.-D. Decotignie, CSEM
Internal Examiner	Prof. T. Falk
Research Director	Prof. Jean-Charles Grégoire
	INRS–Énergie, Matériaux et Télécommunications

“If we knew what it was we were doing, it would not be called research, would it?”

Albert Einstein

Abstract

Increased public access to broadband networks has led to a fast-growing demand for Voice and Video over IP (VVoIP) applications such as Internet telephony (VoIP), video conferencing, and IP television (IPTV). While the evaluation of Video (or Speech) communication systems has been an important field for both academia and industry for decades, the introduction of VVoIP systems has created a new set of issues that require new evaluation methods. Moreover, since we have moved to a unique network for multiple services, it has appeared that traditional QoS measures do not tell a sufficient story and the focus has moved to end user's perceptual quality. Perceived video quality is affected by distortion caused by the encoder and the network impairments. The effect of these distortions depends on different video specifications such as video content, video resolutions, etc., but these parameters have not been widely used in existing perceptual video quality prediction and management models.

The main goal of our research is the development of a control algorithm for perceptual video quality for online video streaming applications. This aim has led to five main contributions as follows:

- Investigation and improvement of packet loss probability estimation methods;
- Investigation and improvement of One Way Delay (OWD) estimation methods;
- Investigation of effects of packet/frame loss on multimedia/video quality according to codec characteristics;

- Accurate (subjective) measurement of the perceived quality of videos with different codec characteristics (Quantization parameter and frame-rate).

- Investigation of different video bit rate-controlling mechanisms/models to manage the end user's perceived quality and development of an algorithm to control the perceptual video quality for video conferencing applications;

Acknowledgments

It is a pleasure to thank the many people who made this thesis possible.

First and foremost, I would like to express my deepest gratitude to my advisor Professor Jean-Charles Grégoire. With his enthusiasm, his inspiration, and his great efforts to explain things clearly and simply, he made the research process more interesting to me. Throughout my PhD period, he provided encouragement, sound advice, good teaching, good company, and lots of fantastic ideas, but he always gave me great freedom to pursue independent work.

I am indebted to my many colleagues for providing a stimulating and fun environment in which I could learn and grow. I am especially grateful to Dr. Keyvan Zarifi, Dr. Vahid Asghari, and Mr. Marcelo Santos.

I am also grateful to the administration staff, especially Madam Hèlén Sabourin, for assisting me in many different ways.

I wish to thank my entire extended family for all their love and encouragement.

Lastly, and most importantly, I wish to thank my lovely wife, Sara, to whom I owe my life, my love, and my soul. This dissertation is dedicated to her who walked this road with me every step of the way. I love you with all my heart and hope that our life together will be filled with love and prosperity.

Ahmad Vakili

September 2012

Contents

List of Figures

List of Tables

Liste des Figures

Liste des Tables

List of Acronyms

ACELP	Algebraic Code-Excited Linear Prediction
ACR	Absolute Category Rating
ADSL	Asymmetric Digital Subscriber Line
AIMD	Additive Increase and Multiplicative Decrease
API	Application Programming Interface
ATM	Asynchronous Transfer Mode
AVC	Advanced Video Coding
AVI	Audio Video Interleave
CCI	Call Clarity Index
CELP	Code-Excited Linear Prediction
C.I	Confidence Interval
CIF	Common Intermediate Format
CLT	Central Limit Theorem
DCCP	Datagram Congestion Control Protocol
DCR	Degradation Category Rating
DMOS	Degradation Mean Opinion Score
DSCQS	Double Stimulus Continuous Quality Scale
DSCS	Double Stimulus Comparison Scale
DSIS	Double Stimulus Impairment Scale
DVC	Distributed Video Codin
EM	ElectroMagnetic waves
FEC	Forward Error Correction
FEPI	Frame Error Propagation Index
FIFO	First In First Out
FR	Full Reference

GAP	Good, Acceptable or Poor
GOP	Group Of Pictures
GPS	Global Positioning System
GUI	Graphical User Interface
HMM	Hidden Markov Model
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
IPTV	IP Television
ISI	International Standardized Index
ITU	International Telecommunication Union
JND	Just Noticeable Difference
LAN	Local Area Network
LDT	Large Deviation Theory
LSE	Least Square Error
MAD	Mean of Absolute Difference
ME	Mouth-to-Ear
MMRP	Markov Modulated Rate Processes
MNRU	Modulated Noise Reference Unit
MOS	Mean Opinion Score
MPEG	Moving Picture Experts Group
MP-MLQ	Multi-Pulse Maximum Likelihood Quantization
MSE	Mean Squared Error
NGN	Next Generation Network
NR	No Reference
NTP	Network Time Protocol
OWD	One Way Delay

PAMS	Perceptual Analysis Measurement System
PC	Pair Comparison
PCM	Pulse Code Modulation
PER	Packet Error Rate
PESQ	Perceptual Evaluation Speech Quality
PLC	Packet Loss Concealment
PLP	Packet Loss Probability
PLR	Packet Loss Ratio
PTT	Post, Telegraph, and Telephone
PQR	Picture Quality Rating
PSNR	Peak Signal to Noise Ratio
PSQM	Perceptual Speech Quality Measure
PSTN	Public Switched Telephone Networks
QCIF	Quarter Common Intermediate Format
QoE	Quality of Experience
QoS	Quality of Service
QP	Quantization Parameter
RAP	Rate Adaptation Protocol
RTP	Real-Time Transport Protocol
RTCP	Real-time Transport Control Protocol
RR	Reduced Reference
RTT	Round Trip Time
SAD	Sum of Absolute Difference
SAT	Service Attribute Test
SLA	Service Level Agreement
SSCQE	Single Stimulus Continuous Quality Evaluation

SSIM	Structural Similarity Index Metric
STP	Set-Top Box
SVC	Scalable Video Coding
SVM	Support Vector Machine
TCP	Transmission Control Protocol
TCPF	TCP Friendly
TEAR	TCP Emulation At Receiver
TFRC	TCP Friendly Rate Control
TFWC	TCP-Friendly Window-based Congestion Control
UDP	User Datagram Protocol
UTC	Universal Time Coordinate
VGA	Video Graphics Array
VoIP	Voice over IP
VQEG	Video Quality Expert Group
VTP	Video Transport Protocol
VVoTP	Voice and Video over IP
VQM	Video Quality Models
WAN	Wide Area Network
XML	Extensible Markup Language

Chapter 1

Introduction

1.1 Overview

In telecommunications, performance is assessed in terms of quality of service (QoS). It is measured either in terms of technology (e.g., for ATM, cell loss, variation, etc.) [1] or at some protocol level (e.g., packet loss, delay, jitter, etc.) [2]. In the days of application-dedicated networks or when inter-networking was the service, such measures were sufficient to characterize quality and the potential negative impact on the service, or alternatively, they were useful as parameters for Service-Level Agreements (SLA) between service providers and users. Today, increased access to broadband networks has led to a fast-growing demand for Voice and Video over IP (VVoIP) applications such as Internet telephony (VoIP), video conferencing, and IP television (IPTV). While the evaluation of Video (or Speech) communication systems has been an important field for both academia and industry for decades, the introduction of VVoIP systems has created a new set of issues that require new evaluation methods. Moreover, since we have moved to a unique network for multiple services, it has appeared that traditional QoS measures do not tell a sufficient story and the focus has moved to Quality of Experience (QoE). QoE is the overall performance of a system from the point of view of the users. In other words, QoE is a measure of end-to-end performance at the service level from the user perspective and an indication of how well the system meets the user's needs [3]. When users talk about quality, they are trying to describe their reaction to, or satisfaction with one or several of these service attributes, according to the nature of the application:

- Connection quality;
- Connection usability;
- Connection security;
- Connection or disconnection reliability;
- ... .

Therefore, although QoE is quite subjective in nature, it is very important that a strategy be devised to measure it as realistically as possible. The ability to measure QoE will give the service provider some sense of the contribution of the network's performance to the overall level of customer satisfaction in terms of reliability, availability, scalability, speed, accuracy and efficiency. As a consequence, even if a service infrastructure has been properly engineered, we must measure QoE delivered, which is fraught with many practical challenges. Information coding is often a quality reducing process per-se, and hence the quality transmitted to the user is not optimal, independently of any transmission mishap. What is lost, corrupted or otherwise delayed also has an impact on quality, as not all parts of the information are considered equal. For unidirectional real time transfer (e.g., video streaming), information loss and coding are the dominant factors, but for bidirectional online communications (e.g., conversation over the internet or video conference) other parameters such as delay and jitter (variation in delay) can be as important as coding and loss [3].

The exponential growth of multimedia applications over the Internet and the importance of the QoE to measure the performance of multimedia services motivated the set up of this research. In this chapter the research questions motivating this project are presented, followed by the research objectives and its main contributions. The rest of this chapter is organized as follows: the motivations of this research are presented in Section 1.2. Section 1.3 introduces the research aims and objectives. The main contributions of this research are summarized in Section 1.4. Section 1.5 briefly describes the outline of the thesis.

1.2 Motivation

QoE is not a new concept by any means. It has long been established in telephony where it was important to measure user satisfaction with the service, and this was done with subjective experiences with a large number of users, and set in terms of a Mean Opinion Score (MOS) of quality, from poor to excellent. However, subjective quality measurement techniques cannot be used in large-scale experiments due to their large overhead, the high cost of listening experts, and their unrepeatable nature [4, 5]. Furthermore, for pro-active troubleshooting of VVoIP performance bottlenecks that manifest themselves as performance impairments such as video frame freezing and voice dropouts, network operators cannot rely on actual end-users to report their subjective perceptual quality. Hence, automated and objective techniques that provide real-time or online VVoIP perceptual quality estimates are vital [5].

QoE combines non-technical parameters such as user perception, experience and expectation with technical parameters such as application- and network-level QoS. From the user's point of view, the QoE-technical part for multimedia transmission over the Internet can be summarized in a QoE value chain which comprises the following [6]:

- Multimedia content providers, multimedia servers, streaming applications, multimedia preparation, etc.

- Network and service providers, network impairments, etc.

- User devices, playback applications, etc.

In order to manage the user-perceived quality, it is vital to understand the quantitative relationship between QoE and all these technical parameters in the QoE value chain.

The relationship between QoS (application and network) and QoE helps network and service providers to manage QoS parameters and service provisioning efficiently and effectively in order to provide a better QoE to users in a cost-effective, competitive, and efficient manner. The first step in this process is to measure the end user's satisfaction level of the service quality, while the application and network QoS parameters are being monitored and measured. Contemporary perceived quality measurement techniques are divided into subjective and objective measurements [4]. Subjective evaluation techniques using human users to rate the video, audio, or data quality can provide the most accurate assessment of output quality from the perspective of a service provider's customers. However, due to their disadvantages already mentioned at the outset of this Section, objective tests for predicting end user perception should be preferred, as more practical [4, 5, 7]. Typically, objective assessment of perceptual quality requires comparison of source and destination information [2, 3, 8]. Predicting quality would require knowing the information source as well as the effects which network propagation may have on the data. For example, the majority of models and systems which exist for estimating video quality in packet networks typically require detailed knowledge of video content and features, and often rely on deep inspection of video packets [7, 9]. These techniques can be called offline techniques because (a) they require time and spatial alignment of the original and reconstructed information, which is time consuming to perform, and (b) (for video transmission) they are computationally intensive due to their per-pixel processing of the video sequences. Another set of objective tests, called Indirectly Objective Tests, use measurements of network impairments (loss, delay, jitter, duration of the defect) to estimate the impact on quality (video or audio) and could be performed online. These techniques can be applied where there is an established relationship between QoE and QoS [3].

All in all, we can conclude that QoS and QoE are two interdependent concepts in the modern multimedia transmissions over IP networks and hence, they should be studied and managed with a common understanding, from planning to implementation and engineering (optimization). Moreover, although the QoS research field has been extensively studied, measuring network impairments for enhancing the QoE is nevertheless an open research area. For instance, in our observations, earlier research on measuring and modelling the packet loss would generally either increase the burden of probe packets' bit rate to the available bandwidth [10–12] or not provide real time information [13–15]. Although these studies are undoubtedly useful to understand the general loss characteristics, they cannot be used in real time performance estimation and consequently online control systems. To cope with the shortcomings of the aforementioned methods, many researchers have tried to link the input process to the probability of loss at intermediate nodes [16–22]. However their proposed methods have their own disadvantages such as computational complexity and inaccuracy. In the case of delay, halving the Round Trip Time (RTT) is the most common and simplest method to estimate the OWD. However, in the Internet, sending and receiving paths between two end users which are usually far from each other are most often not symmetric. Moreover, most popular access technologies are intrinsically asymmetric [23–25]. Therefore, deriving the OWD from the RTT cannot lead to an accurate measurement. Synchronization of two end nodes which are connected to the Internet network independently is another method for measuring the OWD by reading the time-stamp section in IP/RTP/UDP packet. Network Time Protocol (NTP) [26], Global Position System [27], and the IEEE 1588 standard [28] are among synchronization techniques used in special cases [29]. All of these solutions are not fully applicable or ubiquitous in Internet networks [30, 31], or do not resolve the issue of asymmetry. Therefore, the investigation of how to increase the accuracy of online measurement or estimation of QoS parameters is under consideration in this thesis.

The effects of QoS parameters on QoE have been studied from different perspectives which are reviewed in Chapter 2. For instance, the effects of bit/packet/frame loss and the length of loss on the video quality have been discussed in [32–37] and various loss-quality models have been proposed. However, existing models have not taken into account all aspects of loss effects on perceived video quality. For example, the results of those which model the impact of error propagation due to a frame loss on perceived quality of transmitted compressed video do not have an acceptable correlation with experimental results for all content dynamics. Our research investigates this issue and specifically focuses on the effect of frame loss position on videos with different content dynamics.

QoE measurement techniques may have multiple use. They can be used in network design or in choosing suitable (e.g., voice) codecs to try to meet minimal quality requirements. Similarly in case of streaming video (e.g., TV) service, QoE measurement techniques can help to determine how well the network supports delivering a specific level of quality. They can also be used for monitoring purposes, either in the context of a Service-Level Agreements (SLA) or simply for quality assessment. Finally, they can be exploited in an adaptive way to protect quality for a service.

Video streaming is currently commonly employed over the Internet, and it is also expected that video chatting will be one of the key business areas for mobile services through wireless communications (e.g., 3G and 4G). To meet customer expectations, service providers should know the level of quality which is deemed acceptable by customers. Based on this information, service providers need to manage and control resources efficiently. However, managing and deploying more resources not only increases costs but also sometimes is not possible (e.g., in mobile environment, the bandwidth cannot be more than a certain level). Therefore, it seems that designing intelligent applications, which can dynamically adapt themselves with existing networks by managing the video system (e.g., bit rate) without adverse effect on end-users' perceived quality, has become an overwhelmingly important issue. In other words, QoE management by video applications is meant to lead to more efficient and economic deployment of network resources while keeping the end user's satisfaction at an acceptable level. Moreover, since the close cooperation of servers/applications with service providers is not usually feasible, there is a need to develop new multimedia applications which can adapt themselves to the existing network environment and the best effort and competitive nature of the Internet to deliver the best perceptual quality to the end-users. To make a contribution to this subject, we have investigated how video parameters affect the video bit rate and video quality. Contrary to previous studies [38–42], ours specifically addresses the video streaming applications which transmit QCIF-, CIF-, and VGA-size and medium motion videos coded with H.264 (the most pervasively used video type in video conferencing applications) over the limited bandwidth networks and the perceptual quality is assessed through subjective tests.

1.3 Goals

The preexisting service quality measurement and control methods in IP networks do not reflect users' service satisfaction. Thus, to enhance end-user's perception of quality, we plan to design a system of interworking control between experienced quality, transmission QoS parameters, and application layer at server and client (e.g., video coding specifications). This control system is utilized in video streaming application to optimize the perceived video quality according to the network conditions and manage utilization of the available resources. Due to the scale of project, we consider only the scenarios where the QoS and QoE-monitored communications occur between a client and a server.

As the first step, we seek to investigate what factors affect multimedia quality perceived by end users, how these factors are measured or predicted, and how different explicit or implicit modes of information exchange (e.g., the RT(C)P protocol) can be used to detect variations in the perceptual quality. In other words, we want to define the characteristics which should be monitored and measured, and then, we plan to find the correlation between the measured characteristics and users' perception to figure out the role of each piece of information (network's behaviour and characteristics) in experienced quality. In particular, our focus is to monitor and measure the packet loss and One Way Delay (OWD) as the QoS parameters which affect the QoE.

Due to the broad domain of QoS variations, tracking QoE fluctuations based on QoS parameters would be impossible or very complicated. Therefore, our objective is to explore the relationship between QoE and some specific QoS parameters (i.e., loss and bandwidth). Inasmuch as employing the Internet as a transmission infrastructure for videos is major research focus in industry and academia alike and plays an important role in Next Generation Networks (NGN), we concentrate first on determination of perceptual quality as a function of loss pattern for video services such as Video Conference services. Hence, we intend to investigate how the loss pattern affects video quality and more explicitly, if the position of packet/frame loss is important. By answering this question, we propose a video quality model based on frame loss position for different video types. Based on our investigation on the effect of different loss patterns on video quality, we intend to figure out how it is possible for the end users' applications to minimize the quality degradation.

Since the video bit rate varies because of different video characteristics such as frame rate, resolution, compression level, content, etc., a similar network situation may cause end users to perceive a different level of quality for different videos. To investigate the effect of these video characteristics on the end-users' perceived quality some studies have been conducted recently. ITU-T Recommendation G.1070 has modelled the perceived video quality as a function of bit rate, frame rate, and packet loss [43]. Tao Liu et al. in [44] have also investigated the effect of bit rate, frame rate, and packet loss on perceived video quality and have extended the perceptual quality estimation method, introduced by ITU-T Rec. G.1070, to a real-time video quality monitoring. Thomas Zinner et al. in [38] have conducted a measurement study and quantified the effect of 1) video frame rate, 2) scaling method, 3) video resolution, and 4) video content types on the perceived quality by means of the Structural Similarity Index Metric (SSIM) and Video Quality Metric (VQM) full-reference metrics. Objective tests have been used in their study to determine the level of perceptual quality. Furthermore, they have focused on high resolution videos. In [39], Y. Pitrey et al. have evaluated the performance of two AVC and SVC standards for coding the video data in different situations by conducting the subjective video quality tests. McCarthy et al. in [40] have compared the importance of frame rate and quantization (i.e., video quality due to data compression) in the case of watching high motion videos such as a football game in CIF and QCIF sizes. Since the medium motion and lower resolution videos (e.g., videos produced by video conferencing applications) are used widely over the Internet and cellular networks, our research focus is on this type of video.

Considering the mentioned studies, this research intends to accurately answer two main questions: "what is the actual perceived video quality when the video parameters are changed to meet the bandwidth limitation?"; and "what are the best video parameters for a specific video bit rate given the subjective perceived quality by the end users?". This thesis focuses on investigating the effect of different factors such as frame rate and quantization (QP) on video data bit rate and perceived video quality and, consequently, on controlling the QoE with video parameters according to the bandwidth limitations imposed by the network. As it was mentioned earlier, the investigation of different bandwidth measurement methods is one of our objectives. We then proceed to relate these measures to actions at the server which can keep their transmitted video's perceptual quality within acceptable boundaries (performance assurance). Indeed, a control system, whose feedback is the network condition (e.g., estimated bandwidth), controls and adjusts the streaming bit rate by employing a combination of codec characteristics control mechanisms.

To conclude, our objectives include:

- QoE definition and investigation of its measurement methods;
- Investigation and improvement of packet loss probability estimation methods;
- Investigation and improvement of One Way Delay (OWD) estimation methods;
- Investigation of effects of packet/frame loss on multimedia/video quality according to codec characteristics;
- Accurate (subjective) measurement of the perceived quality of videos with different codec characteristics (Quantization parameter and frame-rate).
- Investigation of different video bit rate-controlling mechanisms/models to manage the end user's perceived quality;

1.4 Contributions

The prominent contributions of this research are as follows:

- After reviewing the existing packet loss probability (*plp*) online-estimation methods, we have proposed an accurate approximation for *plp* at an intermediate high speed node where a large number of sources are expected to be aggregated. In this method, based on *Large Deviation Theory*, estimation of the *plp* at the intermediate nodes is based on the input stochastic traffic process.

  (The associated publications are [45, 46].)

- Different One Way Delay (OWD) estimation/measurement methods have been investigated. A straightforward method to accurately measure the transmission delay (i.e., a deterministic delay's part) has been introduced. Considering the measured transmission delay, a couple of additional useful constraints to the set of equations and variables employed in the cyclic-path method have been proposed to improve precision in predicting the OWD. It has been shown that all the proposed methods are free from clock skew awkwardness.

(The associated publication is [47].)

- This research addresses the question of whether or not a specific lost packet/frame, and in particular, its position relative to the I-frames, influences the quality of transmitted compressed video. Using the average Peak Signal to Noise Ratio (PSNR) of the received coded video to measure the amount of distortion, we investigate the effect of frame loss position relative to the I-frames on the total distortion for the videos. Based on our empirical results we have proposed a model to estimate the PSNR of the received frames impaired by distortion propagation. Furthermore, after investigating the probability of different burst loss lengths in noisy environments where the duration of data loss is almost constant, we propose a method for improving the performance of video streaming over noisy channels based on packet scheduling, without an increase in the bit rate.

  (The associated publications are [48, 49].)

- To reach our last objective, extensive measurement studies for investigating the effect of different control parameters (i.e., frame rate and QP) on bit rates limited by network bandwidth have been conducted. We have used the results of subjective tests, conducted for measuring the end-users' perceived video quality, to find the optimum video parameters based on the given network bandwidth and acceptable perceptual quality level; and finally, we propose a video perceptual quality control algorithm based on the mentioned measurements.

  (The associated publications are [50, 51].)

1.5 Document outline

This thesis is laid out as follows. Chapter 2 provides a comprehensive definition of QoE and some background information on its measurement methods. Section 2.2 describes the concept of two perceptual quality measurement families: Subjective and Objective. In Section 2.3 network parameters and their effect on multimedia quality perceived by end user are described.

Chapter 3 discusses how to accurately estimate the packet loss probability at a high speed intermediate node in the Internet network in real time. This chapter continues in Section 3.2 by reviewing prior bodies of work on measuring or estimating the packet loss probability. In Section 3.4, we develop a new plp estimator. Section 3.5 and 3.6 present the simulations and their numerical results to demonstrate the effectiveness of our proposed estimator.

Chapter 4 summarizes the state of the art in One Way Delay measurement methods. Section 4.2 reviews prior bodies of work on measuring or estimating the OWD. In Section 4.3, the three-node model, the cyclic-path/LSE method, and our improvements are explained. In Section 4.4, a method for measuring the transmission delay between two end-nodes is introduced. Simulations and numeric results demonstrate the improvement of the proposed model relative to other models in Section 4.5. The level of accuracy of the proposed method to measure the transmission delay is also demonstrated in that section.

Chapter 5 presents a model to estimate the video quality degradation according to the frame loss position. Prior models for estimating the distortion produced by packet loss is reviewed in Section 5.2. In Section 5.3, we describe the effect of the position of lost frames relative to I-frames on the average $PSNR$ and derive a model that estimates the total propagated distortion. The accuracy of model estimations is demonstrated via simulations in Section 5.4. The effect of packet transmission scheduling on noisy channel performance is examined in Section 5.5.

Chapter 6 states the relation of the video coding parameters and perceived video quality. Section 6.2 introduces different congestion control methods in real-time multimedia transmission as well as recent studies regarding the effect of video parameters on the perceived quality. Section 6.3 presents the coding results for different video parameters. The details of subjective video quality measurement tests and their outcomes are presented in Section 6.4 and 6.5. In Section 6.6 our perceptual quality control algorithm is proposed. Simulations and numeric results demonstrate the effectiveness of the proposed algorithm relative to others in Section 6.7.

Finally, we present the conclusions of our work and suggest directions for future research in Chapter 7.

Chapter 2

QoE Definition and its Measurement Methods

Throughout the world, more and more people are choosing the Internet as the infrastructure for communicating with others, conducting their business, listening to a music, or watching video content. Therefore, interest in Quality of Experience (QoE) has spiked over the few last years in industry and academia. In spite of the network used for access, type of device, content listened to or viewed, everybody has some basic expectation about his/her requested service. Depending on different factors such as the type of device being used for connecting to the networks, type of content being requested, amount of money spent for the service, etc., the level of expectation varies among the users. It is the service providers' job to fulfill the users' expectations, hence the large number of research efforts have been undertaken on QoE. Still, QoE remains an elusive notion, intrinsically because of its subjective nature.

The structure of this chapter is as follows: Section 2.1 presents perspectives on QoE from the literature. Section 2.2 describes the concept of two perceptual quality measurement families: Subjective and Objective. In Section 2.3 network parameters and their effect on multimedia quality perceived by end user are described. Section 2.4 concludes the chapter.

2.1 QoE Definition

The most commonly used definition of the QoE is probably the one given by the ITU-T SG12 [52], which defines QoE as:

> *"The overall acceptability of an application or service, as perceived subjectively by the end user."*

Although this definition is not wrong, it is not complete and does not consider all aspect of the QoE. A better definition of the QoE which has been produced in 2009 Dagstuhl Seminar [53] states the QoE as:

> *"The degree of delight of the user of a service, influenced by content, network, device, application, user expectations and goals, and context of use."*

This definition is more general and better than the ITU-T SG 12's one, but it is not comprehensive enough to provide a meaningful insight into all features such as the temporal aspects of the QoE and the utility-related aspects of the experience. Since the most recently research on QoE have focused on multimedia applications and services, most QoE definitions explain the QoE from an *Audio* or *Video* users' point of view. Although the focus of this research is on multimedia and specifically on *Video*, it is worthwhile to introduce an encyclopedic definition of the QoE. A good example of a high-level QoE definition has been stated by Varela [54] as:

> *"The user's subjective assessment, be it qualitative or quantitative, of the degree of fulfillment of his or her expectations with respect to the utility and/or enjoyment derived from the use of a certain service or application, over a given period of time, for well-defined usage intent and context, and considering the user's own psychological and socioe-*

conomic context."

This generic definition of QoE can be used as a template which can then be specialized to specific fields, user profiles, societal contexts, etc.

Due to the large variety of disciplines encompassed by the notion of QoE, it is not possible to cover all its aspects in a single research, nor to reveal it through a unique test. In our research, we have only focused on what an ordinary user *sees* or *hears*; the quality level of a specific video clip or audio in a relatively controlled environment. Therefore, only technical aspects which influence the *experienced quality* are considered and the other non-technical issues surrounding QoE (e.g., socio-economic context of the users) will remain for future study. Thereby in this dissertation, the QoE term will be used interchangeably with *perceptual quality*. Although this perceptual quality cannot properly be defined, it can be measured. The perceptual quality in question is dependent on a few factors such as network QoS parameters and video coding specifications which are quite well understood.

For pro-active troubleshooting of multimedia application performance bottlenecks that manifest to end-users as performance impairments such as video frame freezing and voice dropouts, service providers and network operators first monitor and measure the quality of service that the end-users encounter [5]. In other words, to indicate how well the system meets the user's needs we should measure the end-to-end performance at the service level from the user's perspective [3]. To measure the perceived quality of multimedia different methods which are discussed in the rest of this chapter, are proposed by researchers.

2.2 Multimedia Quality Measurement

Multimedia quality can be measured or estimated either subjectively or objectively. Subjective quality measurement methods, as implied by their name, are carried out by human subjects to assess the overall perceived media quality. They are the most reliable method of measuring the quality. However, the subjective methods are time and labor consuming and expensive. Thus, objective assessment methods which produce the results comparable with those of subjective methods, are needed. In objective tests, assessing the perceptual quality can be made by *intrusive* or *non-intrusive* measurement. The source data is required by intrusive measurements, whereas non-intrusive methods do not need to access the video or audio original data.

2.2.1 Subjective Perceived Quality Measurement

Subjective multimedia quality measurement tests are basically laboratory experiences in which subjects are exposed to the media samples and asked to rate the samples' quality. There are different types of subjective test depending on what quality aspects is evaluated or what kind of application is assessed. The subjective assessment is classified in two main categories: *qualitative assessment* and *quantitative assessment*.

The qualitative methods attempt to describe the end users' perception of quality and the motivation behind users' decisions made during interaction [55–57]. The results of qualitative methods do not necessarily translate well into numeric scales. The qualitative methods are usually more suited to the *sociologic* aspects, for instance, when the goal is to evaluate the price of a multimedia service or to know how different people react to variation of the perceived quality. On the other hand, quantitative methods focus on the *technical* aspects of quality and they are used and studied more widely in literature than qualitative ones. The main idea behind this type of assessments is to ask a group to rate the media samples according to the specific scale. The Mean Opinion Score (MOS) is the most frequently used score which presents the results of these tests and it summarizes the group's assessment of the perceived quality.

There exist several protocols to perform the subjective assessments of the quality of multimedia. The most frequently used ones are those recommended in ITU standards such as ITU-T Rec. P.910 [58] and ITU-R BT.500-10 [59] for video, ITU-T P.800 [60] for voice, and ITU-T P.920 [61] for interactive multimedia.

Depending on the purpose of the assessment, several procedures are proposed in these standards. In most widely used protocols, the performance of the system under test is rated directly (absolute category rating, ACR) or relative to the subjective quality of a reference system (degradation category rating, DCR) [62].

The following opinion scales, which are the most widely used by ITU-T, can be utilized in an ACR: *excellent*, *good*, *fair*, *poor*, and *bad* or 5, 4, 3, 2, and 1 [2, 62]. The arithmetic mean of all the collected opinion scores (among participants in a test) is the MOS.

To measure the degradation caused by some encoding or transmission scheme, the DCR is more appropriate. In a DCR test, assessors are asked to rate the perceived degradation of the video or voice sample with respect to a high quality reference clip; the subjects are instructed to rate the conditions according to this five-point degradation category scale: degradation is *imperceptable* (5), *perceptible but not annoying* (4), *slightly annoying* (3), *annoying* (2), or *very annoying* (1). The mean value of the results is called the Degradation Mean Opinion Score (DMOS) [63].

Subjective Perceived Voice Quality Assessment

In VoIP the most widely used subjective quality assessment methodology is opinion rating, which is defined in ITU-T Recommendation P.800 [64]. The performance of the system under test is rated directly (ACR) or relative to the subjective quality of a reference system (DCR) [62].

In a DCR test for voice quality, the subjects are asked to rate the conditions according to this five-point degradation category scale: *inaudible* (5), *audible but not annoying* (4), *slightly annoying* (3), *annoying* (2), and *very annoying* (1).

Any subjective test is constructed under a subjective test protocol. One of the most famous protocols is the Service Attribute Test (SAT). This test was developed by Satellite

Business Systems in the early 1980s [2].

The challenge of the evaluation of the MOS is to make it experiment-independent to erase differences in testing date/tester and the mix of quality levels in the experiment. Therefore, ITU-T Recommendation P.810 in 1984, proposed the opinion equivalent-Q method, in which the Modulated Noise Reference Unit (MNRU) is used. MNRU is a reference system that outputs a speech signal and speech-amplitude-correlated noise with a flat spectrum. The ratio of signal to speech correlated noise in dB is called the Q value. This condition helps subjective experiments be reproducible [62].

Note that the above mentioned methods are appropriate for assessing voice quality in one-way speech communications (one speaker, multiple listeners), and not for two-way interactive speech communications.

Kiatawaki et al. in [65], conducted conversational experiments by having two parties and using an adjustable delay speech system. They studied the effect of delay on conversational experience but they did not consider packet loss and variations in delay (jitter).

ITU P.805 describes a subjective test for evaluating MOS_{CQS} score of a conversational quality [66]. This method requires asking two subjects to participate in the conversation over a communication system to complete a specific task. They are asked to rate the quality of conversation using an ACR. The average of opinions of several users (test participants) defines the conversational quality of the system being tested.

Subjective Perceived Video Quality Assessment

For video, as with voice, subjective tests use human viewers to rate the video quality. Subjective tests are done either informally or using formal techniques. Informal assessment of video quality is often done by a service provider craftsperson on site and technical experts (golden eyes) in the video system head end or during commissioning [3]. Formal subjective assessments use many highly qualified experiment participants who view various video clips in tightly controlled environments and rate the quality. Generally TV subjective video picture quality tests are performed following the guidelines established in ITU-R Recommendation 500 (ITU-R, 2002) better known as Rec. 500 [3, 59]. Rec. 500 recommends detailed guidelines for standard viewing conditions, criteria for selection of subjects and video test sequences, assessment procedures and methods for analyzing the collected video quality scores.

The most known and frequently used methods in this standard are:

- ***Double Stimulus Continuous Quality Scale (DSCQS)***: With DSCQS, viewers are shown pairs of video sequences (the impaired sequence and the reference sequence) randomly. Viewers watch each pair twice. After the second watching, viewers are instructed to rate the quality of each sequence in the pair. The difference between two scores is used to quantify changes in quality [59]. The DSCQS is widely accepted as an accurate test method which has a low sensitivity to context effects (see Appendix 3 in [59]).

- ***Single Stimulus Continuous Quality Evaluation (SSCQE)***: The SSCQE allows viewers to dynamically rate the quality of an arbitrarily long video sequence, using a slider mechanism with an associated quality scale (e.g., from *bad* to *excellent* which is equivalent to a numerical scale from 0 to 100). The SSCQE method is more useful for evaluating real-time quality monitoring systems. Proponents of the SSCQE methodology claim that it can be used to assess long video sequences with time-varying quality, whereas DSCQS cannot [59].

- ***Double Stimulus Comparison Scale (DSCS)***: In DSCS, viewers are shown a pair of video sequences. Like the DSCQS method, the pair of video sequences are watched randomly, but unlike DSCQS, the pair is shown once instead of twice. In DSCS, the assessors directly rate the difference between the first and second video sequence on a discrete seven point scale (as opposed to DSCQS where two video sequences are rated separately on a continuous quality scale). The viewers indicate whether the video quality of the second clip was *much better*, *better*, *slightly better*, *the same*, *slightly worse*, *worse*, or *much worse* than the first clip.

All these methods are mainly intended for television signals and are well described in ITU-T Recommendation T.500.11 [59]. The modified versions of these methods for assessing the video clips' quality has been proposed by ITU-T Recommendation P.910 [58] and Video Quality Expert Group (VQEG) [67] which are listed below:

- ***Single stimulus-ACR***: In the ACR method, the video sequences are presented one at a time and are rated by assessors independently on a category scale. The ACR method is also called Single Stimulus Method. Figure 2.1 illustrate the time pattern for the stimulus presentation.The presentation time may vary according to the content of the test material. The voting time is equal or less than 10 s. The five level scale from *Bad* to *Excellent* is used for rating the overall quality in ACR method. A nine or eleven-level scale may be used when the higher discriminative power is required (Annex B in [58]). Hidden reference removal method is provided by VQEG [67] to utilize the ACR method. In this method the reference video is also viewed by subjects who are not aware of watching the original video along with the other test videos. The reference video rating scores are withdrawn from the results of the corresponding test. It helps us to insure the subject's rating accuracy.

- ***Double Stimulus Impairment Scale (DSIS)-DCR***: In the DCR method, the video sequences are presented in pairs; the reference video is always presented as the first stimulus in each pair, while the second stimulus is the degraded video. The DCR method is also called double stimulus impairment scale method. The time pattern for the stimulus presentation and voting is illustrated in Fig. 2.2. The five level

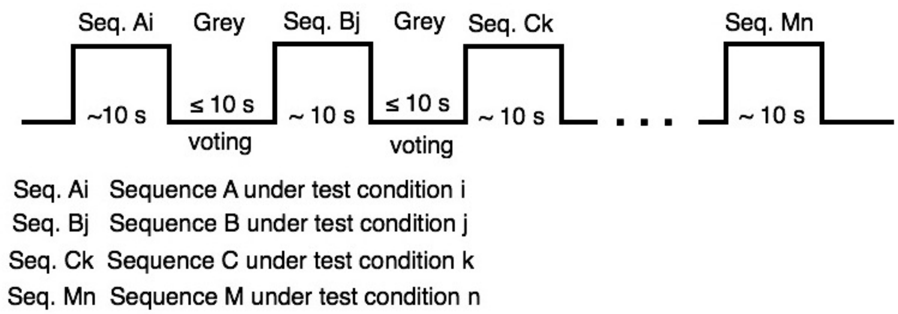


Figure 2.1: Stimulus presentation and voting time in the ACR method.

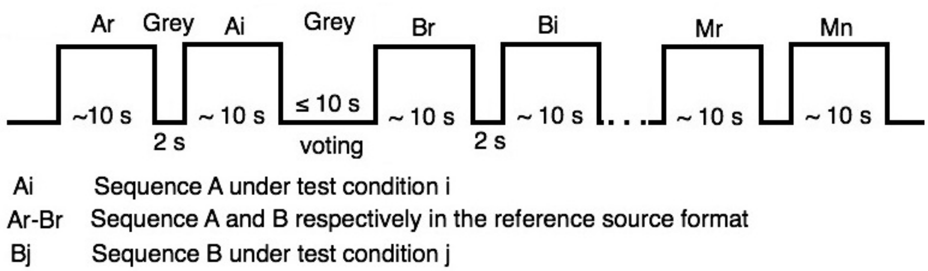


Figure 2.2: Stimulus presentation and voting time in the DCR method.

scale from *Imperceptible* to *Very annoying* is used for rating the degradation in DCR method.

- ***Pair Comparison method (PC)***: In the PC method, the video sequences are presented in pairs; the first and second stimulus are the same sequence which is passed through two different systems under test. Viewers are asked to make a judgment on which element in a pair is preferred in the context of the test scenario. The time pattern for the stimulus presentation and voting in PC method is illustrated in Fig. 2.3.

To insure the subject's rating accuracy, the subjective tests designed for our research were based on Single Stimulus Hidden Reference Removal which is described in detail in 6.4.

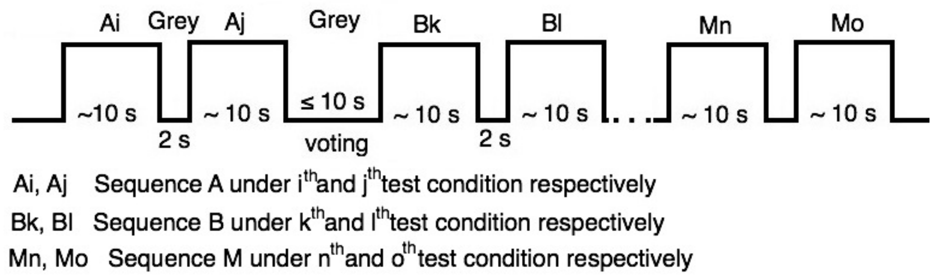


Figure 2.3: Stimulus presentation and voting time in the PC method.

2.2.2 Objective Perceived Quality Measurement

Subjective tests have been used for a long time, yet they are still quite useful means of accurately gauging likely user perception of service quality. However, because of the necessary involvement of a relatively large number of test participants and the extensive sampling requirements, such tests tend to be labour-intensive and relatively expensive. Consequently, one of the major goals for developing test and evaluation methodologies for voice or video quality has been to achieve assessments of similar quality by means of analysis of data that can be acquired quickly and inexpensively via objective measurements.

Objective Perceived Video Quality Assessment

Picture quality assessment from the end-user point of view depends on many factors such as size and resolution, sharpness, contrast, color saturation, viewing distance, user environment, naturalness, and distortion.

We can classify objective video quality measurement in four categories [3]:

- based on models of human video perception
- based on video signal parameters
- based on network impairment parameters

- based on the duration of the network impairment in the video signal

The Picture Quality Rating (PQR) method is based on human video perception [68]. The PQR value for a video field is a nonstandard video quality metric based on the JND-metrix human-vision algorithm [69], which accumulates Just Noticeable Difference (JND) values for image blocks of 32 pixels by 32 lines by 4 fields deep [68].

The Peak Signal to Noise Ratio (PSNR) method is the most famous method and is based on video signal parameters. PSNR estimates QoE by performing frame to frame peak signal-to-noise ratio comparisons of the original video sequence and reconstructed video sequence obtained from the sender-side and receiver-side, respectively. PSNR for a set of video signal frames is given by Equation (2.1) [3, 5, 70, 71].

$$PSNR_{db} = 20\log_{10}(\frac{V_{peak}}{RMSE}) \tag{2.1}$$

where V_{peak} is the maximum possible pixel value of the frame.

The recommended methods to obtain PSNR of a video signal are outlined in ANSI T1.801.03-1995 and the utility of this objective measurement of video quality is well-documented [3, 67, 68]. Because PSNR uses sender video data to compare received data to calculate and predict end-user perception, it is called Full Reference (FR) or intrusive measurement methods. To implement PQR and PSNR as FR measurement methods, special software and hardware are required. PQA 300 is the system which is used for measuring PQR and PSNR [69]. This system needs both sender and receiver side video signals simultaneously. Therefore, if sender and receiver are not located in one place, both video signals should be recorded and inserted into PQA 300 (or other systems) for achieving the PQR or PSNR results.

It is not always feasible to use Full Reference or even Reduced Reference (RR) methods since the reference may not be available. In addition, even the No Reference (NR) method would be prohibitively expensive to deploy widely for ongoing performance monitoring when there are many independent video streams to monitor as each of the streams would require separate decoders. Hence designing and implementing other methods which are based on network impairment parameters for estimating quality is vital. Moreover, the network impairment-based methods are advantageous compared with above-mentioned ones due to their online applicability.

Venkataraman et al. in [72] have discussed the effect of network events such as losses, jitter, and delay on video quality. To study this correlation they have used the PSNR and Video Quality Model (VQM) methods to estimate the quality. They have shown that PSNR reacts sharply with a noticeable drop initially with increasing packet error rate (PER). At high PERs, PSNR is almost constant.

The behaviour of the VQM method for lower PERs is the same as PSNR but at higher PERs, much unlike PSNR, VQM continues to rise. Other studies confirm the behaviour of these methods in lower PERs too. For example, Boyce et al. in [73] have noticed that packet loss rates as low as 3% translated into frame error rates as high as 30% in MPEG video. They have shown that PSNR and VQM have similar behavior with delay. PSNR and VQM remain constant due to pure delay alone, decreasing marginally at high delay. High delay causes packet losses in the network. They have also shown that PSNR reacts strongly to values of jitter exceeding 0.05 s, dropping rapidly but VQM can tolerate jitter levels of around 0.06 s.

In most recent studies for estimating the end-user perceived quality, researchers have tried to focus on one or two network impairment parameters. Reibman et al. in [9] have estimated video QoE by processing bitstreams and they have shown the impact of packet losses. They have presented three methods: FullParse (parsing all the packet's bits and the impact of packet losses on video quality), QuickParse (extracts only high-level information from the video bitstream and the impact of packet losses on video quality) and finally NoParse (which rely only on the number of packet losses and the average bit-rate). They have concluded that there is a trade-off between simplicity and accuracy and they have shown a correlation between the NoParse method's results and reality greater than 70%. The NoParse method estimates the number of packet losses simply by counting the number of observed losses in the video bitstream. Thus, this method can be implemented at the end-user's to estimate video quality online. Since this method utilizes bitstream, it can employ a simple software along with the receiver program.

Shu Tao et al. in [7] have developed a model to characterize the relationship between video distortion and packet loss. In general the loss-distortion model has been based on the impact of network losses on video quality as a function of application-specific parameters such as loss recovery technique, codec bit rate, video characteristic, packetization, etc. They have introduced a relative quality metric (rPSNR) that measures video quality against a quality benchmark that the network is expected to provide. This method estimates video quality without parsing or decoding the transmitted video bit stream and without knowledge of video characteristics. They have demonstrated the robustness and accuracy of the rPSNR-based method through several simulations and experiments. rPSNR can be computed solely based on parameters, some of which are predefined or determined according to the application configurations while others can be obtained through simple network measurements; therefore it can be categorized as an online video quality assessment method. rPSNR can be implemented by installing monitoring software in the client device.

Prasad Calyam et al. in [5] have introduced a framework that can provide online estimates of VVoIP QoE on the network path without requiring any video sequences nor end user involvement. They have estimated end-users' perception of video and voice quality in term of Good, Acceptable or Poor (GAP) grades of perceptual quality solely from the online measured network parameters. To validate their framework, they have compared their result with the subjective test. This method needs to be trained. For training, a subjective test should be deployed. Prasad Calyam et al. in [74] have introduced a set of methods to reduce the number of test cases per human subject for providing rankings without compromising the ranking data required for adequate model coverage. Since subjective tests are time consuming and expensive, reducing the number of tests can significantly improve the testing process.

Objective Perceived Voice Quality Assessment

Recall that, like video, voice quality test algorithms, depending on the information that is made available to an algorithm, can be divided in two categories:

- Full Reference (FR)
- No Reference (NR)

In FR, the algorithm accesses and uses an original reference signal for a comparison. To estimate the end-user perception it compares samples of received voice with the original voice sample (from the talker side).

One of the FR method for assessing the perceived quality of voice is the Perceptual Speech Quality Measure (PSQM) which is defined as ITU-T Recommendation P.861 [2, 3, 62]. This model is based on Bark spectral distortion and Fig. 2.4 shows the operation of PSQM as a block diagram. PSQM was developed to test voice quality through a voice encoder/decoder (codec). Because this model does not have sufficient performance under error-prone coding conditions, it is not useful to evaluation of voice over the Internet, which suffers from packet loss.

Another measurement technique very similar to the PSQM is the Perceptual Analysis Measurement System (PAMS) [2, 3]. This technique was developed by PTT (Post, Telegraph, and Telephone) for the United Kingdom. PAMS is designed for network-wide testing and is capable of taking into account packet loss, delay, jitter, and other events that would affect voice quality in a VoIP network.

Consequently, based on the PSQM and PAMS algorithms, another FR method for assessing the quality of voice is Perceptual Evaluation Speech Quality (PESQ), defined by

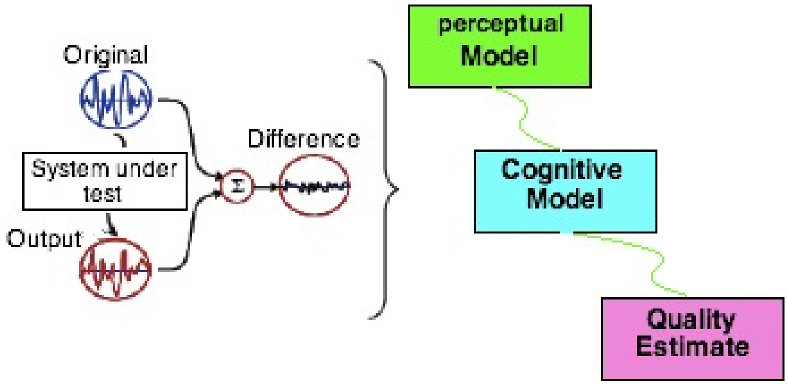


Figure 2.4: Block diagram showing the operation of PSQM and similar algorithms, from [3].

ITU-T Recommendation P.862 [2, 3, 75]. It has additional processing steps to account for signal-level differences and the identification of errors associated with packet loss.

Perceptual Objective Listening Quality Assessment (POLQA), also known as ITU-T Rec. P.863, is the successor of PESQ. Weaknesses of the current P.862 model are avoided in POLQA which is extended towards handling of higher bandwidth audio signals. Similarly to P.862, POLQA supports measurements in the common telephony band (300-3400Hz), but it also has a second operational mode for assessing HD-Voice in wideband and super-wideband speech signals (50-14000Hz) [76].

According to the viewpoint of input information to the test, other objective voice quality models are packet-layer objective models (i.e., parametric models). In these models objective quality assessment is based solely on IP packet information and not on payload data. These techniques are usually used for real time quality monitoring. In these methods, packet loss, delay, and jitter are considered. One of these objective voice quality assessment models is the E-Model. The E-Model is one of the most popular and standardized technique for measuring the quality of the voice [2–4]. Transmission Rating or R as an objective metric indicator is the main output variable of the ITU E-Model recommendation

G.107 [77]. R is a function of 20 input parameters that represent environment, terminal, and network quality factors. The value R can be mapped to a MOS scale and shows the level of subjective quality. Since the E-Model is designed for telephone band (300-3400 Hz) communications, it is not applicable to the evaluation of wide-band (e.g., 150-7000 Hz) communications [78]. Another model based on multiple effects is Call Clarity Index (CCI) which was developed by British Telecom. Since this model was provided for Public Switched Telephone Networks (PSTN) systems, it is unsuitable for evaluating the quality of VoIP which suffers from long delays, packet loss, etc. [4].

Parameter-based methods, such as the E-model, predict the perceptual quality based on given parameters. Other NR tests are signal-based ones which predict the speech quality by utilizing the degraded speech signal. One of the most popular signal-based standard is ITU-T P.563 [79].

Batu Sat et al. in [4] have described their test for comparing some VoIP systems. Their results have compared each system under different network parameter conditions (combination of No, Low, and High for each Loss, Delay, and Jitter). Their tests have shown that Windows Live Messenger is strongly preferred over other systems under lossy conditions. Skype has a better reputation than Google Talk and Windows Live Messenger which are slightly preferred over Yahoo Messenger under jittery conditions. They have used a classifier to help them predict subjective evaluation results using objective measurement and employed a support vector machine (SVM) [80] for classification. They have inserted 22 inputs (features) that could be objectively obtained. Thus, their method can be one of those tests which include massive mathematic calculations. Finally, they have verified their results by subjective tests and the PESQ method.

2.3 Network Parameters

As we have discussed, for realtime perceptual quality measurement and online assessment, objective methods are used. The significance of online measurement methods is to help the service providers improve their service quality to gain customer satisfaction in case of occurrence of the impairments and degradation. The improvement of perceptual quality from the service provider's point of view can be done by changing network parameters. Therefore, the effects of network parameters on final end-user perception should be comprehensively investigated. The most important network parameters in estimating the perceptual quality level are loss, delay, and jitter. The impact of each of these parameters in multimedia delivery over IP depends on several aspects which will be explained in the following sections.

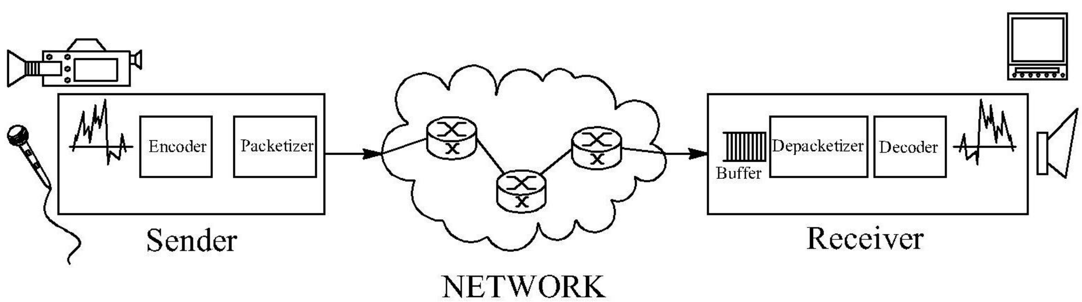


Figure 2.5: Media over IP system.

2.3.1 Loss

Loss often happens because of congestion. Delay and jitter lead to packet discard. Link failures and noise can also cause packet loss.

An overall view of multimedia (i.e., voice) delivery over IP is depicted in Fig. 2.5. The function of the decoder is to convert multimedia signals into a convenient format (binary) for transfering through a network. Afterwards, the bits are packetized in a specific size and format. In this step, a set of source signal bits and some additional information (header) needed for handling the data are gathered in a packet.

In the decoder, the codec determines how the main signals are digitized for transmission. The coding process reduces the quality per-se; the amount of reduction in quality depends on the digitization type and the bit rate of the selected codec. G.711, G.729, and G.723.1 are the most famous codecs for digitizing voice. Table I shows the techniques and data rate of these codecs. Modern encoding schemes, such as G.729 and G.723.1, achieve higher compression but this makes them less tolerable to loss. In other words, the effect of one packet loss on a voice signal coded with G.729 or G.723.1 is significantly adverser than the one coded with G.711. Another system characteristic that affects the role of packet loss through packet-switched network on end-user perception is the way that packets are constructed for transmission (packetizing). The ratio of the header size and the payload, beside the length of the whole packet, plays a significant role in the effect of packet loss on perceived quality. Therefore, the codec in which the length of packets is longer suffers more from the effect of loss on degradation. Dependency of the contents in each packet to the data in the adjacent packets is another effective parameter on packet-loss's impact on the final perceptual quality. This characteristic can also be used for compression in video codecs. So, losing a packet can affect the decoding the adjacent packets. Packet Loss Concealment (PLC) is a method to reduce the effect of loss on perceptual quality which is used at the distant end by encoder. Markopoulou et al. in [81] have said that roughly 1% packet loss for codecs with PLC causes about 4% increment in impairment while in codecs without PLC this increment is about 25%. Another parameter that should be considered in packet loss issues is burst loss. Burst loss generally produces a larger distortion than an equal number of isolated losses [82, 83].

Voldhaug et al. in [84] have shown the effect of packet-loss on high quality wide-band audio. They have shown that the acceptability decreases significantly even for low loss rates (0.5%). Once we have 1.5% packet loss the audio quality mapped to acceptability drops to 29% which can not be considered acceptable anymore.

Cermak et al. in [85] have presented the end-user perceived quality for video as a function of packet loss, frame size, bit rate, and codec. They have explained the relationship of all of these parameters vs. perceptual quality assessment. Shu Tao et al. in [7] have developed a model which characterizes the relationship between video distortion and packet loss as a function of loss recovery technique, coding bit rate, video codec, and packetization. By knowing the characteristics of the video content, their method has evaluated the quality of a video transmitted on a network by monitoring the loss impairment.

Reibman et al. in [9] have presented three methods to estimate the video quality by measuring the Packet Loss Rate (PLR). In FullParse and QuickParse methods, video quality has been estimated by considering the video content characteristics and PLR. In NoParse, they have relied only on measurements of the PLR. They have used the Mean Squared Error (MSE) of sent pixels and received pixels as a rough measure of video quality. They have stated that the correlation between the real MSE and the calculated MSE for FR is more than 99% and for NP is around 93%. It is clear that MSE is not linearly proportional to perceptual quality; but it is definitely helpful to estimate it.

2.3.2 Delay

According to Fig. 2.5, delay is the product of sending, transferring and receiving process. While sending, encoding, and packetizing produce a delay related to codec type. For example, table 2.1 shows the amount of delay which is produced by some voice codecs. In this table, voice segment duration is the time that codec must wait to receive and buffer the voice sample before the encoding with the number of bits in segment can be effected. In the Internet, network elements are the most important source of delay. Propagation time is one of the inevitable causes of delay. Routing, queuing, multiplexing etc. in network equipments are the other causes of network delay. The receiver, buffering, decoding, and other processes produce delay too.

Table 2.1: Voice codecs characteristics [2].

Codec	Encoding Technique	Segment Duration, ms	Segment Size, bits	data Rate, bit/s
G.711	PCM	0.125	8	64000
G.723	MP-MLQ	30	189	6300
G.723.1	ACELP	30	158	5300
G.729	CELP	10	80	8000

In one-way media streaming over IP, pure constant delay does not affect end-user perception. The end player can buffer the received data and play it after a while to make sure that there is enough data to play. Because the waiting time at the receiver is limited, the data with a long delay is assumed to be a data loss and its impact is the same as a lost packet. Delay plays an important role in interactive communications; because the receiver parts cannot wait and buffer the received packets for a long time, thus the packets which are received with a relatively long delay are discarded.

For IPTV and Video on Demand (VoD), the delay would be very important for interactive control. Channel change speed or channel zapping delay is important for customers. Delays greater than 2 sec for channel change cause costumers' dissatisfaction, but the indication of receiving the commands should be less than 200 ms [3].

For VoIP, end-user perception due to delay impairment depends on the turn-talk frequency. The faster the turn-talk, the more important the role of the delay in perceived quality. In conversation, delay can cause double-talk and interruption which degrades conversational quality. If the system does not use echo cancellation, delay might aggravate the effect of echo destructive on user satisfaction. Lai et al. in [86] have shown that in a typical conversation in daily life, if the one way delay exceeds 300 ms, the MOS goes under 3. It can happen sooner for faster turn-talk. For example if the participants, say, alternate number from 1 to 10 in turn as fast as possible, the MOS may fall below 3 when one way delay exceeds 40 ms. Coleand et al. in [87] have said that the impairment associated with the Mouth-to-Ear (ME) delay can be approximated by a piece-wise linear function of delay. They have shown that in a normal conversation, for one way delays less than 177 ms, conversations occur naturally, whereas at delays in excess of 177 ms conversations begin to strain and break down. There is a numerous literature on delay but there is no consensus on its effect [88]. ITU-T G.114 recommends 400 ms for the upper threshold of one-way delay for an acceptable conversation quality [89]. Gueguin et al. in [90] have shown that in an echo free system, the MOS is greater than 3.5 for a conversation impaired with 600 ms one-way delay.

For FR tests (e.g PESQ) there are some methods for determining the amount of delay. Cross-correlation, envelope cross-correlation, average zero-crossing rate, cepstral distance, and statistical norms are the most popular methods for delay determination [91, 92]. For the objective tests which are based on the measurement of the network's parameters, the delay is computed using the information in the Real Time Protocol (RTP) header.

2.3.3 Jitter

Jitter is the variation in delay/queuing caused by differences in the time taken for the packets to cross the network. Buffering in packet switched networks causes jitter. Jitter increases when the traffic becomes more bursty [93]. Jitter can degrade video quality as much as packet loss [94]. Jitter is considered because the decoding of digital signal is a synchronous process and the data must be fed to the decoder at a constant rate [3]. For decreasing the effects of the jitter, the receiver part employs a jitter buffer. Variation in packet arrival times is smoothed out using this buffer. Defining the size of the jitter buffer is an important issue to study [2, 95–97]. A trade-off between the dropped frame probability and increases in the transmission delay is the issue in calculating the size of the jitter buffer. Jitter is a more substantial parameter for interactive communications rather than data streaming (e.g., IPTV). By deployment of the jitter buffer, jitter is not considered a separate impairment because the effects of the jitter in the output are defined as delay (waiting time in jitter buffer) or as distortion of packet loss (packets arrive after the deadline or in case of full buffer will be dropped) [3]. For broadcasting TV, the jitter buffer in the Set-Top Box (STB) adds delay in the range of 100 to 500 ms to the final end to end delay.

Calyam et al. in [74] for VoIP have suggested that for a good conversation, the jitter should be less than 20 ms; but between 20 ms and 50 ms it can be acceptable if the delay for a good conversation is below 150 ms and the delay for acceptable conversation is between 150 ms and 300 ms. Kelly in [98] has said that "experience has shown that an end-to-end delay of 200 ms is still usually satisfactory for most users". Therefore jitter should not be more than 20 to 50 ms. The recommended amount of jitter for transmitting the video encoded with MPEG-2 is less than 50 ms [3].

2.4 Conclusion

The prime and foremost criterion for assessing the quality of voice and video communication is end-user perception of quality of service or in other words subjective quality. The most widely used metric for measuring subjective quality is the Mean Opinion Score (MOS). However, although subjective quality tests are the most reliable methods, they are also expensive and time consuming. Thus, using methods for estimating subjective quality from physical quality parameters is vital. The tests which use these methods are called objective quality tests. As we have mentioned, most studies which work on objective tests have also used subjective tests to validate their results. So, we can say that, to design a system, both subjective and objective metrics should be considered in the evaluations since each alone is inadequate; but to control and adjust the network elements under which the system works, objective methods, such as online measuring should be employed.

Chapter 3

Packet Loss Probability Estimation

3.1 Introduction

The quality assessment of media communication systems and the parameters which affect this quality have been an important field of study for both academia and industry for decades. Due to the interactive or online nature of media communications and the existence of applicable solutions to deduce the effect of delay and jitter (e.g., deployment of a jitter buffer at the end user node [95, 96]), data loss is a key issue which must be considered. If there is a possibility for online accurate measurement of packet loss, then the network service providers can take the appropriate action to satisfy the contractual Service Level Agreement (SLA) or to improve and troubleshoot their service without receiving end user feedback.

Packet loss often happens because of congestion. In other words, buffer overflow at the outgoing interface in intermediate network nodes causes packet loss. Since measuring packet loss ratio at the intermediate nodes in high speed networks does not seem applicable in real time, some recent research has focused on estimation of packet loss probability (plp) [10, 19, 20, 45, 99].

According to central limit theory, the aggregated input traffic at intermediate nodes in the network core can be described with a Gaussian model [100, 101]. Based on the Large Deviation Theory (LDT) and the large buffer asymptote approach, the plp can be estimated by a stochastic process considering the probability of buffer overflow in a finite buffer system where b is the buffer size (or tail probability $\mathbb{P}\{Q > b\}$ in a infinite buffer system). Since the input traffic is described by a Gaussian process, the latter can be identified by an online measure of the mean and variance of the input traffic.

In this chapter, we propose a tighter approximation of plp based on the input traffic process and the information which was measured in the past. In other words, we use some online measures and historical data for accurate estimation and thus improve on earlier proposed estimates. Our plp estimation method can also compose with systems whose buffer size is not large enough to meet the assumptions of the large buffer asymptote approach.

Furthermore, this estimate can integrate well with a quality control architecture. Using the online estimated plp as feedback information, a control system could properly throttle the ingress traffic rate and keep the plp below some target upper bound value of packet loss in an SLA. An overall architecture of measurement, estimation, and control loop to keep the quality of service/experience within the SLA bounds is shown in Fig. 3.1. In this figure, the estimated plp is used as an online transducer in a control loop of packet loss.

This chapter continues in Section 3.2 by reviewing prior bodies of work on measuring or estimating the packet loss probability. Section 3.3 provides some useful definitions which are employed in this chapter. In Section 3.4, we develop a new plp estimator. Section 3.5 presents the testbed and the simulations used to assess the quality of our estimator. Numerical results and comparison that demonstrate the effectiveness of our proposed estimator are presented in Section 3.6. Section 3.7 concludes the chapter.

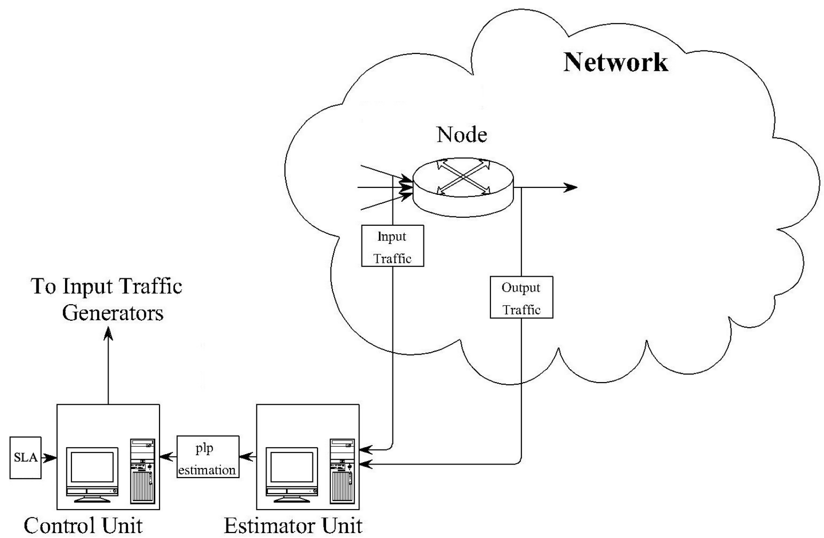


Figure 3.1: Measurement, estimation, and control loop schematic.

3.2 Previous work

In our observations, earlier research on measuring and modelling the packet loss would generally either increase the burden of probe packets' bit rate to the available bandwidth [10–12] or not provide real time information [13–15]. For example, [13] and [14] have characterized loss traces by identifying mathematical models. Yin Zhang et al. in [102] and [103] have analyzed the stationarity of the loss process on the Internet paths and studied its predictability. Although these studies are undoubtedly useful to understand the general loss characteristics, they cannot be used in real time performance estimation and consequently online control systems.

To obtain real time network performance information such as available bandwidth, delay, and loss, various probing techniques have been recently used by researchers. For instance, [14, 104] and [105] have employed packet pair and packet train techniques, respectively, to measure bottleneck bandwidth. He et al. in [106] have used probing methods to explore end-to-end traffic by exploiting the long range dependence nature of Internet traffic. The authors of [11] and [12] have measured the loss rate on individual links by end-to-end multicast/unicast probes and different inference techniques. Further, Tao and Guérin in [10] have used a probing method to construct a Hidden Markov Model (HMM) [107] to capture the main characteristics of the loss process such as loss length distribution, loss distance, etc. The disadvantage of these methods is to increase the burden of probe packets' bit rate to the available bandwidth when better accuracy is required.

To cope with the shortcomings of the aforementioned methods, many researchers have tried to link the input process to the probability of loss at intermediate nodes. Behavior of the FIFO scheduler fed by many on-off sources was investigated by Anick et al. in [16]. Elvalid et al. and Stern et al. in [17] and [18], respectively, extended Anick's work by presenting a simple approximation of the loss for a very large buffer size system whose input can be modelled with Markov Modulated Rate Processes (MMRP). Their mathematical models are derived from large deviation theory (LDT).

Studies which estimate loss probability based on input traffic process generally fall into one of the following methodological categories given their underlying assumptions:

- *Large buffer asymptote*: In this approach, the intermediate node's buffer size is assumed to be large. The value of overflow and consequently loss attained in the case of small buffer size is extrapolated using the large buffer asymptote. Chang in [108] and the references therein review this topic comprehensively. Zhang and Ionescu in [19–22] have extended this research to estimate the loss probability.

- *Large number of independent and stationary sources*: This method is based on the homogeneity of n identical sources that feed the intermediate node's input buffer. Likhanov and Mazumdar in [109] used this methodology to estimate the loss probability.

- *Aggregate traffic approximation*: This approach is used to reduce the computational complexity of input traffic model estimation. It is employed when an intermediate high-speed node's input traffic consists of a large number of individual user traffic flows with unique characteristics, in which case the large number of sources asymptotic method is not applicable [110]. The main justification for a packet loss probability estimation based on aggregate traffic approximation is the Bahadur-Rao Theorem, which computes the asymptotic tail distribution of the sum of n identically non-lattice random variables when $n \to \infty$ [111].

In our research, we have used the *large buffer asymptote* approach for online packet loss estimation. Our work revisits Zhang and Ionescu's research [19–22] (i.e., recent work on this topic); we will review their method and explain how we overcome its shortcomings at the end of section 3.4.

3.3 Definitions

The input traffic model and packet loss probability are explained in this section. All the definitions are related to a high speed intermediate node in which the received packets are served with First In First Out (FIFO) scheduling.

3.3.1 Input traffic model

According to the Central Limit Theorem (CLT), the aggregated traffic at an intermediate link in a high-speed network can be well approximated by a Gaussian process [112–114]. Moreover, characterizing the input process of a large number of sources with the traditional Markovian models seems infeasible. Therefore, in our study the input process $\lambda(t)$ is characterized by a Gaussian process and presented by

$$\lambda(t) = \mu t + \sigma Z(t) \tag{3.1}$$

where μ and σ^2 are the mean and variance of arrival rate (i.e., $\lambda(t)$), respectively and $Z(t)$ is a centered Gaussian process.

3.3.2 Packet loss probability

The packet loss probability, P_{loss}, is defined as the long term ratio of the number of lost packets to the number of input packets. It is expressed by the following formula:

$$P_{loss} = \lim_{N\to\infty} \frac{\sum_{k=1}^{N}(q_{k-1} + \lambda_k - c - b)^+}{\sum_{k=1}^{N}\lambda_k} = \frac{\mathbb{E}[l_k]}{\mathbb{E}[\lambda_k]} \tag{3.2}$$

where $(x)^+$ denotes $max\{x, 0\}$, b is buffer size, c is output link capacity, and q_k and l denote the number of packets that occupy the buffer in the time interval $[k, k+1)$ and the number of lost packets, respectively.

The packet loss ratio, $plr(k)$, is defined as the short term ratio of the amount of packets lost to the amount of input packet. It is expressed by the following formula:

$$plr(k) = \frac{l_k}{\lambda_k} \tag{3.3}$$

where l_k is the number of lost packets during the time slot $[k, k+1)$ and λ_k is the number of packets that arrive during the time slot $[k, k+1)$.

Kim and Shroff in [112] showed that the plp in a buffer of size x can be well approximately mapped from the tail probability in the infinite buffer system. Tail probability also called the *overflow probability* $\mathbb{P}\{Q > x\}$ is expressed as

$$\mathbb{P}\{Q > x\} = \lim_{N\to\infty} \frac{1}{N} \sum_{k=1}^{N} I(Q_k > x) \tag{3.4}$$

where $I(A)$ is an identification function which is equal to 1 if A is true and equal to 0 otherwise, and Q is the dynamic queue size. Although $\mathbb{P}\{Q > x\}$ is averaged by time and

plp is averaged by the input, [112] shows the following relationship between $\mathbb{P}\{Q > x\}$ and plp:

$$P_{loss}(x) = \alpha\mathbb{P}\{Q > x\} \tag{3.5}$$

where α is constant and equal to $P_{loss}(0)/\mathbb{P}\{Q > 0\}$ and $P_{loss}(0)$ denotes the packet loss probability in a bufferless system.

3.3.3 Effective bandwidth

The *effective bandwidth* of arrival traffic process $A(t)$ is defined as

$$\omega(\theta, t) = \frac{1}{\theta t} ln\mathbb{E}[e^{\theta A(t)}] \quad 0 < \theta, t < \infty \tag{3.6}$$

where θ and t are system parameters determined by the channel capacity and buffer size, the QoS requirement, and the characteristics of the multiplexed sources [115]. Based on the Gärtner-Eliss theorem [116, 117], $\omega(\theta, \infty)$ exists when the input traffic is Gaussian. So,

$$\omega(\theta^*, \infty) = \lim_{t\to\infty} \frac{1}{\theta^* t} ln\mathbb{E}[e^{\theta^* A(t)}] = c \tag{3.7}$$

where c is link capacity. Glynn and Whitt in [118, 119] have proved that overflow probability can be related to θ^* which is calculated from (3.7) as following

$$\lim_{x\to\infty} \frac{1}{x} ln\mathbb{P}\{Q > x\} = -\theta^*. \tag{3.8}$$

3.4 Packet Loss Probability Estimator

There are several approaches to estimate packet loss probability. Sending probe packets periodically through the path and processing the returned signals for predicting the performance of path (e.g., packet loss ratio, delay, etc.) is one of the recent methods for estimating the plp [10, 120]. The disadvantage of this method is to increase the burden of probe packets' bit rate to the available bandwidth when greater accuracy is requested.

Estimation of plp based on stochastic input traffic process is another approach in this field [19, 20, 121]. In this method some important assumptions are made as follows: 1) Measurement and estimation take place at intermediate nodes in high-speed network core links, therefore the input traffic is a mix of a large number of individual traffics and thus the Gaussian process model is considered to represent the stochastic input traffic process [100, 101]; and 2) the size of the buffer should be large, otherwise the queue process is not exponential and the behaviour of the traffic in small buffers cannot be approximated by a logarithmically linear behavior [108, 122, 123], so the input traffic process cannot estimate plp.

Following the Gaussian model assumption for the input traffic, the effective bandwidth in this model [115] is given by:

$$\omega(\theta, t) = \mu + \frac{\theta}{2t}\sigma^2 VarZ(t) \tag{3.9}$$

where θ is the *space* parameter, *t* is the *time* parameter which corresponds to the most probable duration of the buffer congestion period prior to overflow, μ is defined as the *traffic mean*, *Var* represents the second moment of *Z(t)* and is equal to t^{2H}, σ^2 is the *variance* of the input traffic random variable, and *H* is the *Hurst* parameter.

The Hurst parameter H shows the degree of self-similarity in the traffic. H=0.5 corresponds to a well behaved Gaussian traffic while any value larger than 0.5 indicates a self-similar traffic source. Based on the classical assumption for input traffic [121, 124], the H parameter is set to 0.5. So the effective bandwidth is finite, independent of time, and can be simplified into:

$$\omega(\theta, t) = \mu + \frac{\theta}{2}\sigma^2. \tag{3.10}$$

Further, if μ and σ exist, effective bandwidth, in case of $t \to \infty$, is equal to link capacity (see (3.7)). Therefore,

$$\omega(\theta^*, \infty) = \mu + \frac{\theta^*}{2}\sigma^2 = c. \tag{3.11}$$

Based on our second assumption of large buffer asymptotic approach for packet loss estimation, the overflow probability for the large buffer size can be approximated by a logarithmic behavior as follows [118, 119]:

$$\exists\kappa \in \mathbb{R}^+, \mathbb{P}\{Q > x\} = \kappa e^{-\theta^* x} \tag{3.12}$$

where θ^* is the solution of (3.11). Note that such an approximation in (3.12) is more precise when the buffer size x is large [108]. Therefore, $\mathbb{P}\{Q = x\}$ can be defined by

$$\mathbb{P}\{Q = x\} = \kappa(e^{\theta^*} - 1)e^{-\theta^* x}. \tag{3.13}$$

To estimate the packet loss probability, $\mathbb{E}[l_k]$ of (3.2) is defined as follows (recall that b is buffer size):

$$\mathbb{E}[l_k] = \sum_{i=b+1}^{\infty}(i - b)\mathbb{P}\{Q = i\} \simeq \int_b^{\infty}(x - b)\mathbb{P}\{Q = x\}\, dx. \tag{3.14}$$

From (3.13) and (3.14), we have

$$\mathbb{E}[l_k] = \kappa(e^{\theta^*} - 1)\frac{e^{-\theta^* b}}{\theta^{*2}} \tag{3.15}$$

where θ^* calculated from (3.11) is

$$\theta^* = 2\frac{c - \mu}{\sigma^2}. \tag{3.16}$$

Solving (3.11) in θ^* and replacing in (3.15) define P_{loss} by the following equation:

$$P_{loss} = \frac{\mathbb{E}[l_k]}{\mathbb{E}[\lambda_k]} = \kappa(e^{2\frac{(c-\mu)}{\sigma^2}} - 1)\frac{e^{-2b\frac{(c-\mu)}{\sigma^2}}}{4\mu\frac{(c-\mu)^2}{\sigma^4}}. \tag{3.17}$$

Applying the logarithm to (3.17), we derive the following estimator:

$$ln(P_{loss}) = ln(e^{2\frac{(c-\mu)}{\sigma^2}} - 1) - 2b\frac{(c-\mu)}{\sigma^2} - ln\left(4\mu\frac{(c-\mu)^2}{\sigma^4}\right) + ln(\kappa). \tag{3.18}$$

In line with other similar studies [19, 20], we change the base of the logarithm function from e to 10. Thus, (3.18) can be replaced by:

$$log(P_{loss}) = log(e^{2\frac{(c-\mu)}{\sigma^2}} - 1) - 2b\frac{c-\mu}{\sigma^2}log(e) - log\left(4\mu\frac{(c-\mu)^2}{\sigma^4}\right) + log(\kappa). \tag{3.19}$$

Replacing μ and σ with their measurement value $\bar{\mu}(k)$ and $\bar{\sigma}(k)$ changes (3.19) into the following equation:

$$\begin{aligned} log(P_{loss}) = log(e^{2\frac{(c-\bar{\mu}(k))}{\bar{\sigma}^2(k)}} - 1) - 2b\frac{c-\bar{\mu}(k)}{\bar{\sigma}^2(k)}log(e) \\ -log\left(4\bar{\mu}(k)\frac{(c-\bar{\mu}(k))^2}{\bar{\sigma}^4(k)}\right) + \kappa' \end{aligned} \tag{3.20}$$

where $\kappa' = log(\kappa)$ and $\bar{\mu}(k)$ and $\bar{\sigma}(k)$ are defined as:

$$\bar{\mu}(k) = \frac{1}{N}\sum_{i=0}^{N-1}\bar{\lambda}(k-i) \tag{3.21}$$

and

$$\bar{\sigma}^2(k) = \frac{1}{N-1}\sum_{i=0}^{N-1}\left[\bar{\lambda}(k-i) - \bar{\mu}(k)\right]^2 \tag{3.22}$$

where $\bar{\lambda}(k)$ is the measured input packet rate in the kth time interval and N is the number of time intervals for calculating the average of the mean and variance of the packet rate.

In the rest of the chapter let $epl(k)$ denote the $log(P_{loss})$, which is estimated by

$$epl(k) = log(e^{2\frac{(c-\bar{\mu}(k))}{\bar{\sigma}^2(k)}} - 1) - 2b\frac{c-\bar{\mu}(k)}{\bar{\sigma}^2(k)}log(e) - log\left(4\bar{\mu}(k)\frac{(c-\bar{\mu}(k))^2}{\bar{\sigma}^4(k)}\right) \quad (3.23)$$

and let $plp(k)$ denote the logarithm of real packet loss probability during the time slot $[k, k+1)$ which can be expressed by:

$$plp(k) = log\left(\frac{l(k)}{\lambda(k)}\right) \quad (3.24)$$

where $l(k)$ is the number of packets lost during the time slot $[k, k+1)$ and $\lambda(k)$ is the number of packets that arrive during the time slot $[k, k+1)$.

Some estimation errors are expected due to the assumption made for the stochastic traffic process (e.g., time independent traffic assumption and $H = 0.5$) and the simplifications and approximations employed in (3.23) (e.g., κ' is eliminated from (3.20)). Numerical results in the next section show that estimating the plp with (3.23) completely follows the variation of plp, although there is an almost constant offset between the real plp value and epl which is best explained from ignoring the constant κ' in (3.20).

To eliminate this difference it is proposed to use the offline measured plp and compare it with the estimated one to obtain the offset. We therefore present an improved estimator, iep, defined as:

$$iep(k) = epl(k) + \frac{1}{n}\sum_{l=1}^{n}[plp(k-l-m) - epl(k-l-m)] \quad (3.25)$$

where m is the number of interval periods after which the data of plp is available and $epl(k)$ and $plp(k)$ are calculated via (3.23) and (3.24), respectively.

With this improved estimator, the required time for measuring and calculating the plp is represented by m in (3.25), where the mean of errors between epl and plp during a moving window (i.e., n time intervals) in the past (i.e., m time intervals ago) is added to epl to estimate the new plp. Note that the duration of the time interval is independent from the measurement and calculation speed of plp. In other words, the estimator depends on m, in (3.25), only for the duration of the measuring time interval.

As we have mentioned in section 3.2, Zhang and Ionescu [19,21,22] also have proposed a packet loss probability estimator based on LDT and buffer asymptote approach. Their estimator describes packet loss probability by:

$$epl' = log(P_{loss}) = -2b\frac{c-\mu}{\sigma^2}log(e) - log\left(2\mu\frac{(c-\mu)}{\sigma^2}\right). \tag{3.26}$$

To cope with their estimator's error, they have introduced a Reactive Estimator (re) [20] which is defined as:

$$re(k) = epl'(k) + \frac{1}{n}\sum_{l=1}^{n}[plp(k-l) - re(k-l)] \tag{3.27}$$

where epl' is packet loss probability estimated by (3.26).

A careful examination of (3.27) reveals that the error re attempts to correct will decrease to the amount of difference between re and plp, whereas the error really is the difference between epl' and plp.

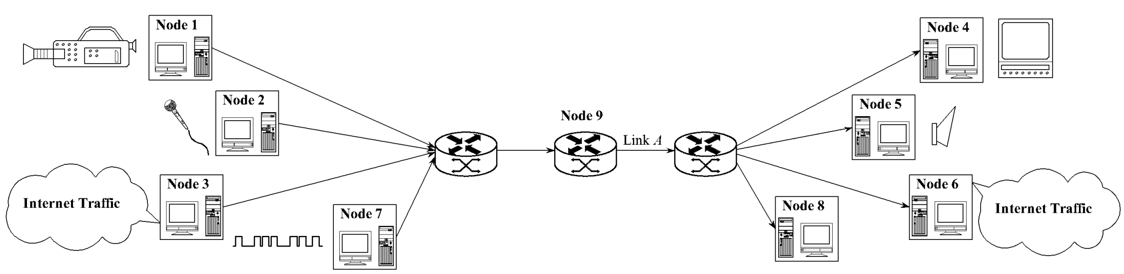


Figure 3.2: Testbed topology.

We thus claim that our proposed estimator, iep, does a better job at tracking plp. To investigate the accuracy and applicability of the aforementioned estimators and to compare their performance with that of our estimator, we propose to conduct simulations. In these simulations, the effects of different configurations of network traffic and packet loss ratio on estimators' performance are examined, and then will be discussed in detail in Sections 3.5 and 3.6.

3.5 Simulation Testbed

The NS-2 software [125] is used to simulate the network. The network topology which is simulated is shown in Fig. 3.2.

An MPEG2 traffic flow is generated by node 1 and the RTP protocol is deployed for transferring video data to node 4. Node 2 generates the voice traffic flow which is coded by G.729. This data is transferred to node 5. Node 3 and node 6 are designed to generate the common Internet traffic flow for background traffic and make the aggregated traffic situation closer to the Gaussian distributed traffic for stochastic input traffic process. The Tmix module in NS-2 is utilized in node 3 and 6 in order to generate realistic Internet network traffic [126]. The protocol deployed for communications between nodes 3 and 6 is TCP. Since the background traffic is TCP-based, congestion (i.e., buffer overflow and loss) affects traffic flows, which leads to a situation similar to that of a real Internet network traffic. Nodes 7 and 8 generate the on-off traffic to randomly increase the probability of packet loss. Measurement of the input and output traffics is performed at node 9. Since the focus is on node 9, the bandwidth of all links except link A is set to 100 Mbps and the buffer size of all nodes except node 9 is set to 500 packets. We vary the size of the node 9 buffer from 5 packets to 100 packets to examine different router configurations. To generate different amounts of packet loss, the bandwidth of link A varies between 7.4 Mbps and 7.8 Mbps. With these settings loss takes place only in node 9. When the bandwidth of link A is set to 7.8 Mbps and nodes 7 and 8 do not generate any traffic, the packet loss probability is about 0.1 percent and when the bandwidth is decreased to 7.4 Mbps, the packet loss probability in node 9 increases to about 1 percent, which is closer to the amount where the effect of loss on media communication quality becomes annoyingly noticeable. By turning on the traffic of nodes 7 and 8 at some short periods of time, the packet loss probability reaches 7 percent which is an unacceptable amount of packet loss for media communications. In the next section the numerical values of the different estimators in these situations will be examined.

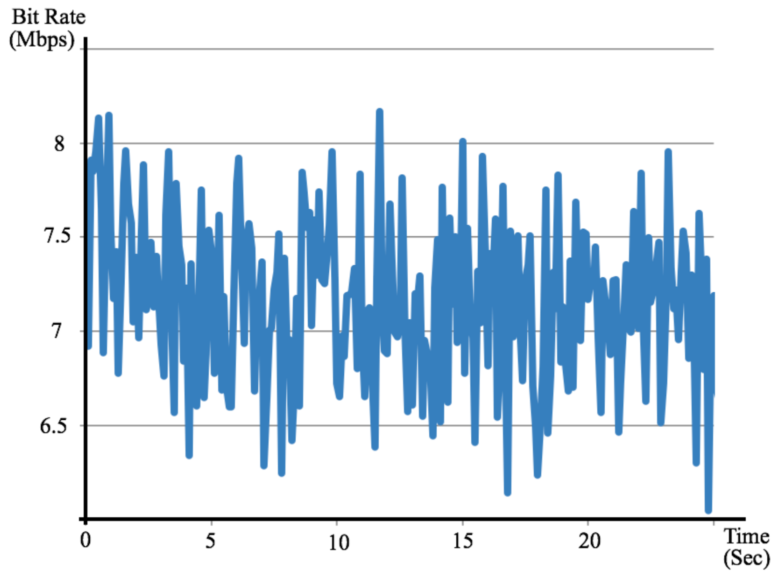
3.6 Numerical Results Analysis

This section presents the experimental results of the evaluation of the performance of the proposed estimator for the different types of traffic generated in the simulation testbed. The accuracy of the loss probability predicted by our proposed estimator is compared to that of a couple of other recent estimators.

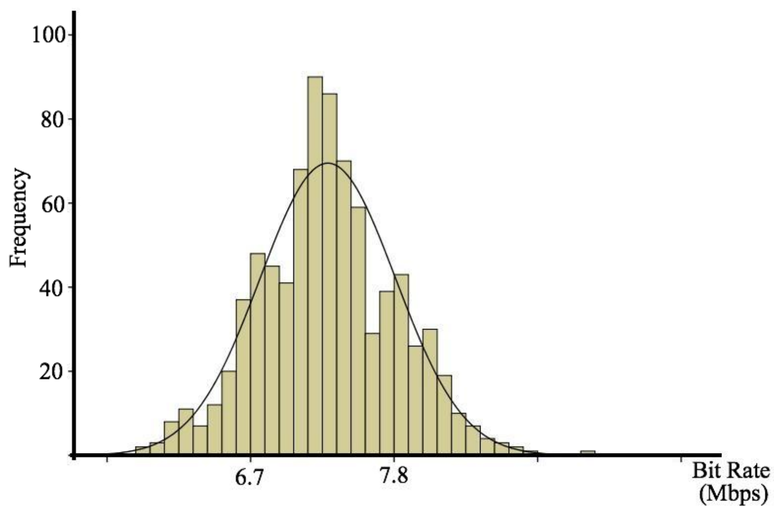
3.6.1 Input traffic

The crucial assumption in estimating loss probability based on an input traffic process is the Gaussian behavior of the aggregated input traffic. Therefore, the verification of this statement (i.e., the aggregated input traffic process is a Gaussian process) is the first test which should be performed. So, the received times of all packets for aggregated traffic are measured, while node 1 generates MPEG2 traffic flow, a voice traffic is generated by node 2, and node 3 sends an approximate common Internet traffic mix through the core of testbed.

In this study the graphical technique is used for normality testing, although, the Chi-Square test [127] could also be used to verify the assumption of Gaussian behavior of input traffic in our simulations. Fig. 3.3 which shows the instantaneous input traffic bit rate and the distribution of input traffic visually, verifies that in our simulations the aggregated



(a) Instantaneous input traffic bit rate.



(b) Histogram of input traffic distribution.

Figure 3.3: Aggregated input traffic characteristics in the network core.

traffic in the core link can be approximated by Gaussian traffic and consequently, the main assumption of proposed estimator is met.

3.6.2 Individual flow loss

To satisfy the SLA and to take the appropriate action on each flow's source, a control system needs to be aware of the packet loss probability of each flow. However, only the aggregated traffic loss probability can be estimated by the proposed estimator.

The simulation results show that the loss ratio of each flow (e.g., MPEG2 flow) is very close to loss ratio of the aggregated traffic. Therefore, it can be concluded that the estimated loss probability of aggregated traffic can be used as the individual probability of packet loss. Fig. 3.4 verifies this statement by showing that the measured MPEG2 flow's packet loss ratio is very close to the packet loss ratio of aggregated traffic in node 9.

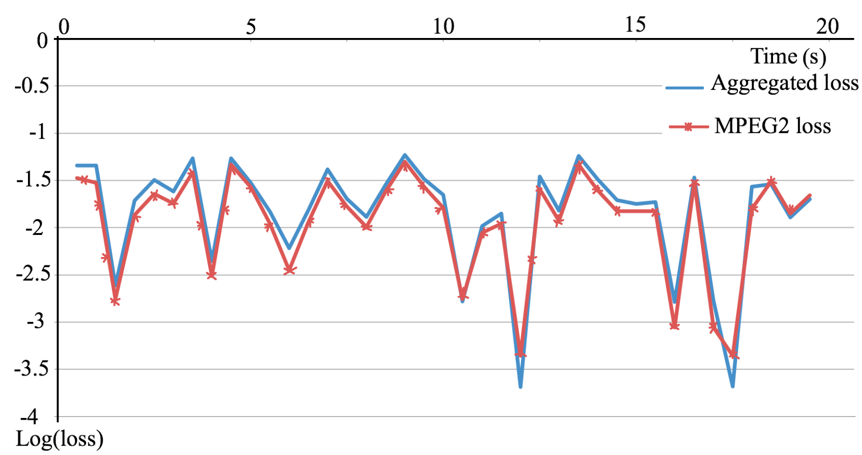


Figure 3.4: Comparison of the MPEG2 loss ratio with the aggregated traffic loss ratio.

3.6.3 Estimator performance

First, to evaluate that if the epl from (3.23) follows the plp variation with an almost constant offset, a situation has been investigated in which the bandwidth of link A was 7.4 Mbps and there was no traffic coming from nodes 7 and 8. As shown in Fig. 3.5, although there is an offset between plp and epl, epl follows the variation of plp thoroughly and this can be seen as a clear sign of soundness of the use of epl as the main part of proposed estimator.

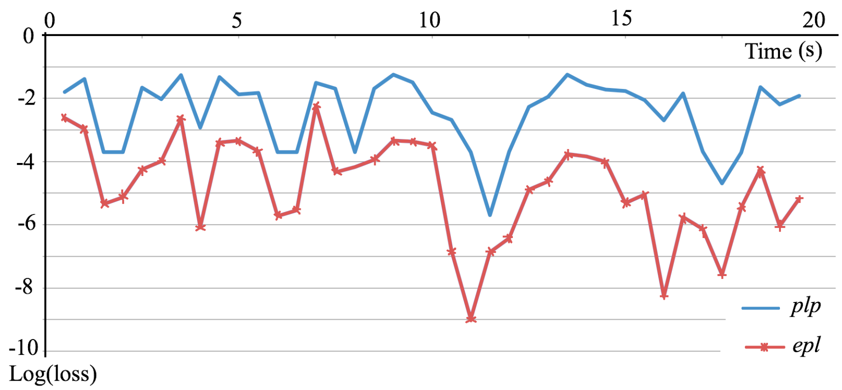


Figure 3.5: Comparison of plp (measured loss) and epl (estimated loss with offset).

Next, all the mentioned estimators (i.e., epl', re, and our proposed estimator, iep) are evaluated and their performance compared in different situations.

Fig. 3.6 shows the performance of the different estimators in a situation where the bandwidth of link A is 7.8 Mbps and there is no traffic coming from nodes 7 and 8. The accuracy of proposed estimator (iep) to estimate the plp compared to the other estimators is demonstrated in this figure.

In all experiments the time interval is 100 ms. In Fig. 3.6 iep is calculated according to (3.25) where m is 5. This means that iep uses plp data measured up to 500 ms earlier.

Since the amount of loss in the former example might be negligible for media communications, we change the network conditions to increase the loss ratio and then re–evaluate

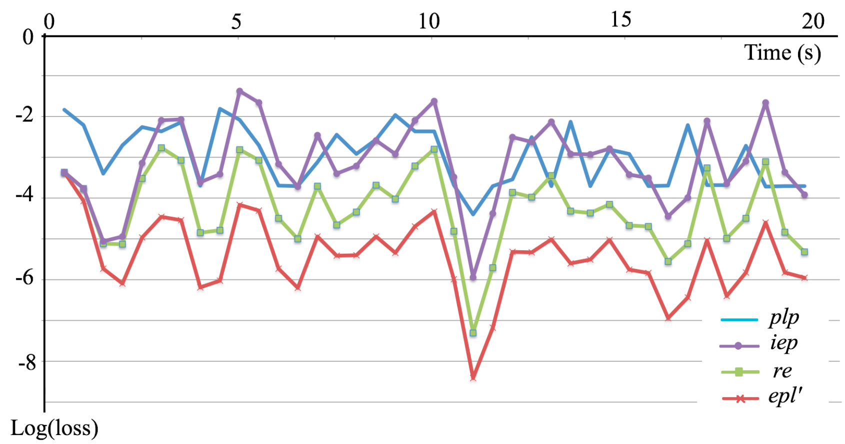


Figure 3.6: Measurement and estimation of packet loss probability when plp is about -2.5.

the accuracy of estimators. To achieve this situation, the buffer size of node 9 is decreased to 10 packets. Fig. 3.7 shows the results of this experience: during the time periods of $[10, 15]$, nodes 7 and 8 add network traffic and bring the loss ratio close to 7 percent ($log(plp) = -1.5$). As Fig. 3.7 shows, the effect of simplification and approximation in (3.26) and (3.27) on the operation of epl' and re methods is more apparent at this larger loss ratio.

Tables 3.1 and 3.2 summarize the statistics for the different estimators with varying loss ratio. In all comparisons the error is defined as the difference between estimated and measured plp.

As mentioned before, the buffer size affects the plp and the accuracy of estimators [122, 123]. The larger the buffer size, the lesser plp and the better the accuracy of the estimation. The effect of buffer size on estimation methods re and epl' has been examined in [19] and [21], respectively. The value of m, in (3.25), also affects the accuracy of iep estimation.

To examine the accuracy of the proposed estimator in different configurations (i.e., buffer size and m), we introduce a new variable, $error$. Given that the errors of logarithmic

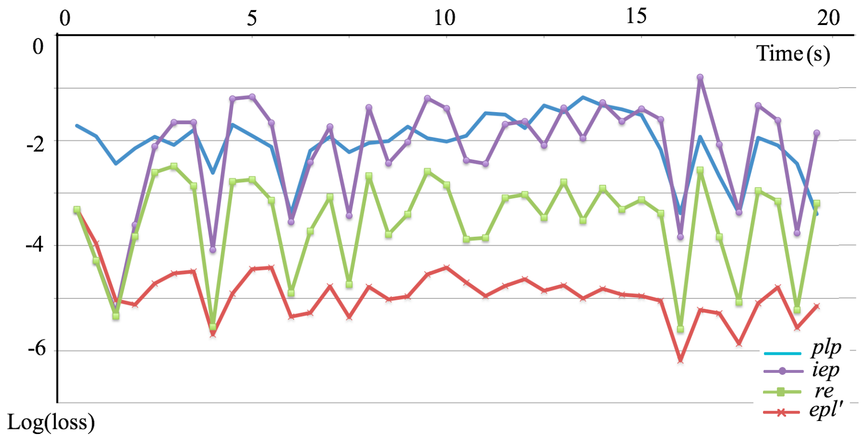


Figure 3.7: Measurement and estimation of packet loss probability when plp is about -1.5.

variables (plp's) are not easily comparable, $error$ is defined as follows to make it more sensible to small variations:

$$error = 10^{estimation} - 10^{plp} \tag{3.28}$$

Fig. 3.9 shows the probability density function of $error$ when buffer size is 10, 30, and 100 packets, and m is 5, 10, and 20 ($m = 10$ means using a plp measured 1 s before), and the effect of buffer size on estimation. Fig. 3.9-(a),(b), and (c) show that our proposed estimator has better performance in the case of a larger buffer. Note that a larger buffer size causes more latency which is not suitable particularly for multimedia transmission; hence, it should be set carefully. However, in our simulations, the buffer size of 100 packets causes only a 15 ms delay which could be even lower in real high speed intermediate networks.

It can be also shown by Fig. 3.9 that the offline measuring speed affects the accuracy of our proposed estimator: the faster the measurement, the more accurate the estimation.

Further considering the effect of buffer size on estimations derived from (3.23), it appears that the accuracy of estimation (iep) will improve if the role of the measured plp is increased. Therefore, (3.25) is changed to:

$$\begin{aligned} iep(k) &= p \times epl(k) \\ &+\frac{1}{n}\sum_{l=1}^{n}[plp(k-l-m) - p \times epl(k-l-m)] \end{aligned} \tag{3.29}$$

where p is the proportional coefficient and is less than 1. To increase the importance of the

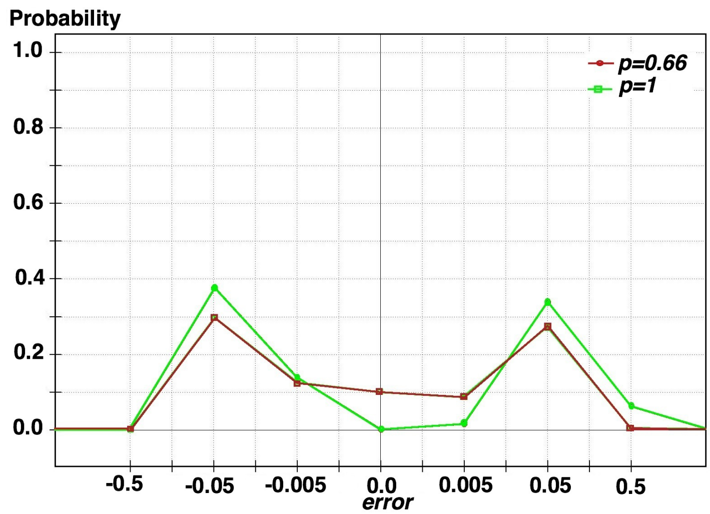


Figure 3.8: PDF of $error$ for estimator which uses (3.29) when buffer size is 10.

second term in (3.29), n is increased from 3, which is recommended in [20], to 10 and to decrease the effect of first part, p is set to $\frac{2}{3}$. For a smaller p, when a considerable variation happens to plp, the estimator (iep) cannot follow the plp properly and the value of $error$ will be significant.

Fig. 3.8 shows the value of $error$ when buffer size is 10 and (3.29) is used for estimation. Comparing Fig. 3.8 and Fig. 3.9(a), the effectiveness of the changes in estimation is clear.

Table 3.1: Statistics Synopsis on Loss Probability Estimation for Different Estimators When plp is About -2.5.

Estimator	Error* Mean	Error Variance	Error Min	Error Max
iep	0.16	0.77	0.016	2.24
epl'	2.47	0.60	0.88	4.22
re	1.27	0.70	0.25	2.98

Error* is equal to difference between estimations (iep, epl', and re) and plp.

Table 3.2: Statistics Synopsis on Loss Probability Estimation for Different Estimators When plp is About -1.5.

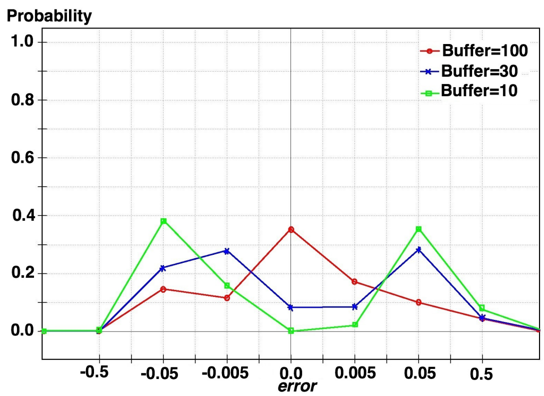
Estimator	Error Mean	Error Variance	Error Min	Error Max
iep	0.19	0.45	0.016	2.8
epl'	2.86	0.50	1.59	3.8
re	1.49	0.24	0.20	2.93

3.7 Conclusion

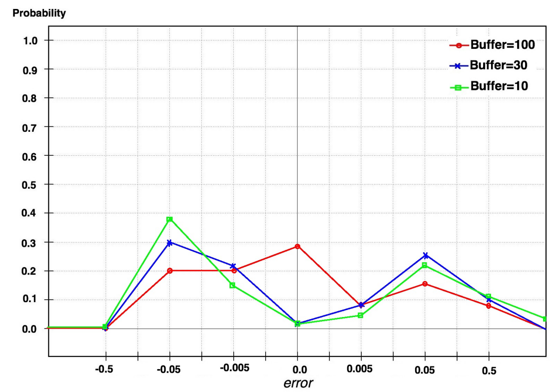
One of the most important issues in multimedia perceived quality is packet loss, which has an especially critical role in interactive communications. Accurate online network-based measurements of loss are necessary to give service providers the means to estimate the quality received by a user and to give them an opportunity to take remedial actions to satisfy the contractual SLA. Increased use of multimedia communications in the Internet has led to a renewed interest in the measure and estimation of loss, in the form of the plp, in modern communication networks. More specifically, recent studies have focused on estimation of the plp by measurement of input traffic based on LDT and the large buffer asymptote. In this chapter, we have reviewed the theory behind the finite buffer overflow probability (tail probability in infinite buffer) estimation. Based on *central limit theory*, by modelling the input traffic of an intermediate high speed node as a Gaussian process, we have introduced a new approximation for plp. Combining this online approximation with the offline output traffic measurement, we have proposed an accurate plp estimator which significantly improves the quality of the estimate compared to the recent proposed plp estimators which have used similar theoretical basis.

To study the accuracy of the estimates, we have used the NS-2 simulator with the input traffic which is very similar to the Internet traffic at the measurement node. Overall, the simulation results demonstrate the effect of different configurations, such as buffer size, on the estimates. The analysis of the results shows the improvement of accuracy in plp estimation achieved by our new calculation method.

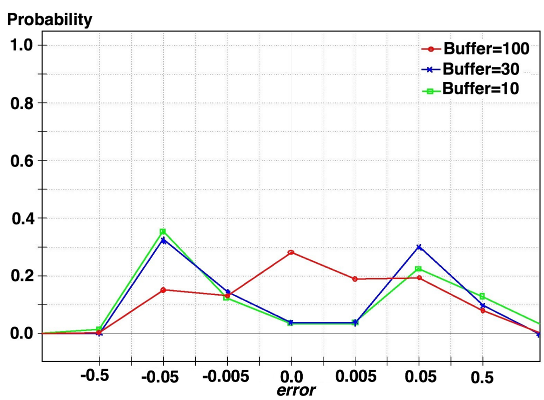
To conclude, the advantages of our proposed estimator are: 1) an increase in the accuracy of estimation by using the measured parameters properly, 2) flexibility on the duration of measuring time interval, and 3) an estimate of plp reasonably accurate in the case of a small buffer.



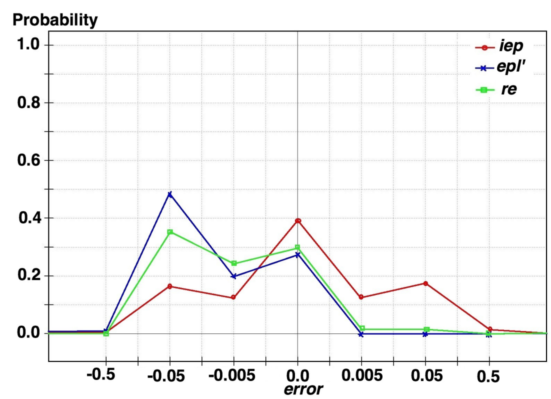
(a) PDF of $error$ for different buffer size when m is 5.



(b) PDF of $error$ for different buffer size when m is 10.



(c) PDF of $error$ for different buffer size when m is 20.



(d) PDF of $error$ for different estimators when buffer size is 100, m=5, and there is no random traffic.

Figure 3.9: The comparison of PDF of $error$ for different conditions.

Chapter 4

One-Way Delay Estimation

4.1 Introduction

We observe an ever-growing use of the Internet as the infrastructure of interactive multimedia communications. However, due to the "best effort" nature of the Internet, Quality of Service (QoS) is not predictable most of the time. Network and traffic engineers as well as service providers are interested in online measurements of QoS parameters to take appropriate actions to satisfy the contractual Service Level Agreement (SLA). Since the One-Way Delay (OWD) gives straightforward information about the state of the network (e.g., probability of congestion, loss ratio, available bandwidth, etc.), it is considered as the most valuable performance metric [30].

Halving the Round Trip Time (RTT) is the most common and simplest method to estimate the OWD. However, in the Internet, sending and receiving paths between two end users which are usually far from each other, are most often not symmetric. Moreover, most popular access technologies are intrinsically asymmetric [23–25]. Thus, the packets in each path encounter different types of queue conditions and also link performance. Therefore, deriving the OWD from the RTT cannot lead to an accurate measurement. Synchronization of two end nodes which are connected to the Internet network independently is another method for measuring the OWD by reading the time-stamp section in IP/RTP/UDP packet. Network Time Protocol (NTP), Global Position System, and the IEEE 1588 standard are among synchronization techniques used in special cases [29]. All of these solutions are not fully applicable or ubiquitous in Internet networks [30, 31], or do not resolve the issue of asymmetry.

A novel method for OWD measurement based on a cyclic-path method with least square error (LSE) has been presented in recent studies [128, 129]. In this chapter, a couple of additional useful constraints to the set of equations and variables employed in this method are proposed to improve precision in predicting the OWD. Further, the effect of different types of constraints on the estimation's outcome is examined. It is shown that the constraints which limit the OWD calculation region asymmetrically are more effective in accurate estimation of the OWD. In this regard, an accurate method is also introduced to measure the transmission delay which causes the asymmetric constraint. The cyclic-path considered in this study includes two main end user nodes as well as an auxiliary one.

The rest of the chapter is organized as follows. The chapter continues in Section 4.2 by reviewing prior bodies of work on measuring or estimating the OWD. In Section 4.3, the three-node model, the cyclic-path/LSE method, and the proposed improvements are explained. In Section 4.4, a method for measuring the transmission delay between two end-nodes is introduced. Simulations and numeric results demonstrate the improvement of the proposed model relative to other models in Section 4.5. The level of accuracy of the proposed method to measure the transmission delay is also demonstrated in that section. Section 4.6 concludes the chapter.

4.2 Previous work

Researchers define the OWD in different ways [29]. Its general definition is the difference between the occurrence time of sending the first packet bit and receiving the last packet bit at the destination node [130].

Studies focusing on OWD have either measured it by synchronizing the end nodes, or estimated it without clock synchronization. The OWD measurement is not possible if the destination and sender clocks are not synchronized [29]. The OWD_{ij} can be easily defined by the following expression for synchronized nodes [131]:

$$OWD_{ij}^{k} = t_{ij}^{k} - \Delta \tag{4.1}$$

where t_{ij}^{k} is the difference between the jth node's clock and the kth packet's time-stamp and Δ is the difference between the ith and jth nodes' clocks based on Universal Time Coordinate (UTC).

Time synchronization in a network can be accomplished in a number of ways, the most popular ones being: NTP [26], GPS, and the IEEE standard 1588 [28]. Using NTP as the synchronization protocol for measuring the OWD is not recommended for the Internet because of this protocol's symmetric paths assumption which may not always be met, especially in access networks. The accuracy of this method to synchronize the nodes in WANs is about 20 ms [130]. Based on the extant literature, some systems have used NTP for measuring the OWD. For instance, Smotlacha in [132] has conducted three setups with different NTP server locations to measure the OWD. He has shown that the average OWD measurement error has been improved to 8 ms.

GPS operations rely on the receivers (i.e., the nodes willing to be synchronized) communicating with 4 of the 24 satellites which exist for this purpose (three satellites are used to determine the position of receiver and the fourth is used to synchronize the clock of receivers) [27]. In spite of providing reliable clock synchronization with high accuracy in the order of less than 1 μs [133], the use of the special hardware and antenna is the dominant disadvantage of this method in synchronizing ordinary nodes in the Internet. Although IEEE standard 1588 provides an accurate synchronization (i.e., about 1 μs [130]), it requires special hardware and its applicability is restricted to distributed control system and LANs [134].

Given the downsides, estimating the OWD without clock synchronization is the subject of active research. Luong and Biro in [135] have devised a scheme to determine the OWD by measuring all adjacent node-pairs' clock offsets. This method needs a new service to be implemented in all intermediate routers so as to measure and return their relative neighbours' offsets. Tsuru et al. in [136] have suggested a method to improve the accuracy of clock offset estimation. However, they have assumed that transmission and propagation delays are the same in both directions which is only an approximation in the Internet.

Gurewitz et al. in [128, 129] have introduced a novel approach to estimate the OWD. Their approach is based on conducting multiple one-way measurements among a couple of nodes within closed loop paths. They have proposed an uncertain estimation of deterministic parts of OWDs. To achieve a relatively accurate estimation, a large number of network nodes have been used in their method. In this chapter, their estimation method is investigated for the case of only three nodes—a more feasible venture in real situations. Further, an accurate method for measuring the deterministic part of OWD is introduced which significantly improves the accuracy of cyclic-path method. The relative importance of different parts of the deterministic delay to limit the estimation domain and consequently to achieve the accurate estimate is also examined.

Another critical operation to obtain precise measurement of the OWD is detection, estimation, and removal of *clock skew* which has been subject to a large number of studies [137–143]. *Clock skew* is the difference between the speeds of two end-node clocks, and is crucial to assess when measurements take place at two distinct nodes and clock synchronization is considered. In our study, all measurements are performed in the same place; the successive measurements take place in a short time interval (i.e., around 200 ms); and the OWD estimation does not need synchronization. Moreover, the average clock skew in typical computers is around 1 part per million (ppm) [144] which would mean that the measurement error due to clock skew for a RTT of 270 ms (i.e., the longest RTT in the Section 4.5) would be around 0.27 μs; therefore it is proposed that the issue of clock skew is not relevant to the proposed estimation method.

4.3 OWD Estimation

4.3.1 Third-party model

The only accurate delay measurement between two independent (asynchronous) end users is an RTT measurement which can be conducted at both sides. System load issues notwithstanding, RTT measurements at both sides will be equal if clock skews are negligible for both nodes, i.e., they only amount to the drift that occurred while the measurement was made. In this situation, there is only one equation and two variables (i.e., OWD's of sending and receiving paths). Applying the LSE method to solve this equation leads to $OWD = \frac{RTT}{2}$ (RTT-halving). Assigning $\frac{RTT}{2}$ to OWD is not satisfactory in the Internet in which neither traffic load in both directions nor the transmission paths are symmetric [14, 145]. To increase the number of independent equations, an auxiliary node named "third-party node" is employed. Fig. 4.1 depicts an instance of the proposed third-party model in which we aim at estimating the OWD between two of the nodes (e.g., A and B). In this model, let t_{ij} denote OWD for the path from node i to node j. RTT_{ij} denotes the RTT between nodes i and j, so $RTT_{ij} = RTT_{ji}$. From a straightforward reading of Fig.

4.1, all independent equations obtained by measuring the delays are as follows:

$$\begin{cases} t_{ab} + t_{ba} = RTT_{ab} \\ t_{bc} + t_{cb} = RTT_{bc} \\ t_{ca} + t_{ac} = RTT_{ac} \\ t_{ab} + t_{bc} + t_{ca} = RTT_{abc}. \end{cases} \tag{4.2}$$

Note that each RTT is measured by one node; therefore asynchronous clocks of different nodes do not affect these measurements. The number of variables is always greater than the maximum number of independent cyclic-path delay measurements (i.e., independent equations) by $N - 1$, where N is the number of nodes [129]. Thus, the number of variables in the above mentioned case is greater than the number of equations by 2 (6 variables and 4 equations).

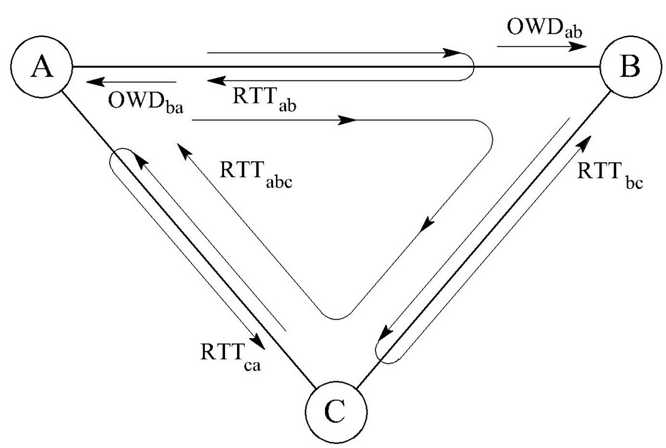


Figure 4.1: Third-party model.

4.3.2 More accurate estimation: Improvement of LSE with new constraints

Considering $\bar{t}_{ab}$ as the estimation of t_{ab}, it is desirable to minimize $(\bar{t}_{ab} - t_{ab})^2$. Using the LSE method, O. Gurewitz et al. in [128] have suggested to calculate the mean of all possible results of t_{ab} from (4.3) to achieve the best estimation of t_{ab}:

$$\begin{cases} t_{ab} + t_{ba} = RTT_{ab} \\ t_{bc} + t_{cb} = RTT_{bc} \\ t_{ca} + t_{ac} = RTT_{ac} \\ t_{ab} + t_{bc} + t_{ca} = RTT_{abc} \\ t_{bc} = x \qquad \text{with } 0 < x < \min(RTT_{bc}, RTT_{abc}) \\ t_{ca} = y \qquad \text{with } 0 < y < \min(RTT_{ac}, RTT_{abc}). \end{cases} \tag{4.3}$$

This set of equations simply adds straightforward constraints on the boundaries of the value of t_{bc} and t_{ca}, taking into account the possible asymmetry of link delays. The OWD consists of two parts: deterministic and stochastic. The deterministic part is divided into 1) propagation delay, 2) transmission delay, and 3) processing delay, while the stochastic part is due to queuing delay [146]. It is possible to estimate the range of the OWD's deterministic part for each path and at each direction; the propagation delay can be calculated based on the approximate geographical region of nodes according to IP addresses and speed of EM waves in the physical medium; the processing delay can be estimated based on codec type (e.g., frame size and lookahead size of the G.723.1 codec are 30 ms and 7.5 ms, respectively [147]), again assuming the measures are done at the application level. In Section 4.4, an accurate yet straightforward method for measuring the transmission delay is introduced.

Let t^c_{ij} denote the estimate of the deterministic part of the OWD for the path from node i to j. To achieve more precise results, (4.3) is modified by adding constraints on the range of the value of a couple of delays as follows:

$$\begin{cases} t_{ab} + t_{ba} = RTT_{ab} \\ t_{bc} + t_{cb} = RTT_{bc} \\ t_{ca} + t_{ac} = RTT_{ac} \\ t_{ab} + t_{bc} + t_{ca} = RTT_{abc} \\ t_{bc} = x \quad \text{and } t^c_{bc} < x < \min(RTT_{bc} - t^c_{cb}, RTT_{abc} - t^c_{abca}) \\ t_{ca} = y \quad \text{and } t^c_{ca} < y < \min(RTT_{ac} - t^c_{ac}, RTT_{abc} - t^c_{abbc}) \\ t^c_{ab} < t_{ab} < RTT_{ab} - t^c_{ba} \end{cases} \tag{4.4}$$

where t^c_{abca} and t^c_{abbc} are equal to $t^c_{ab} + t^c_{ca}$ and $t^c_{ab} + t^c_{bc}$, respectively. The estimated OWD for the path from node A to node B (i.e., $\bar{t}_{ab}$) is the mean of all possible results of t_{ab} from the equations set (4.4). Thus, $\bar{t}_{ab}$ can be calculated by:

$$\bar{t}_{ab} = RTT_{abc} - (\overline{t_{bc} + t_{ca}}). \tag{4.5}$$

According to (4.4), $\overline{t_{bc} + t_{ca}}$ is calculated based on the following constraints:

$$\begin{cases} t_{bc} + t_{ca} < RTT_{abc} - t^c_{ab} \\ RTT_{abc} - RTT_{ab} + t^c_{ba} < t_{bc} + t_{ca} \\ t^c_{bc} < t_{bc} < RTT_{bc} - t^c_{cb} \\ t^c_{ca} < t_{ca} < RTT_{ac} - t^c_{ac}. \end{cases} \tag{4.6}$$

Fig. 4.2 shows the region in which the mean of $t_{bc} + t_{ca}$ is calculated based on (4.6). It is obvious that tightening the constraints' region results in a more precise estimation.

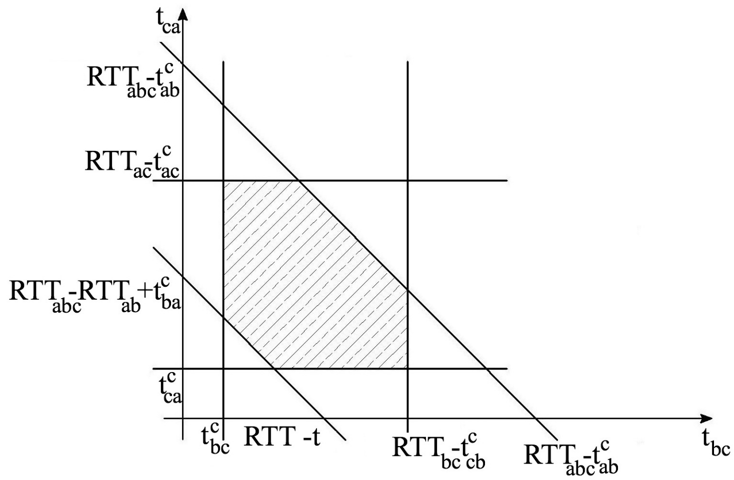


Figure 4.2: The region defined by (4.6).

4.3.3 Effect of different constraints on the improvement of estimation accuracy

As shown in the numeric results section, the *third party* model combined with LSE is sensitive to the values of the estimated deterministic delays. Recall that the deterministic delays in (4.6) (i.e., t^c_{ab}, t^c_{ba}, etc.) are divided into propagation, transmission, and processing delays. Propagation and processing delays are almost equal for backward and forward paths, so their estimation or measure limits the region of constraints symmetrically. In contrast, transmission delays are not equal due to the different end-users' access networks characteristics. Therefore, the boundaries of the $\overline{t_{bc} + t_{ca}}$ calculation region change asymmetrically with the measurement or estimation of different transmission delays for opposite paths. Given the low variation in the mean of $t_{bc} + t_{ca}$, when the constraints decrease the region of mean calculation (see (4.6)) symmetrically, $\bar{t}_{ab}$ from (4.5) does not follow the changes of constraints region significantly and remains almost constant for the tighter constraints.

Thus, measuring or estimating the propagation and processing delay may not lead to desirable accurate OWD estimates. On the contrary, asymmetric decrease of the region in which the mean of $t_{bc} + t_{ca}$ is calculated (see Fig. 4.2) causes more significant changes in the mean results and consequently in the final estimated OWD (i.e., $\bar{t}_{ab}$). Therefore, the more rigorous the transmission delay measurement the more accurate the OWD estimation.

4.4 The proposed transmission delay measurement method

Since the transmission delay is the most important portion of the deterministic part of the OWD which can add an asymmetric constraint to (4.6), its accurate measurement is vital. In this regard, several bandwidth estimation methods have been developed. The authors of [148] and [149] have used the packet pair technique to measure bottleneck bandwidth. In [104] and [150], analyzing a packet's RTT has been used to measure the link bandwidth for each hop. All these methods have assumed that the links between nodes are symmetric. Jiang in [151] has introduced an algorithm that can measure each hop's link bandwidth in both directions. Given that the only required data for estimating the OWD based on the cyclic-path/LSE method with constraints is the total transmission delay in each direction, a simple method is introduced, inspired by the aforementioned techniques, to measure the transmission delay for each path (i.e., the total of end-nodes' and middle hops' transmission delays). In addition to the simplicity of the proposed method, its ability to measure the transmission delay accurately is quite advantageous.

Fig. 4.3 shows the arbitrary link which connects two end-nodes. The synchronization of end-nodes is not required. To measure the transmission delay, node a starts sending packets of size B and D (bytes) to node b. The terminology used in the proposed measurement method is denoted as follows:

$T_a(B)_j$: departure time of the jth B bytes-sized packet from sender (i.e., node a), which is derived from the header's timestamp;

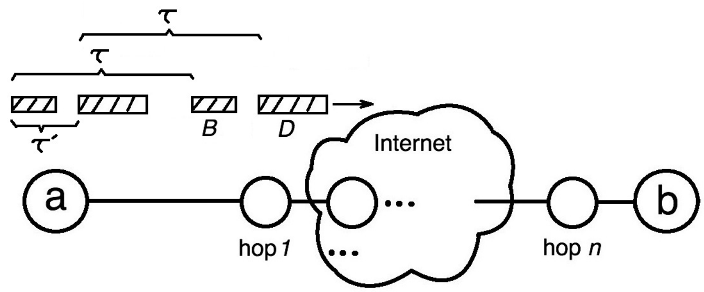


Figure 4.3: Two end-node connection with Internet.

$T_b(B)_j$: arrival time of the jth B bytes-sized packet at receiver (i.e., node b);

$T_a(D)_j$: departure time of the jth D bytes-sized packet;

$T_b(D)_j$: arrival time of the jth D bytes-sized packet;

$\Delta_b(B)_j$: forward delay of the jth B bytes-sized packet measured by node b; i.e., $T_b(B)_j - T_a(B)_j$;

$\Delta_b(D)_j$: forward delay of the jth D bytes-sized packet measured by node b; i.e., $T_b(D)_j - T_a(D)_j$; and

$t_{ab}^{trans}(I)$: transmission delay between nodes a and b for I bytes-sized packets.

Note that since nodes a and b are not synchronized, $\Delta_b(B)$ and $\Delta_b(D)$ do not have to be equal to OWD and their difference is due to the two end-nodes' clock offset and clock skew. To reduce the effect of the variation of the stochastic delay part, many packets of size B and D are sent and the means of forward delay measurements (i.e., $\overline{\Delta}_b(B)$ and $\overline{\Delta}_b(D)$) replace $\Delta_b(B)$ and $\Delta_b(D)$, respectively. It should be noted that similar to [137, 139–141], the proposed method has assumed that the clock skew for both nodes is constant.

Given

$$T_b(B)_0 = T_a(B)_0 + OWD_0(B) + \text{offset}_0 + OWD_0(B) \times Y \tag{4.7}$$

$$T_b(B)_1 = T_a(B)_1 + OWD_1(B) + (\text{offset}_0 + (Y - X)\tau) + OWD_1(B) \times Y \tag{4.8}$$

$$T_b(B)_j = T_a(B)_j + OWD_j(B) + (\text{offset}_0 + j(Y - X)\tau) + OWD_j(B) \times Y, \tag{4.9}$$

where offset_0 is the relative offset of the receiver clock with respect to the sender clock at the time of sending the first B bytes-sized packet; X and Y are the clock skews of nodes a and b relative to the *true clock* [139] (e.g., UTC); and τ is the interval between two consecutive same-size probes. The clock offset for two nodes at the time of sending the first D bytes-sized packet is $(\text{offset}_0 + (Y - X)\tau')$ where τ' is the interval between two consecutive B and D bytes-sized packets (Fig. 4.3). Therefore,

$$T_b(D)_j = T_a(D)_j + OWD_j(D) + (\text{offset}_0 + (Y - X)\tau' + j(Y - X)\tau) + OWD_j(D) \times Y \tag{4.10}$$

$$\Delta_b(B)_j = OWD_j(B) + (\text{offset}_0 + j(Y - X)\tau) + OWD_j(B) \times Y \tag{4.11}$$

$$\Delta_b(D)_j = OWD_j(D) + (\text{offset}_0 + (Y - X)\tau' + j(Y - X)\tau) + OWD_j(D) \times Y \tag{4.12}$$

$$OWD_j(I) = t_{ab}^{trans}(I) + c_j \tag{4.13}$$

$$\overline{OWD}(I) = t_{ab}^{trans}(I) + \lim_{N\to\infty} \frac{1}{N} \sum_{j=0}^{N} c_j = t_{ab}^{trans}(I) + \overline{c} \tag{4.14}$$

$$\overline{\Delta}_b(B) = t_{ab}^{trans}(B) + \overline{c} + \text{offset}_0 + \frac{N+1}{2}(Y - X)\tau + \overline{OWD}(B) \times Y \tag{4.15}$$

$$\overline{\Delta}_b(D) = t_{ab}^{trans}(D) + \overline{c} + \text{offset}_0 + (Y - X)\tau' + \frac{N+1}{2}(Y - X)\tau + \overline{OWD}(D) \times Y \tag{4.16}$$

$$t_{ab}^{trans}(I) = \sum_{i=0}^{m} \frac{I}{v_i} = I \sum_{i=0}^{m} \frac{1}{v_i} = \alpha \times I, \tag{4.17}$$

where $\overline{c}$ consists of the mean of other parts of the OWD, N is the number of B or D bytes-sized probes in the measurement period, m is the number of hops between two end-nodes, and v_i denotes the speed of link between hop $i+1$ and hop i. Hence,

$$\overline{\Delta}_b(D)-\overline{\Delta}_b(B) = t_{ab}^{trans}(D)-t_{ab}^{trans}(B)+(Y-X)\tau'+(t_{ab}^{trans}(D)-t_{ab}^{trans}(B))Y. \quad (4.18)$$

Substituting (4.17) into (4.18) results in

$$\overline{\Delta}_b(D)-\overline{\Delta}_b(B) = \alpha\times(D-B)(Y+1)+(Y-X)\tau' = (Y+1)t_{ab}^{trans}(D-B)+(Y-X)\tau'. \quad (4.19)$$

Considering an average clock skew in computers of typically around 1 ppm [144], the transmission delay can be accurately approximated by

$$t_{ab}^{trans}(D-B) = \overline{\Delta}_b(D)-\overline{\Delta}_b(B). \quad (4.20)$$

The accuracy of (4.20) is verified in Section 4.5 by including an arbitrary clock skews in the simulations.

To investigate the accuracy of the results as well as the effects of different constraints on OWD estimation, a number of simulations are conducted and their numeric results and evaluations are presented in the next section.

4.5 Simulations and numeric results

To calculate the OWD in different situations, a program has been created using the MATLAB software. The results of the improved model are compared to the model proposed by [128]. The effects of different constraints on OWD estimation are examined as well.

For instance, the OWDs for the three bidirectional links in Fig. 4.1 are:

$$\begin{aligned} \text{OWD}_{ab} &= 100\,\text{ms} \quad & \text{OWD}_{ba} &= 110\,\text{ms} \\ \text{OWD}_{bc} &= \ \ 70\,\text{ms} \quad & \text{OWD}_{cb} &= 110\,\text{ms} \\ \text{OWD}_{ac} &= \ \ 70\,\text{ms} \quad & \text{OWD}_{ca} &= \ \ 90\,\text{ms}. \end{aligned}$$

The goal is to estimate the OWD for the path from node A to node B. All the essential measurements for (4.3) and (4.4) are:

$$\begin{aligned} RTT_{ab} &= 210\,\text{ms} \\ RTT_{bc} &= 180\,\text{ms} \\ RTT_{ac} &= 160\,\text{ms} \\ RTT_{abc} &= 260\,\text{ms}. \end{aligned}$$

The estimated deterministic parts of OWD's are as follows:

$$\begin{aligned} t^c_{ab} &= t^c_{ba} = 70\,\text{ms} \\ t^c_{bc} &= t^c_{cb} = 60\,\text{ms} \\ t^c_{ca} &= t^c_{ac} = 40\,\text{ms}. \end{aligned}$$

The estimated $\bar{t}_{ab}$s based on the RTT-halving method, the cyclic-path model, and the cyclic-path with constraints model which is proposed in this chapter are 105 ms, 98.31 ms, and 100.29 ms, respectively. The constraints which are used to limit the region of OWD estimation are symmetric (e.g., $t^c_{bc} = t^c_{cb}$). Therefore, there is no significant difference between the OWD estimated by the cyclic-path/LSE method with and without constraints. This inefficiency is more obvious for the case where the cyclic-path method without any constraints cannot estimate the OWD reasonably. Consider, for example, the case in which the OWD and RTT delays for an arbitrary three-nodes network in Fig. 4.1 are:

$$\begin{aligned}
&\mathrm{OWD}_{ab} = 140\,\mathrm{ms} \quad \mathrm{OWD}_{ba} = 130\,\mathrm{ms} \\
&\mathrm{OWD}_{bc} = \ \ 90\,\mathrm{ms} \quad \mathrm{OWD}_{cb} = 100\,\mathrm{ms} \\
&\mathrm{OWD}_{ac} = \ \ 45\,\mathrm{ms} \quad \mathrm{OWD}_{ca} = \ \ 40\,\mathrm{ms} \\
&RTT_{ab} = 270\,\mathrm{ms} \\
&RTT_{bc} = 190\,\mathrm{ms} \\
&RTT_{ac} = \ \ 85\,\mathrm{ms} \\
&RTT_{abc} = 270\,\mathrm{ms}
\end{aligned}$$

and the estimated deterministic parts of OWD's (i.e., symmetric constraints) are:

$$\begin{aligned}
&t^c_{ab} = t^c_{ba} = 60\,\mathrm{ms} \\
&t^c_{bc} = t^c_{cb} = 40\,\mathrm{ms} \\
&t^c_{ca} = t^c_{ac} = 10\,\mathrm{ms}.
\end{aligned}$$

The estimated $\bar{t}_{ab}$s based on the cyclic-path method with and without constraints are 133.16 ms and 132.60 ms, respectively, which are worse than the result of the RTT-halving model.

As shown in Fig. 4.4, the symmetric decrease of the calculation region for $\overline{t_{bc} + t_{ca}}$ for this case does not move the mean point, so adding the symmetric constraints cannot help the cyclic-path method to improve the accuracy of the estimation significantly.

On the other hand, measuring the transmission delay for all paths adds asymmetric constraints to the cyclic-path method's equation set. Although the constraint tightening which is caused by the measurement of the transmission delay is small compared to the propaga-

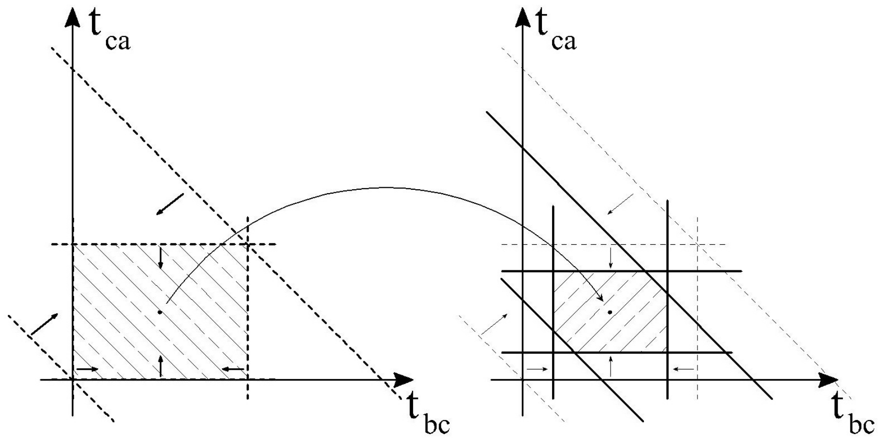


Figure 4.4: The effect of symmetric constraints on $\overline{t_{bc} + t_{ca}}$.

tion and process delay constraints, the effect of the transmission delay measurement on the accuracy of OWD estimation based on the cyclic-path method is more effective than that of other symmetric constraints. For instance, in the previous example, if the forward (upload) bandwidths of three nodes are different (e.g., $BW_a < BW_c < BW_b$, where BW_i is the forward bandwidth of node ith), and the estimated or measured deterministic parts of the delays are as follows:

$$t^c_{ab} = 17\,\text{ms} \qquad t^c_{ba} = 6\,\text{ms}$$
$$t^c_{bc} = 7.5\,\text{ms} \qquad t^c_{cb} = 19\,\text{ms}$$
$$t^c_{ca} = 3\,\text{ms} \qquad t^c_{ac} = 4.5\,\text{ms},$$

then the estimated $\bar{t}_{ab}$ based on the cyclic-path method with constraints will be $139\,\text{ms}$. Note that the mentioned asymmetric deterministic parts of delays in the above example are selected equal to the transmission delays of each path, based on the simulation outcomes explained at the end of this section. Real data, gathered from residential ADSL nodes located at Canada, Middle East, and Europe, has been utilized in the simulations for nodes

a, b, and c, respectively.

To present an overall view of the propositions about the effect of different types of constraints on the estimation outcomes, the previous example network is simulated and different types of constraints are examined. Fig. 4.5 depicts the effect of different symmetric constraints on the final estimation. This figure reveals that more tightened symmetric limitations result in the estimated OWD to be almost constant.

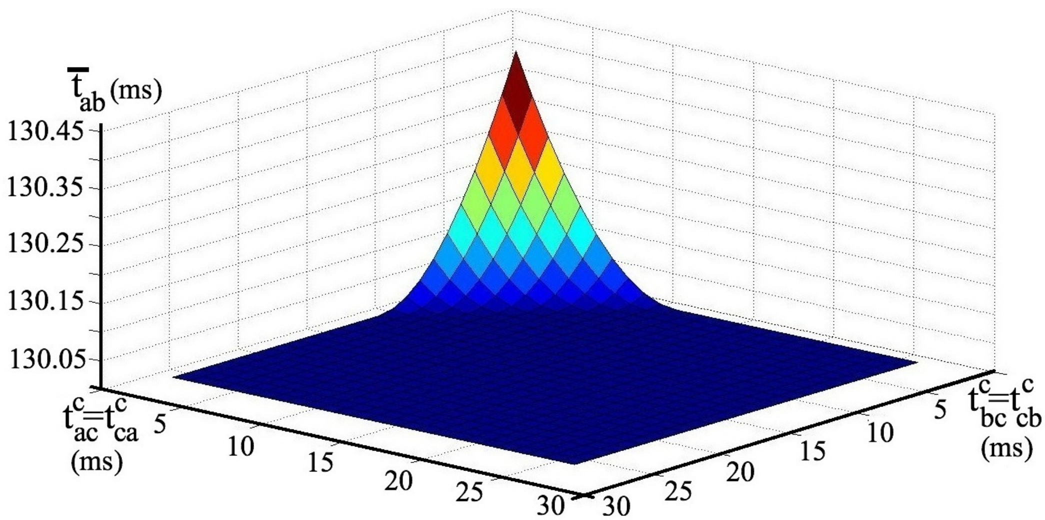


Figure 4.5: The effect of symmetric constraints (propagation and process delays) on $\bar{t}_{ab}$ (i.e., estimated OWD).

Fig. 4.6 shows that measurement or estimation of transmission delays as the asymmetric part of deterministic delay influences the final estimated OWD more effectively compared to other deterministic delays. It means that the results of the cyclic-path method are different for various amounts of transmission delays , whereas symmetric constraints do not affect the results in the same manner.

As mentioned in Section 4.2, Gurewitz et al. in [129] have proposed a method for estimating the deterministic parts of the OWD by looking at the last n packets which traverse the link from node A to node B and vice versa. The smallest Δ_B^{min} and Δ_A^{min} are selected

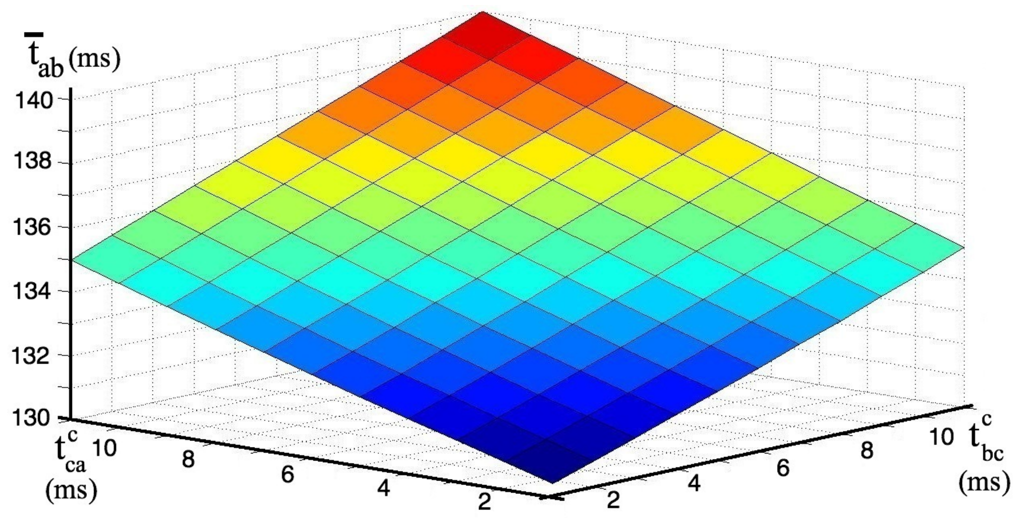


Figure 4.6: The effect of asymmetric constraints (transmission delay) on $\bar{t}_{ab}$ (i.e., estimated OWD).

as the t^c_{ab}+offset and $t^c_{ba}-$offset, respectively. The constant RTT can then be calculated as:

$$t^c_{ab} + t^c_{ba} = \Delta^{min}_A + \Delta^{min}_B. \tag{4.21}$$

By repeating this process to estimate the deterministic part of other cyclic paths and applying the LSE method, the deterministic part of OWD can be estimated for each path. This estimation may give accurate results only when the nodes are close to each other, without a large number of hops in between. To prove this statement, the RTT for two nodes located in Canada and the Middle East is measured 1000 times. The smallest measured RTT was 243 ms which cannot be the deterministic part of RTT considering the propagation speed, distance, and download and upload bandwidths.

The accuracy of the proposed method, using the estimated deterministic part of the OWD, is compared to the one used in Gurewitz et al.'s [129] by conducting a simulation based on Fig. 4.7. In this simulation, nodes 2, 4, and 6 are designed to generate the common Internet traffic flow for background traffic and make the aggregated traffic situation similar

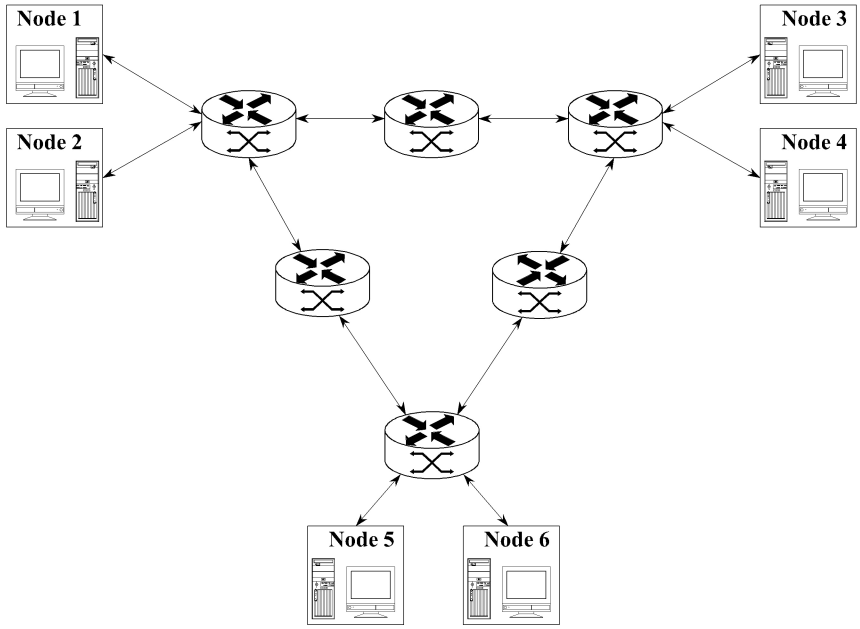


Figure 4.7: Testbed topology for cyclic-path simulation.

to that of a real Internet network traffic. The Tmix module in NS-2 is utilized in nodes 2, 4, and 6 in order to generate realistic Internet network traffic [126]. All links' delays and bandwidths are set based on the real measured data for three residential Internet nodes located in Canada, the Middle East, and France. The intermediate nodes' buffer size were set so that the loss ratio is less than one percent (i.e., normal for Internet network) and averages of OWDs over 120 s measurement intervals are as follows:

$$\text{OWD}_{ab} = 140\,\text{ms} \qquad \text{OWD}_{ba} = 130\,\text{ms}$$
$$\text{OWD}_{bc} = 90\,\text{ms} \qquad \text{OWD}_{cb} = 100\,\text{ms}$$
$$\text{OWD}_{ac} = 45\,\text{ms} \qquad \text{OWD}_{ca} = 40\,\text{ms}$$

Each of nodes 1, 3, and 5 sends 20 packets per second to the other two. The size of each

Table 4.1: Simulation detail results for the path from node A to node B

Estimation method	Min(t_{ab})	Max(t_{ab})	Mean(t_{ab})	$\overline{t_{ab}^{c}}$	$\overline{t_{ab}}$
RTT-halving	97 ms	371 ms	140 ms	-	135 ms
Cyclic-path without constraint	97 ms	371 ms	140 ms	-	132.60 ms
Cyclic-path with [129]'s constraints	97 ms	371 ms	140 ms	90.75 ms	136.11 ms
Cyclic-path with proposed constraints	97 ms	371 ms	140 ms	53 ms	139 ms

[1] $\overline{t_{ab}^{c}}$ and $\overline{t_{ab}}$ are estimated deterministic part of OWD_{ab} and estimated OWD_{ab}, respectively.
[2] $\overline{t_{ab}}$s are estimated based on the average RTTs.

probe packet is 250 bits.

Table 4.1 shows the detailed results for the path from node A to node B based on different OWD estimators. It should be noted that $\overline{t_{ab}^{c}}$, which is estimated based on the method in [129], is the estimation of Min(t_{ab}) which is not, in itself, an effective constraint. The $\overline{t_{ab}^{c}}$, estimated based on our proposition, is equal to the sum of the measured transmission delay (i.e., 17 ms which is the outcome of next simulation) and the propagation delay (i.e., 36 ms for a distance of 10,000 Km between Canada and the Middle East).

The rest of this section presents the simulations conducted for investigating the accuracy and applicability of the transmission delay measurement method introduced in Section 4.4 to overall OWD measurement. The NS-2 software [125] is used to simulate a network; the network topology which is simulated is shown in Fig. 4.8.

The goal is to measure the transmission delay of the link between nodes 1 and 2. Node 1 generates two types of 25 and 50 bytes packets 10 times per second, which does not add a significant load into the network. These packets are transferred to node 2. Nodes 3 and 4 are designed to generate the common Internet traffic flow for background traffic and make the aggregated traffic situation similar to that of a real Internet network traffic.

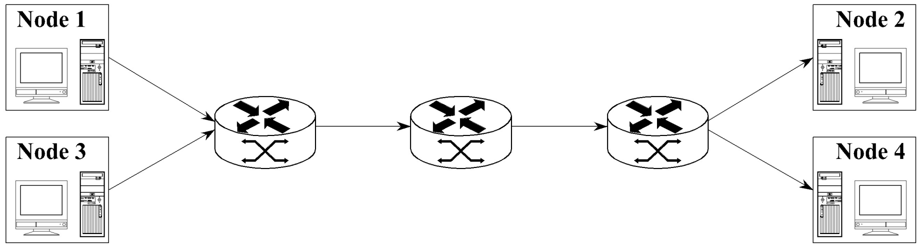
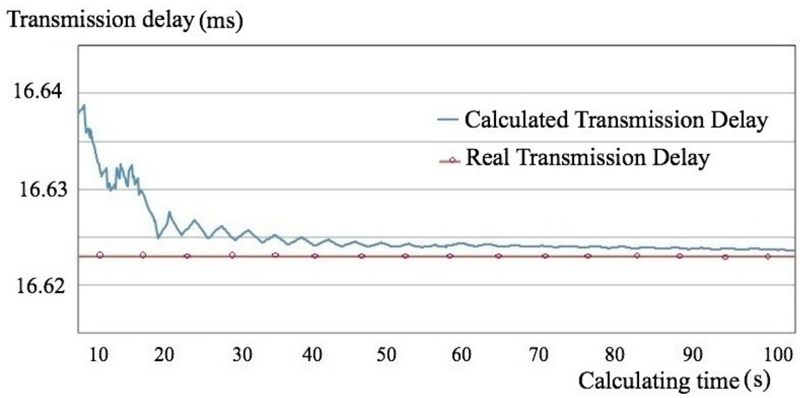


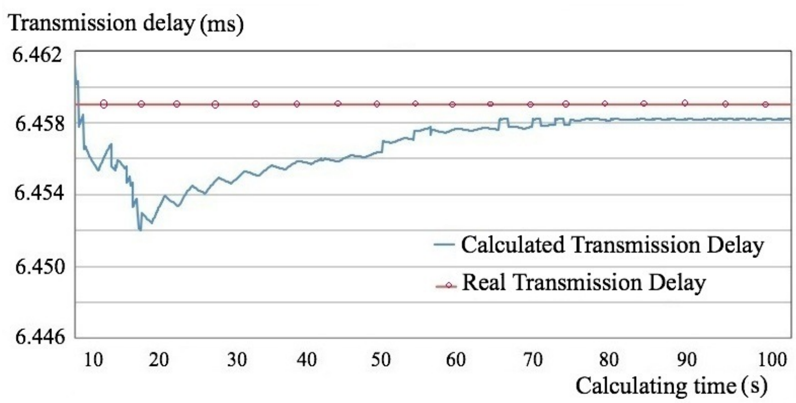
Figure 4.8: Testbed topology for transmission delay measurement simulation.

The Tmix module in NS-2 is utilized in nodes 3 and 4 in order to generate realistic Internet network traffic [126]. The upload and download bandwidths of node 1 are set to 940 Kbps and 5.87 Mbps, respectively (i.e., the real measured values for a residential ADSL in Montreal, Canada). The upload and download bandwidths of node 2 are set to 123 Kbps and 464 Kbps, respectively (i.e., the real measured values for a residential ADSL in a city in the Middle East). To illustrate that the clock skew indeed does not noticeably affect the accuracy of the proposed transmission delay measurement method, clock skews of +10 ppm and -15 ppm are added to nodes 1 and 2, respectively, in the simulation. The results of this simulation are used in the aforementioned example for OWD estimation with asymmetric constraints. Fig. 4.9 shows the accuracy of the proposed method to calculate the G.711 codec's packets' transmission delay between nodes 1 and 2 for different calculation times (packet size is 250 bytes). This figure expresses that the error of calculation is less that 1 percent, which even decreases over time. An increase in the measurement time (i.e., increase of N in (4.14), (4.15) and (4.16)) results a decrease in a difference between $\overline{c}$ in (4.15) and (4.16) and its true mean in (4.14). Hence, as demonstrated in Fig. 4.9, the estimation of transmission delay converges to its true amount over time. It should be mentioned that the amount of error caused by neglecting the clock skews would be less than 0.5 μs which is not noticeable in the test outcomes.

These simulations illustrate the impressive level of accuracy and applicability of the proposed method in measuring the transmission delay. Therefore, considering the precision and simplicity of measuring the transmission delay along with the propagation and processing delay measurements, the OWD estimation based on the cyclic-path method becomes more reliable.



(a) Transmission delay from node 1 to node 2;



(b) Transmission delay from node 2 to node 1;

Figure 4.9: Forward and backward calculated transmission delay for the G.711 packets (i.e., 250 bytes).

4.6 Conclusion

The one-way delay is a most valuable performance metric because it gives straightforward information about the network, such as congestion probability, loss ratio, and available bandwidth. In this chapter, a recently proposed method for estimating the OWD has been analyzed and improved upon.

This method is based on conducting multiple RTT measurements among pairs of nodes and applying the LSE method to find the more reasonable estimate of the OWD between two specific nodes. This is done by measuring all possible independent RTTs between them and one auxiliary third-party node. To secure more accurate estimates, some additional constraints have been introduced to the regular cyclic-path's set of equations. All the new constraints can be easily estimated based on the known behavior of nodes and their connection paths. To measure the transmission delay, which is often an asymmetric constraint, an accurate method has been introduced.

To compare the accuracy of the results obtained by the proposed method, with cyclic-path, and traditional RTT-halving models, a simple three-node network was investigated in different situations. The results have confirmed the improvement of estimation errors in the proposed model. Furthermore, the influence of different types of constraints on cyclic-path method has been examined. It was demonstrated that the asymmetric constraints are more effective in improving the results than the symmetric ones and how they can be effectively estimated.

The contributions of our research in this chapter can be summarized as follows. First, we have investigated the effects of parameters which impose different types of constraints upon the cyclic-path's equation set on the accuracy of the OWD estimation. Second, we have introduced a straightforward method to accurately measure the transmission delay. Third, we have shown that the OWD between faraway nodes with a couple of unknown hops in-between can be estimated more accurately by employing our method to measure the transmission delay, whereas the method proposed in [129] cannot accurately estimate a constant delay between nodes which are far apart. Finally, the methods discussed in this chapter are free from clock skew awkwardness.

Chapter 5

Loss Effects on Video Quality

5.1 Introduction

Packet loss is one of the key factors in determining the quality of transmitted video over the Internet. Further, to reduce storage space and to transmit video over bandwidth–limited networks, compression of video bitstream is an essential issue. To compress video data, the H.264 codec [152]—the state of the art in video compression—employs, among other techniques, spatial transforms and motion compensated prediction between consecutive frames to exploit spatial and temporal redundancy, respectively. Despite indisputable benefits in compression, the resulting high vulnerability of compressed video to data loss has become an overwhelmingly important issue.

Error propagation is a critical problem afflicting transmitted compressed video over error-prone channels [153]. Indeed, dependency of each coded frame to previous frames' data propagates the error to subsequent frames. Hence, error resilient video communication has been subject to extensive studies and improvements in this field have been significant. Example of well-investigated proposed solutions in this field are Forward Error Correction (FEC) [154], path diversity with multiple description video [155, 156], packet interleaving

[153], and Intra/Inter-mode switching [157]. The accuracy of distortion prediction caused by packet/frame loss is the main issue in these methods.

Using the average $PSNR$ as a measure of perceptual quality of video [71], we investigate the effect of frame loss position relative to I-frames on total distortion for the videos and propose a model to estimate the $PSNR$ of the received frames impaired by distortion propagation. We derive from this model a linear relationship between the average $PSNR$ and the relative position of the lost frame w.r.t. the last I-frame. The accuracy of the model is validated by simulations of the transmission of three standard video sequences coded with the H.264/AVC codec over an error-prone channel. We also propose a simple packet transmission schedule for improving the performance of noisy channels, and verify its efficiency through calculating the probability of different lengths of burst loss.

The rest of this chapter is organized as follows. The chapter continues in Section 5.2 by reviewing prior models for estimating the distortion produced by packet loss. In Section 5.3, we describe the effect of the position of lost frame relative to I-frames on average $PSNR$ and derive a model that estimates the total propagated distortion. The accuracy of model estimations is demonstrated via simulations in Section 5.4. The effect of packet transmission scheduling on noisy channel performance is examined in Section 5.5. Section 5.6 concludes this chapter.

5.2 Previous work

Earlier work on modelling the effect of loss generally model the distortion as being eliminated by the next I-frame [33, 34, 153]. For example [153] has modelled the error propagation with a linear attenuation factor resulting from Intra update as well as a geometric attenuation factor from spatial filtering. As will be seen in next sections, this model does not have an acceptable correlation with experimental results in case of the medium and high motion videos, beside the fact that it also cannot accurately model the distortion when a single loss happens for a low motion video.

Färber et al. in [33, 34] have proposed a theoretical framework to model the influence of the errors on video quality. A reciprocal attenuation factor models the effect of spatial filtering, derived from calculating the signal power spectrum density. This model also assumes that distortion only continues until next I-frame, whereas it is shown in this chapter that the contrary is true.

In [32], an experimental approach was used to model the 1 bit error propagation in case of using data partitioning as a resilience tool in MPEG-4 video. They investigated CIF-size videos. Tan et al. in [35] have also studied the temporal propagation of small errors in a single frame for H.264 video. Yang et al. in [36] have used Expected Length of Error Propagation (ELEP) as a simple performance metric to present an unequal packet loss resilience scheme. However, their model is not accurate enough to estimate the error propagation and video quality degradation under packet loss.

In [158], Maugey et al. have proposed a theoretical model for the error propagation generated by a frame loss in a Distributed Video Coding (DVC) framework. Using rate-distortion functions, they have analyzed the impact of a frame loss on the average distortion of a group of pictures depending on the role of the lost frame within the GOP. Sun et al. [159] have proposed a Frame Error Propagation Index (FEPI) to model the frame significance in GOP. They have used their model to propose a loss protection scheme.

Most of the above distortion models only consider the average loss rate in the absence of another factor, i.e., the burst length, and therefore are less efficient for the case of video transmission over burst-loss channels. In addition to covering this shortcoming, we also focus on QCIF videos coded by H.264 which encounter packet/frame losses.

5.3 Effect of frame loss position on distortion

We propose a model for estimating the average $PSNR$ of a constructed video encountered with single or burst losses. To develop this model, we consider low (*Bridge-Close*), medium (*News*), and high motion (*Football Game*) videos, coded with H.264 using reference software JM 17.1. The resolution size, frame rate, and Quantization Parameter (QP) of all video sequences are QCIF (176×144), 30 fps, and 28, respectively, as QCIF remains heavily used in the Internet. The first coded frame is Intra-coded (I), followed by Inter-coded frames (P), and Intra updates occur after every 30 P-frames. To conceal the error, the lost frame is replaced with the last correctly received frame. The codec wraps each P-frame of low and medium motion videos into a single packet of varying size. However, each frame of the high motion video is packetized into two packets on the average.

5.3.1 Definitions

If a packet is lost during transmission, a common method for error concealment on the decoder side is to replace the lost data by the last correctly received frame data. In this case, let $\tilde{p}_n^i$ denote the i^{th} pixel of the n^{th} reconstructed frame at the decoder, and p_n^i denote the i^{th} pixel of the n^{th} original coded frame at the encoder. The total frame error at frame n is defined as

$$e_n = \sum_{i=1}^{M} (\tilde{p}_n^i - p_n^i), \tag{5.1}$$

where M is the number of pixels in each frame (i.e., $176 \times 144 = 25344$ in QCIF video). The Mean Square Error (MSE) associated with frame error e_n is given by

$$d_n = E[(\tilde{p}_n^i - p_n^i)^2] = \frac{1}{M} \sum_{i=1}^{M} [(\tilde{p}_n^i - p_n^i)^2]. \tag{5.2}$$

Since the codec employs motion compensation and inter-prediction to encode consecutive frames, distortion propagates to subsequent frames. Thus, the total distortion for a single lost frame at n is

$$D_n = \sum_{i \geq n} d_i. \tag{5.3}$$

The $PSNR$ of the video signal of frame n is given by

$$PSNR_{dB}[n] = 20 \log_{10}(\frac{V_{peak}}{RMSE}) = 20 \log_{10}(\frac{V_{peak}}{\sqrt{d_n}}), \tag{5.4}$$

where V_{peak} is the maximum possible pixel value of the frame and $RMSE$ is the root mean square error between received and original frames [71].

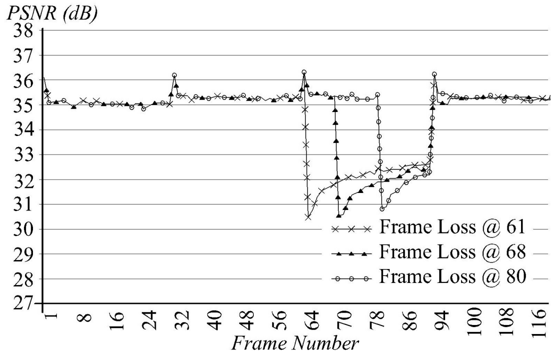
5.3.2 Empirical observations

As mentioned above, three types (low, medium, and high motion) of video sequences are investigated.

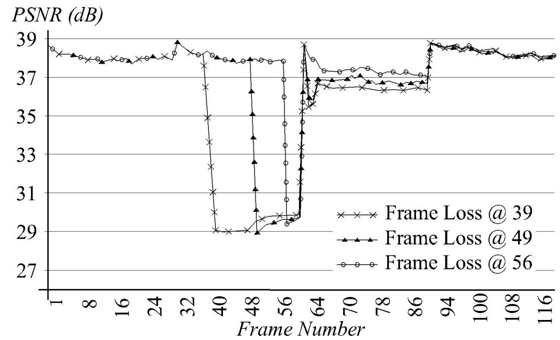
Fig. 5.1 shows the frames' $PSNR$ when a random single loss occurs, for all three types of video. Distortion propagation for each type of video is clearly demonstrated in this figure. Maximum distortion occurs at the lost frame point and the distortion of subsequent frames is attenuated by spatial filtering. Intra updates eliminate the effect of distortion propagation at the subsequent I-frames. In the case of medium and high motion videos, it is observed that although the Intra update plays a key role in correcting the distortion, it cannot completely prevent the propagation of errors to other P-frames, due to high level of compression and dependency between frames within the H.264 codec.

5.3.3 Estimation model

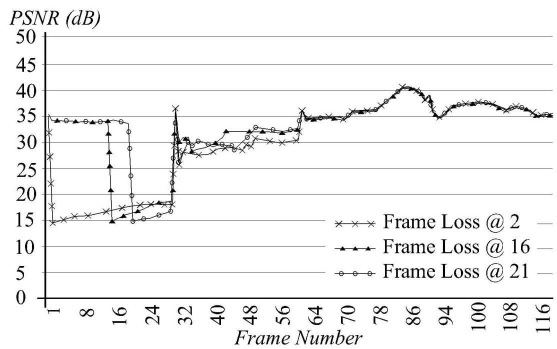
Based on our observations of error propagation behaviour for all three types of video, we model the distortion propagation process with an exponential attenuation factor resulting from spatial filtering and a conditional attenuation factor from Intra update of I-frames.



(a) $PSNR$ of *Bridge-Close* video which encounters a single loss at frame 62, 68 and 80.



(b) $PSNR$ of *News* video which encounters a single loss at frame 39, 49 and 56.



(c) $PSNR$ of *Football Game* video which encounters a single loss at frame 2, 16 and 21.

Figure 5.1: $PSNR$ of three types of video which encounter a single frame loss.

This model estimates the average $PSNR$ of the specific period in which the loss occurs. With an Intra update period of M, if the frame loss occurs at n, the $PSNR$ for the $(n+l)^{th}$ P-frame will be

$$PSNR[n+l] = PSNR[n] + \beta\left(\overline{PSNR} - PSNR[n]\right)\left(1 - e^{-\alpha l}\right) \tag{5.5}$$

where $\overline{PSNR}$ is the average $PSNR$ of the encoded video without any loss. α and β ($\alpha, \beta < 1$) account for the effects of spatial filtering and Intra update, respectively. Since α depends on the strength of the spatial loop filter of the codec and the power spectrum

Table 5.1: Suggested values of α for three types of video with different bit rates

	Low Motion *Bridge-Close*	Medium Motion *News*	High Motion *Football Game*
$BitRate(bps)$	25000	40000	532000
α	0.4	0.3	0.1

Table 5.2: Suggested amounts of β for three types of video based on the temporal distance between the examined frame and the lost frame ($cM^* < n$)

β	Low Motion *Bridge-Close*	Medium Motion *News*	High Motion *Football Game*
cM* $< n + l <$c(M+1)	1/3	1/6	1/6
c(M+1)$< n + l <$c(M+2)	1	3/4	2/3
c(M+2)$< n + l <$c(M+3)	1	1	3/4
c(M+3)$< n + l$	1	1	1

$^*c$ is a positive integer and M is the Intra update period.

density of the input error signal, we assume that it is constant for the entire recovery period, and independent of frame index n [37, 153]. Tables 5.1 and 5.2 list our proposed values for α and β, obtained from the aforementioned empirical tests for each of the three types of video.

To quantify the amount of scene motion, various indices, such as Mean of Absolute Difference (MAD) or Sum of Absolute Difference (SAD), or their combinations can be used [160]. Hence, α and β selection can be performed automatically based on one of these indices. Measuring these indices are beyond the scope of our research, however.

Prior studies have employed a geometric attenuation factor (α'^l) [153] or $1/(1 + \alpha'' l)$ [33, 34] to show the effect of spatial filtering on MSE, and a $(1 - \frac{l}{M})$ attenuation factor to model the effect of Intra update on MSE. The simulations presented in Section 5.4 show that our proposed model is more accurate in estimating the quality of the reconstructed video.

According to (5.5), the $PSNR$ (and distortion) of subsequent frames of the lost frame depends on the lost frame $PSNR$ and its position. Therefore, we can conclude that the total distortion (D_n), as well as the average $PSNR$, depend on the position of the frame loss relative to the last or next I-frame. Furthermore, the numeric simulation results presented in Section 5.4 also support our conclusion that the average $PSNR$ is dependent on the index of n^{th} lost frame, whereas, to the best of our knowledge, previous studies have not explicitly considered the relationship between D_n and n, or may have implicitly assumed that D_n merely depends on d_n and is independent of frame index n [37, 153, 161, 162].

For $x < n < y$ ($y - x$ is large enough to assume $e^{-\alpha(y-n)} \to 0$), the average $PSNR$ between x^{th} and y^{th} frames is given by

$$E[PSNR] = \frac{1}{y - x + 1} \sum_{i=x}^{y} PSNR[i], \tag{5.6}$$

where $PSNR[i]$ is derived from (5.5) for $i > n$. Putting (5.5) into (5.6) and assuming that $PSNR[n]$ is approximately constant for any P-frame loss between two consecutive I-frames, as observed in Fig. 5.1, (5.6) becomes

$$E[PSNR] = An + B, \tag{5.7}$$

where A and B depend on α, β, $PSNR[n]$ of the lost frame, and the average $PSNR$ when there is no loss.

In the next section, we verify the validity of (5.7) by conducting simulations and show how the (A, B) parameters are computed.

Table 5.3: Average $PSNR$ error (dB) for a single P-frame loss, given by proposed and geometric models

Sequence	Low Motion *Bridge-Close*	Medium Motion *News*	High Motion *Football Game*
Proposed Model Error (dB)	0.010	0.005	0.013
Geometric Model Error (dB)	3.65	3.134	1.137

5.4 Simulation and Numeric Results

We have conducted simulations to investigate the accuracy and applicability of our proposed model for calculating the total distortion and average $PSNR$, and compare it with the models proposed by [153]. NS-2, a standard tool in network research, is employed to simulate the IP network testbed for transmission and reception as well as the generation of different loss patterns at specific times and positions [163]. The RTP/UDP/IP stack is used for media transmission.

Fig. 5.2 compares the accuracy of the $PSNR$ estimated by the proposed model with that obtained by the prior geometric method for the low motion (*Bridge-Close*) video. The improvement of estimation accuracy is more obvious in Fig. 5.3 which compares the proposed model with a geometric method to estimate the distortion for the medium motion (*News*) video. The high motion video type also shows similar trends. As we see in this figure, the proposed model estimates the $PSNR$ more accurately than the previous estimation method. Part of this difference can be due to the assumption made in previous studies to stop distortion propagation at the first subsequent I-frame. As observed from our simulations (Fig. 5.1), this is not a comprehensive assumption for the videos coded with H.264, especially for the medium and high motion videos. Hence, to achieve a higher level of accuracy, propagation of the distortion through subsequent P-frames, i.e., after the first subsequent I-frame, should also be considered. This observation can also be explained by our selection of attenuation factors as the spatial filtering and Intra update correction, knowing that in highly efficient codecs such as H.264, the dependency of consecutive frames is high enough to keep the degradation of subsequent P-frames' quality at a noticeable level. Table 5.3 lists the average errors of predicting the $PSNR$ by both methods and presents the precision of our proposed model for quality estimation compared to the geometric model.

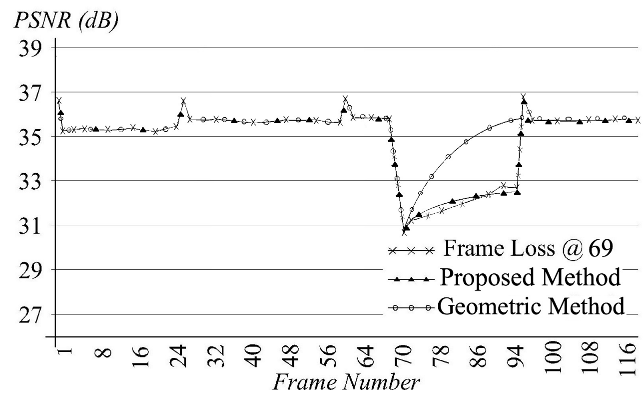


Figure 5.2: Accuracy of proposed vs. geometric methods to estimate the $PSNR$ for subsequent frames of the lost frame in *Bridge-Close* video

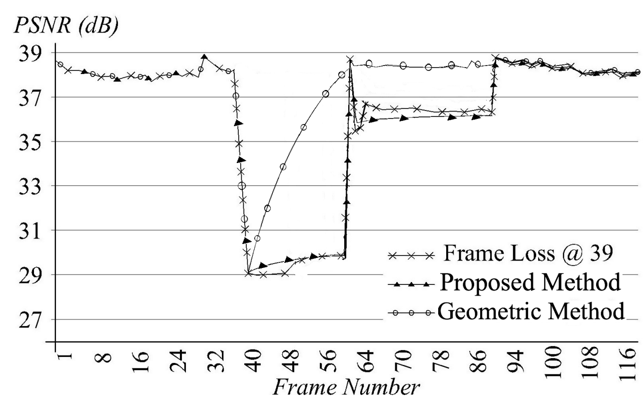


Figure 5.3: Accuracy of proposed vs. geometric methods to estimate the $PSNR$ for subsequent frames of the lost frame in *News* video

Fig. 5.4a plots the average $PSNR$ versus a single P-frame loss relative position to the preceding and subsequent I-frames while Fig. 5.4b demonstrates the same results for burst loss with length 2. The linear relationship between average $PSNR$ and n clearly appears, and we see that (5.7) is quite precise in case of burst loss. Note that A and B in (5.7) may not be constant for different I-frame intervals, since the amount of motion can vary throughout a video.

In Fig. 5.4, A and B are obtained from the numerical results. To calculate A and B at the encoder, the average $PSNR$ is calculated via (5.4) and (5.6) when a lost frame is reconstructed through loss concealment. This calculation should be repeated for another loss in the same interval of two consecutive I-frames.

We must note that our simulations reveal that (5.7) also applies in situations of burst loss. Thus, by employing our proposed model, it would be possible to estimate the distortion of loss patterns simply by calculating A and B, and knowing the relative position of lost frames, which is trivial enough.

5.5 Effect of transmission pattern on average $PSNR$ in noisy channels

Fig. 5.1 has shown a significant drop in media quality in case of loss, from 10% to 50%, depending on the type of video. These effects can be worsened in case of burst loss. We consider here to what degree quality can be improved by a simple effect of changing the transmission schedule of packets. Various authors (e.g., [34, 161, 162, 164, 165]) have demonstrated that the ratio of total quality degradation to burst length is not linear and shorter burst length is more effective to decrease the average $PSNR$ than longer burst length, at comparable loss rates. In a noisy environment, especially in wireless networks, errors and packet loss tend to occur in similar bursty shapes [165, 166]. Therefore, when we use such channels it would be beneficial to reduce the occurrence of short burst losses,

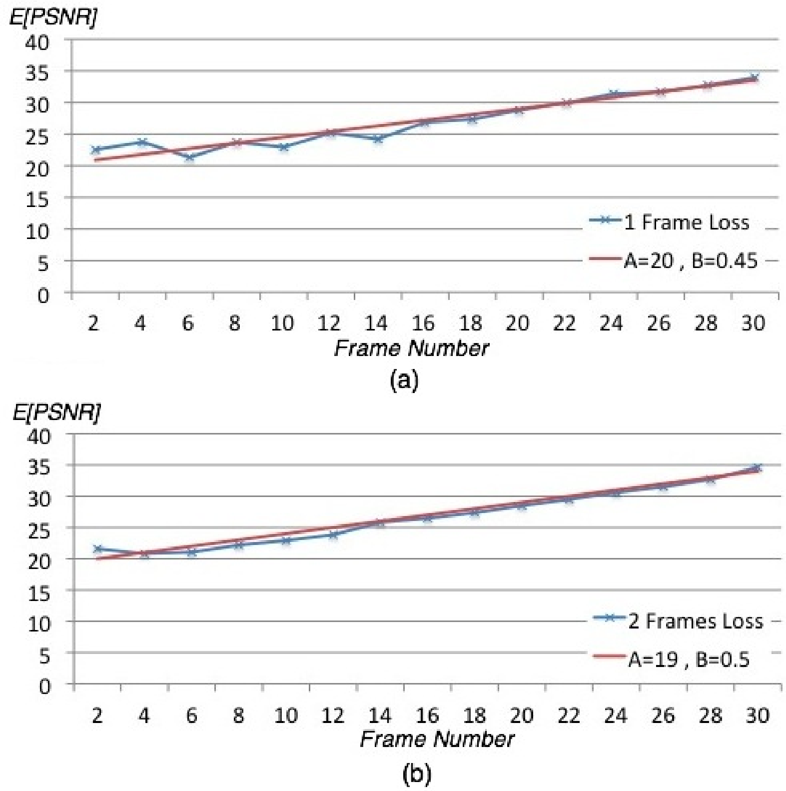


Figure 5.4: Average $PSNR$ for different loss relative positions to the preceding I-frame

to reduce total distortion. We explore here the impact of uniform rate packet transmission as opposed to a bursty transmission which occurs naturally because of video frame fragmentation and codec behaviour.

5.5.1 Packet loss model

Fig. 5.5 shows the underlying packet loss model used in our study. To simplify the model, we assume that all packets/frames are lost with a probability of 1 if they are sent during the noisy period, and received correctly otherwise. The noisy periods occur independently and are identically distributed with probability p. Since Packet Loss Ratio (PLR) greater than 10 percent is annoying in video transmission and the distortion is large enough to force

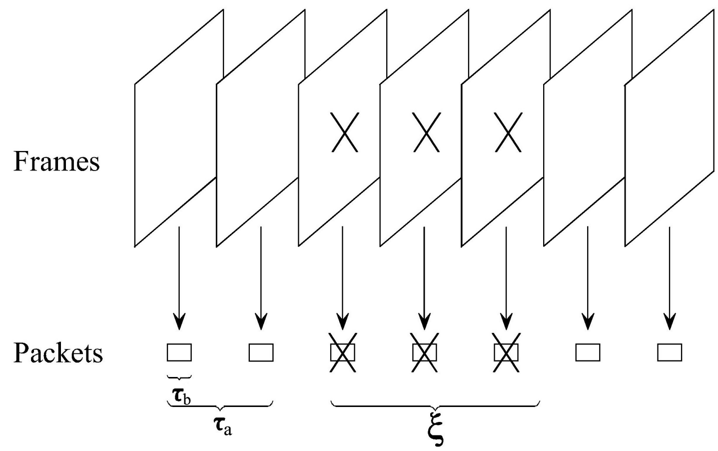


Figure 5.5: Packet loss model.

the user to re-optimize the network [164, 167], only noisy periods shorter than 100 ms are considered in our model.

Given the typically available residential Internet bandwidth and QCIF video bit rates, and also because of the uniform streaming method, the channel will be divided into periodic cycles of busy and idle time periods (see Fig. 5.5) during streaming time. In these patterns, the idle periods are longer than the busy ones.

The maximum burst loss length in this model is given by

$$n_{max} = \begin{cases} \lceil \frac{\xi}{\tau_a} \rceil + 1 & \tau_a \lceil \frac{\xi}{\tau_a} \rceil - \xi < \tau_b \\ \lceil \frac{\xi}{\tau_a} \rceil & \text{otherwise} \end{cases} \tag{5.8}$$

where ξ, τ_b and τ_a denote noisy period length, busy period length, and the total length of an idle-busy pair, respectively (Fig. 5.5), and $\lceil \, \rceil$ indicates the ceiling function.

The probability of burst losses with different burst length in this model is given by

$$P[n_{max}] = \begin{cases} \frac{\tau_b+\xi-\tau_a\lceil\frac{\xi}{\tau_a}\rceil}{\tau_a} \times p & \tau_a\lceil\frac{\xi}{\tau_a}\rceil - \xi < \tau_b \\ \\ (1 - \frac{n_{max}\times\tau_a-\tau_b-\xi}{\tau_a}) \times p & \text{otherwise} \end{cases} \tag{5.9}$$

$$P[n_{max}-1] = \begin{cases} (1 - \frac{\tau_b+\xi-\tau_a\lceil\frac{\xi}{\tau_a}\rceil}{\tau_a})\times p & \tau_a\lceil\frac{\xi}{\tau_a}\rceil - \xi < \tau_b \\ \\ \frac{n_{max}\times\tau_a-\tau_b-\xi}{\tau_a} \times p & \text{otherwise} \end{cases} \tag{5.10}$$

where p is the probability of noisy period occurrence.

Since the duration of noisy periods does not depend on the quantity of sent packets, the duration and probability of noisy periods are not affected by changes in streaming patterns. In Fig. 5.6, we observe a bursty transmission of n_{max} packets. The probabilities of burst losses with different burst length in this pattern are given by

$$P[n] = \begin{cases} \frac{2\tau_b+\xi-n_{max}\tau_b}{n_{max}\tau_a} \times p & n = n_{max} \\ \\ \frac{2\tau_b}{n_{max}\tau_a} \times p & 0 < n < n_{max}, \\ \\ \frac{n_{max}(\tau_a-\tau_b)-\xi}{n_{max}\tau_a} \times p & n = 0 \end{cases} \tag{5.11}$$

The probability of “no loss” increases if we send the packets with a bursty pattern through a channel with constant noisy period length (Fig. 5.6). Therefore, the total distortion in bursty sending pattern is improved compared to the common uniform sending pattern.

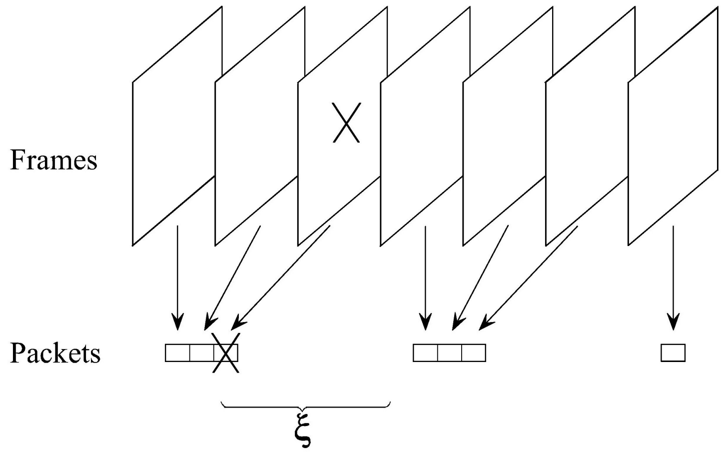


Figure 5.6: Bursty sending model in noisy environment

5.5.2 Simulation

To investigate the improvement of transmission performance using bursty scheduling, we have conducted a number of simulations and calculated the average $PSNR$ for different streaming type/noise situations. For instance, in the *News* video sequence streaming, an 80 ms noisy period occurs every second ($PLR \simeq 8\%$), where the packets/frames are sent both uniformly and with a bursty pattern. The channel bandwidth is assumed to be 512 Kbps, and MTU for IP protocol is set to 1024. The total average $PSNR$ is calculated by

$$E[PSNR] = \sum_{0}^{n} \left(E[PSNR_n] \times p_n \right). \tag{5.12}$$

Table 5.4 lists the simulation outcomes including probabilities of different burst lengths for uniform and bursty sending patterns, along with the relevant average $PSNR$. By substituting this table's content in (5.12), the total average $PSNR$ for uniform and bursty transmission patterns will be 35.89 and 36.23 dB, respectively. Given that the average $PSNR$ in no loss case is 39.12 dB (see Table 5.4), the average $PSNR$ in bursty transmission case is degraded by 2.89 dB, which demonstrates more than 10 % improvement

Table 5.4: Average $PSNR$s and probabilities of different burst length for uniform and bursty transmission patterns when there is one 80 ms noisy period every second and the packets/frames are sent in a uniform or in bursty pattern.

Burst length (n)	3	2	1	0
Probability in uniform transmission	0.5	0.5	0	0
Probability in bursty transmission	0.77	0.05	0.05	0.13
$E[PSNR_n]$ (dB)	35.72	36	36.6	39.12

compared to the average $PSNR$ degradation in uniform transmission case, i.e., 3.23 dB. According to (5.11) and depending on the probability of some burst lengths, the average $PSNR$ can improve up to 30 %. In case of burst transmission, although the probability of burst loss of length 3 increases, the total distortion decreases because of the decline in the probability of burst loss of length 2. This is due to the fact that the ratio of total quality degradation to burst length is not linear (e.g., [34, 161, 162, 164, 165]), i.e., the impact of the decreased probability of burst length of 2 is greater than that of increased probability of burst length of 3.

Finally, we observe that the advantages of our technique compared to that of other error resilience techniques are: 1) not increasing the bit rate, 2) being simple, and 3) not requiring any changes in receiver side.

5.6 Conclusion

In this chapter, a new method for estimating the $PSNR$ for the subsequent frames of the lost frame was presented. Based on this model, there is a linear relationship between average $PSNR$ and the relative position of lost frame to the last I-frame, regardless of the amount of motion from frame to frame. To achieve more accurate estimations of $PSNR$ (closer to the actual values), two new factors were introduced: an exponential and a conditional attenuation factor accounting for spatial filtering and Intra update of I-frames, respectively. According to the comprehensive simulations for three types of videos, we can conclude that the distortion propagation estimated by the proposed model is significantly more accurate than the one estimated by other recent methods. Furthermore, the proposed model may be used to estimate the average quality of a video, if the position (index) of lost frame is known.

We have also proposed and verified a packet transmission schedule for improving the performance over noisy channels. The proposed technique is simple and requires neither any changes at the receiver side nor any increase in the bit rate.

Chapter 6

Perceptual Video Quality Management

6.1 Introduction

To reduce storage space and to transmit video over bandwidth-limited networks, compression of video bitstream is essential. To compress video data, the H.264 codec [152]—the state of the art in video compression—employs, among other techniques, spatial transforms and motion compensated prediction between consecutive frames to exploit spatial and temporal redundancy, respectively. According to the dynamic nature of video content, the data rate of coded bit streams can be changed on the fly. Moreover, the "best effort" nature of the Internet makes it a competitive environment for different applications to increase their throughput; hence congestion and consequently loss and delay inevitably happen within the network. Despite the indisputable benefits of compression, the compressed video data is highly vulnerable to data loss. Indeed, dependency of each coded frame to previous frames' data means that error due to loss is propagated to subsequent frames. Thus, the distortion caused by data loss interferes with the objective of video quality. As we have moved to a unique (IP) network for multiple services, it has appeared that traditional network-level QoS parameters do not tell a sufficient story for media quality and the focus for quality assessment has moved to *quality of experience* (QoE) which has been defined by the ITU-T as the overall acceptability of an application or service, as perceived subjectively by the end-user.

Further, since the video's bit rate varies because of different video characteristics such as frame rate, resolution, compression level, content, etc., a similar network situation may cause end users to perceive a different level of quality for different videos.

Video conferencing is currently commonly employed over the Internet, and it is also expected that video chatting will be one of the key business areas for mobile service providers (e.g., 3G and 4G). To meet customer expectations, service providers should know the level of quality which is found acceptable by customers. Based on this information, service providers need to manage and control resources efficiently. However, managing and deploying more resources not only increases costs but also sometimes is not possible (e.g., in mobile environments, the bandwidth cannot be more than a certain level). Therefore, it seems that designing flexible (intelligent) applications, which can dynamically adapt themselves with existing networks by managing the video system (e.g., bit rate) without adverse effect on end-users' perceived quality, has become an overwhelmingly important issue. In other words, perceptual quality management by video conferencing applications is meant to lead to more efficient and economic deployment of available resources while keeping the end user's satisfaction at an acceptable level. Control mechanisms for perceptual quality include monitoring of the information regarding the network and end users' condition as well as adjusting the corresponding influential factors. For video streaming, the Scalable Video Coding extension of codec H.264 (H.264/SVC) provides a solution for spatial, temporal, and quality scalability with a smooth switching between different bit rates streaming [168].

Two main questions this chapter tries to answer are "what is the actual perceived video quality when video parameters are changed to meet the bandwidth limitation?" and "what are the best video parameters for specific video bit rates considering the perceived (subjective) quality by the end users?" This chapter investigates the effect of different coded-video factors such as frame rate and quantization parameter (QP) on video data bit rate and perceived video quality. Further, it looks into QoE control through adjusting these video parameters given the bandwidth limitations imposed by the network. To focus our study and make new contributions to the extant literature, we have selected the QCIF (176×144), CIF (352×288), and VGA (640×480) video resolutions and medium motion video content (e.g., talking head) which are heavily used in video conferencing applications in the Internet and mobile networks.

Our research prominent contributions are threefold; 1) extensive measurement studies for investigating the effect of different control parameters (i.e., frame rate and quantization) on bit rates limited by network bandwidth have been conducted; 2) we present the results of subjective tests conducted for measuring the end-users' perceived video quality, to find the optimum video parameters based on the given network bandwidth and acceptable perceived quality level; and 3) we propose a perceptual quality control algorithm based on the mentioned measurements. To conduct simulations for verifying the efficiency of the proposed perceptual quality management algorithm and to implement this algorithm in the practical applications, a congestion control technique derived from prior art is introduced in this chapter. It should however be noted that the perceptual quality control algorithm put forth in this chapter is independent of the proposed congestion control technique and can be employed in combination with different congestion control algorithms (see Appendix A).

The rest of the chapter is organized as follows: Section 6.2 introduces different congestion control methods in real-time multimedia transmission as well as recent studies regarding the effect of video parameters on the perceived quality. Section 6.3 presents the coding results for different video parameters. The details of subjective video quality measurement tests and their outcomes are presented in Section 6.4 and 6.5. In Section 6.6 our perceptual quality control algorithm is proposed. Simulations and numeric results demonstrate the effectiveness of the proposed algorithm relative to others in Section 6.7. Section 6.8

concludes the chapter.

6.2 Background

6.2.1 Effect of video parameters on Perceptual quality

In case of real-time data streaming over the Internet, one of the most effective methods to cope with congestion-induced degradation is reducing the bit rate. To reduce the bit rate of a video coded with H.264, different methods can be deployed as follows: 1) frame rate reduction, 2) resolution decrease, and 3) increased compression or quality decrease. To investigate the effect of these methods on the end-users' perceived quality some studies have been conducted recently. ITU-T Recommendation G.1070 has modelled the perceived video quality as a function of bit rate, frame rate, and packet loss [43]. Tao Liu et al. in [44] have also investigated the effect of bit rate, frame rate, and packet loss on perceived video quality and have extended the perceptual quality estimation method, introduced by ITU-T Rec. G.1070, to a real-time video quality monitoring. Thomas Zinner et al. in [38] have conducted a measurement study and quantified the effect of 1) video frame rate, 2) scaling method, 3) video resolution, and 4) video content types on the perceived quality by means of the Structural Similarity Index Metric (SSIM) and Video Quality Metric (VQM) full-reference metrics. Objective tests have been used in their study to determine the level of perceptual quality. Furthermore, they have focused on high resolution videos. In [39], Y. Pitrey et al. have evaluated the performance of two AVC and SVC standards for coding the video data in different situations by conducting the subjective video quality tests. McCarthy et al. in [40] have compared the importance of frame rate and quantization (i.e., video quality due to data compression) in the case of watching high motion videos such as a football game in CIF and QCIF sizes. Knoche and Sasse in [169] have discussed the preferred video size by viewers for a given video resolution. A perceptual quality prediction model for QCIF videos has been introduced in [41] and [42]. The authors have focused on the effect of loss rate, mean burst length, bit sending rate, and video content type on the perceived video quality. They have employed Peak Signal to Noise Ratio (PSNR) as the video quality metric in their research. However, since the PSNR's outcome as an objective video quality measurement may differ substantially from the real perceived quality measured by subjective tests, especially when the change in frame rate is employed to adjust the bit rate [170, 171], we have used the subjective measurement method in our research.

In this chapter, the effects of frame rate as well as compression level on bit rate and consequently on the end user's perceived quality are investigated. Our study has focused on medium-motion videos with QCIF, CIF, and VGA resolutions, which to the best of our knowledge have not been specifically subject to similar studies despite the fact that they are widely used over the Internet or mobile networks (e.g., by video-chat applications).

6.2.2 Rate control for congestion and loss avoidance

Loss avoidance is an overwhelmingly important issue in real-time video streaming over the Internet, over and above congestion control techniques. The sending sources use either open loop or closed loop techniques for streaming data through a bottleneck link. If the network and traffic characteristics are precisely known to the sending source in advance, the open loop control technique is quite useful and advantageous [172].

However, to share the network resources fairly among the various flows, the closed loop control mechanism is preferable. The loss rate which is monitored and measured by the receiver node is the most common feedback signal for the closed loop congestion control systems (e.g., congestion control in TCP). The most popular congestion control technique is the one which uses Additive Increase and Multiplicative Decrease (AIMD) behavior of TCP-Reno's congestion window control [173]. Since real time multimedia transmissions do not use a perfect connection-oriented protocol and also since AIMD causes rapid fluctuations in bit rate, multimedia applications employ different congestion control mechanisms to satisfy the perceived quality by smooth streaming, in addition to helping other TCP flows to fairly control congestion. The well-known congestion control mechanisms which use RTP/UDP are TCP Friendly Rate Control (TFRC, [174]), TCP Emulation At Receiver (TEAR, [175]), Datagram Congestion Control Protocol (DCCP, [176]), Rate Adaptation Protocol (RAP, [177]), TCP-Friendly Window-based Congestion Control (TFWC, [178]), Video Transport Protocol (VTP, [179]), and TFRC-Wireless ([180]). All of these methods emulate the AIMD behaviour of TCP-Reno's congestion window control. With the exception of VTP, all of them calculate an equivalent rate R to TCP-Reno by the following formula:

$$R = \frac{M}{RTT\sqrt{\frac{2p}{3}} + 3p(1+32p^2)t_{RTO}\sqrt{\frac{3p}{8}}} \tag{6.1}$$

where RTT is the round trip time, p is the packet loss rate, t_{RTO} is the retransmission time, and M is packet size [174]. Since these methods try to emulate TCP-Reno's behaviour perfectly, they inherit TCP-Reno's disadvantages (e.g., taking into account the occasional losses which are not caused by congestion).

For VTP, the sending rate ($R(i+1)$) of the next RTT period ($RTT(i+1)$) is calculated by this formula:

$$R(i+1) = \frac{R(i) \times RTT(i) + 1}{RTT(i) + \Delta RTT(i)} \tag{6.2}$$

where $R(i)$ and $RTT(i)$ are the sending rate and the RTT of the i^{th} RTT period, respectively, and $\Delta RTT(i)$ is the difference between $RTT(i)$ and $RTT(i-1)$. The problem with the VTP technique is that it assumes the buffer of the router facing the bottleneck link to be large.

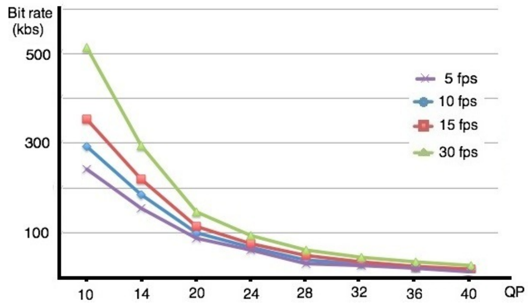
Although all these TCP-friendly techniques provide relatively smooth transmission rates, a smooth data rate does not always smooth video playback [181]. In this chapter, a bit rate control technique is proposed which utilizes a composition of TCP-friendly

(TCPF) and bandwidth estimation techniques. In this technique, the occasional losses are not taken into account by the TCP-friendly bit rate formula. Moreover, quality degradation is decreased by resending the most important frames (e.g., I-frames) at specific moments when the probability of loss occurrence is high.

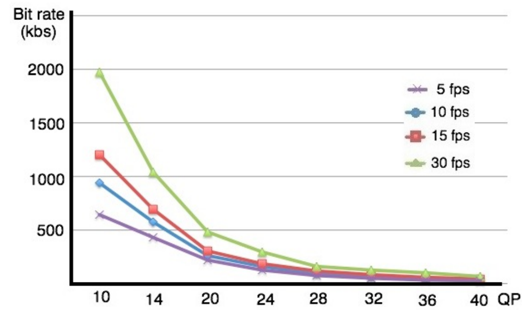
6.3 Effect of video parameters on video bit rate

We have investigated the effects of frame rate and quantization parameter (i.e., related to compression level) on the video bit rate. Since our research focus has been on video conferencing over the Internet, we consider medium motion video (e.g., talking head) and QCIF (176×144), CIF (352×288), and VGA (640×480) as the video content type and resolution format, respectively. Standard Akiyo and Foreman videos with 10-second length were selected as the medium motion sequences, coded with H.264 using the reference software JM 18. The first coded frame is Intra-coded (I), followed by Inter-coded frames (P), and Intra updates occur every second. The frame rates which we have investigated are 30, 15, 10, and 5 fps. We have measured the bit rate for the video sequences which have been coded at different compression levels by setting the value of the quantization parameter (QP) to 10, 14, 20, 24, 28, 32, 36, and 40. All these measurements have been conducted for different frame rates (i.e., 5, 10, 15, and 30 fps). Therefore, 32 video clips were produced and investigated for each resolution.

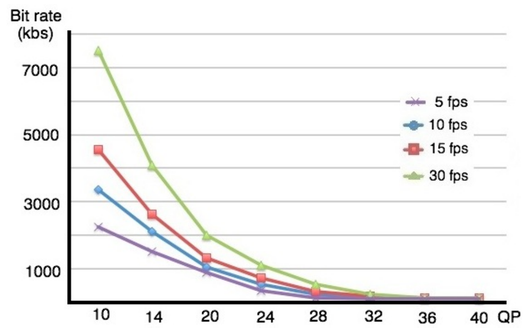
Fig. 6.1 shows the relation between video bit rate, frame rate, and compression level/QP for different video sizes. A careful examination of Fig. 6.1 reveals that, when the frame rate decreases, the bit rate does not decrease in case of higher QP value as much as that of lower QP value. It is shown in Fig. 6.2 that the bit rate vs. frame rate curve flattens when the video data is compressed more (i.e., larger QP).



(a) Bit rate versus QP for QCIF-size video (Akiyo).



(b) Bit rate versus QP for CIF-size video (Akiyo).



(c) Bit rate versus QP for VGA-size video (Foreman).

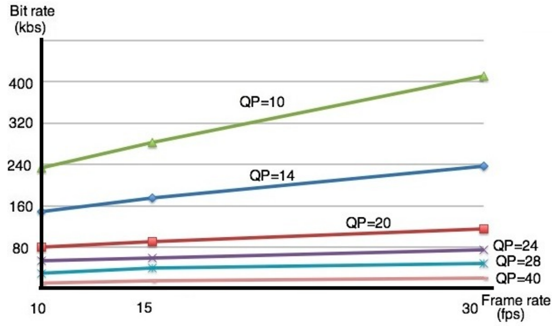
Figure 6.1: Bit rate versus QP for different frame rates.

6.4 Subjective assessment experimental setup

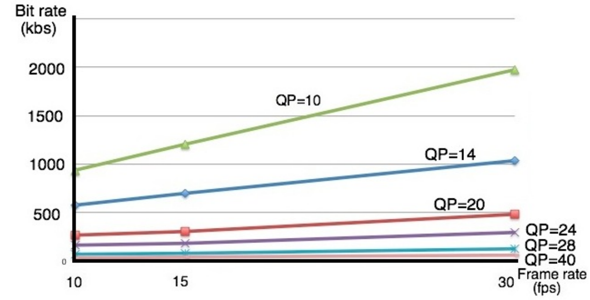
To examine the effects of frame rate and compression level of the video sequence on end-users' perceived quality, we have conducted 96 subjective tests (32 video clips for each of QCIF, CIF and VGA resolutions discussed in Section 6.3).

6.4.1 Subjective test methodology

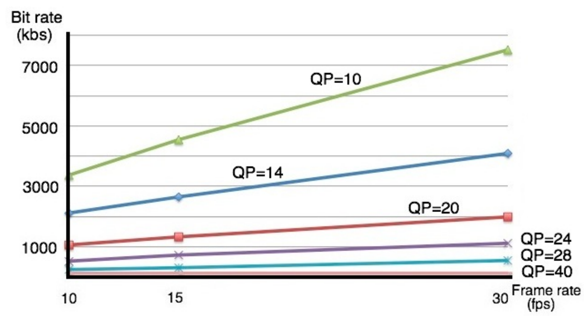
Our subjective tests have been performed following the guidelines established in ITU-T Recommendation P.910 [58]. Quality ratings are made using the ACR rating scale. The



(a) Bit rate versus frame rate for QCIF-size video (Akiyo).



(b) Bit rate versus frame rate for CIF-size video (Akiyo).



(c) Bit rate versus frame rate for VGA-size video (Foreman).

Figure 6.2: Bit rate versus frame rate for different QPs.

test videos are viewed randomly one at a time and rated independently on a 11-level scale from 0 to 10 (bad and excellent are set to 1 and 9, respectively). To obtain the MOS, all subjects' ratings are averaged (see Chapter 2).

We have used the single stimulus-hidden reference removal method [67] in which the reference video is also viewed by subjects who are not aware of watching the original video along the other test videos. The reference video rating scores are withdrawn from the results of the corresponding test. It helps us to insure the subject's rating accuracy.

6.4.2 Subjects

Twenty to twenty five graduate students participated in our subjective tests for each video size. The age range for participants was between 22 to 55 years. None of them were working in the field of video quality, although some of them were familiar with audio quality.

6.4.3 Test setup

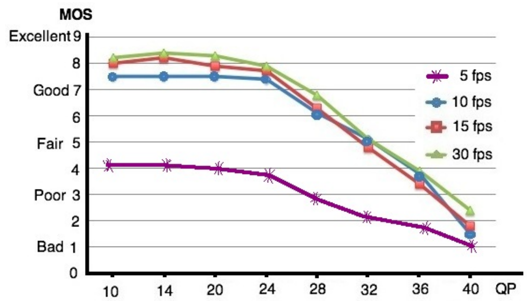
The tests were conducted in a quiet laboratory. A 15" MacBook Pro at its maximum resolution (1400×900) was used. The video clips were viewed in their original size in the middle of the screen surrounded by a dark background. The viewers' distance to the screen varied between 6 to 8 times the video's height. The video clips were displayed randomly individually for each assessor.

6.5 Experimental results

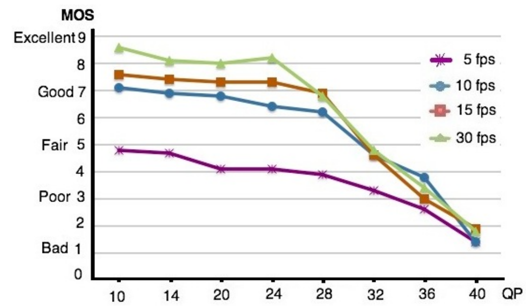
The mean of the rating scores was calculated for 33 video clip tests (i.e., 8 different compression levels for 4 different frame rate videos and the reference video) for each resolution. Fig. 6.3 presents the MOS of perceived video quality for different frame rates and compression levels. These MOSs with their associated 95% confidence intervals are also reported in Table 6.1.

6.5.1 Perceived quality and frame rate

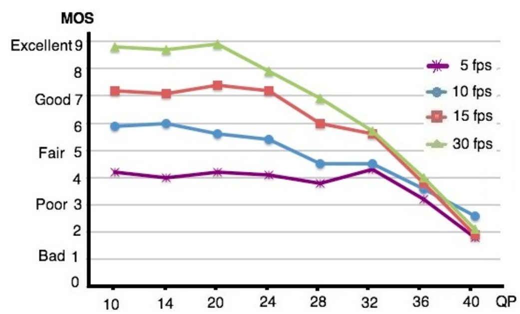
It is observed in Fig. 6.3 that no matter the resolution, very low frame rate videos (e.g., 5 fps) are not acceptable for the users of video conferencing applications. Therefore, a conservative critical value of 10 fps can be proposed for the video conferencing applications. The scores of other frame rates (i.e., 10 fps and more) are very close for QCIF-size video. Hence, we can state that in the medium motion videos typically viewed at a QCIF resolution, frame rates greater than approximately 10 fps are not noticeable for the end-users, whereas the effect of frame rate on the perceived video quality is more significant for videos with higher resolutions.



(a) MOS versus QP for QCIF-size video (Akiyo).



(b) MOS versus QP for CIF-size video (Akiyo).



(c) MOS versus QP for VGA-size video (Foreman).

Figure 6.3: MOS versus QP for different frame rates.

6.5.2 Perceived quality and compression level

Fig. 6.3 reveals that the MOS of any compression level (QP) less than 24 are not significantly different. To ease the observation of this statement, Fig. 6.4 shows the MOS versus frame rate greater than 10 fps for different QP values. Based on these results, our proposed quantization critical value for all frame rates (equal or greater than 10 fps for QCIF

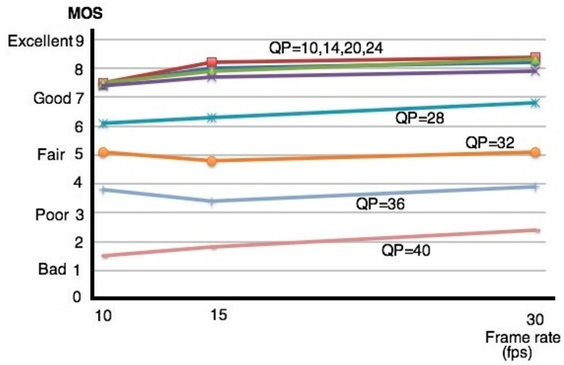
Table 6.1: *MOS* and 95% *Confidence Interval* (C.I) values for different video clips.

		Quantization Parameter (QP)									
		10	**14**	**20**	**24**	**28**	**32**	**36**	**40**		
Frame rate	**10**	7.56	7.48	7.5	7.52	6.8	5.2	3.8	1.52	**MOS**	QCIF
		0.46	0.39	0.44	0.48	0.52	0.58	0.48	0.44	**C.I.**	
	15	8.05	8.3	8	7.64	6.36	4.72	3.48	1.8	**MOS**	
		0.41	0.34	0.50	0.40	0.33	0.42	0.48	0.40	**C.I.**	
	30	8.2	8.4	8.32	7.87	6.8	5.08	3.8	2.44	**MOS**	
		0.67	0.33	0.38	0.36	0.39	0.39	0.25	0.42	**C.I.**	
Frame rate	**10**	7.13	6.88	6.81	6.44	6.19	4.63	3.75	1.44	**MOS**	CIF
		0.5	0.43	0.48	0.51	0.54	0.84	0.38	0.36	**C.I.**	
	15	7.56	7.44	7.25	7.38	6.94	4.63	3	1.94	**MOS**	
		0.59	0.47	0.55	0.59	0.49	0.62	0.44	0.38	**C.I.**	
	30	8.56	8.06	8	8.19	6.81	4.81	3.44	1.81	**MOS**	
		0.44	0.46	0.59	0.48	0.65	0.45	0.84	0.32	**C.I.**	
Frame rate	**10**	5.88	6	5.59	5.35	4.53	4.53	3.59	2.59	**MOS**	VGA
		0.53	0.41	0.67	0.67	0.53	0.48	0.41	0.61	**C.I.**	
	15	7.41	7.12	7.35	7.24	6.06	5.65	3.88	1.88	**MOS**	
		0.63	0.47	0.63	0.43	0.46	0.53	0.33	0.55	**C.I.**	
	30	8.82	8.71	8.88	7.88	6.88	5.71	4	2.12	**MOS**	
		0.48	0.58	0.47	0.67	0.55	0.60	0.67	0.58	**C.I.**	

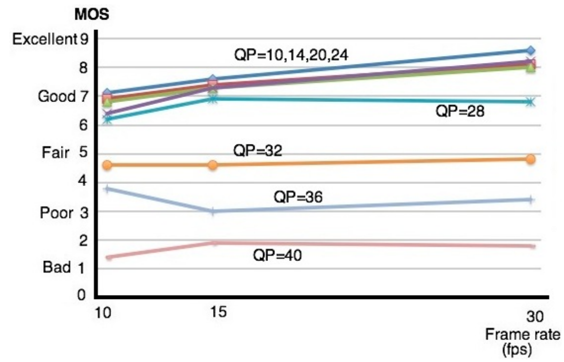
and CIF-size videos and equal or greater than 15 fps for VGA-size videos) is $QP = 30$, whereas $QP \approx 24$ can be the quantization critical value for VGA-size videos with frame rate of 10 fps.

6.5.3 Perceived quality and bit rate

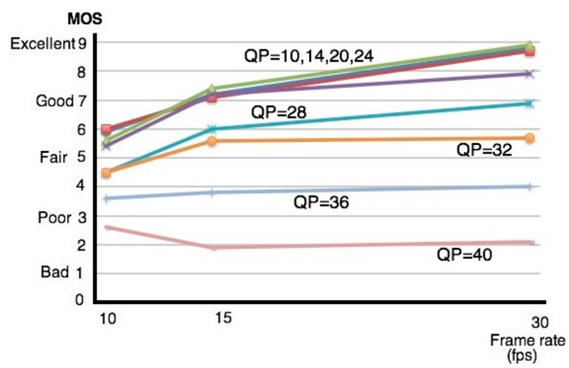
Fig. 6.5 demonstrates the relation between end user satisfaction and video streaming bit ratefor different video resolutions. Fig. 6.5-(a) shows that in case of transmitting the QCIF-size video through a network with a bandwidth greater than 100 Kbps, neither increasing the frame rate nor decreasing the quantization parameter (increasing the bit rate) signifi-



(a) MOS versus frame rate for QCIF-size video (Akiyo).



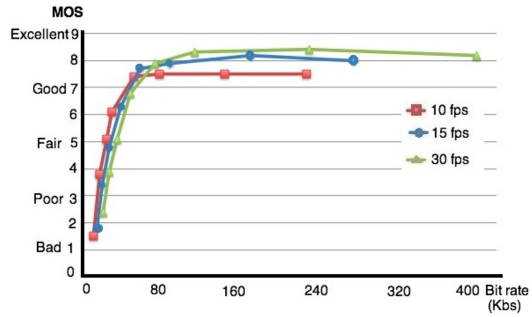
(b) MOS versus frame rate for CIF-size video (Akiyo).



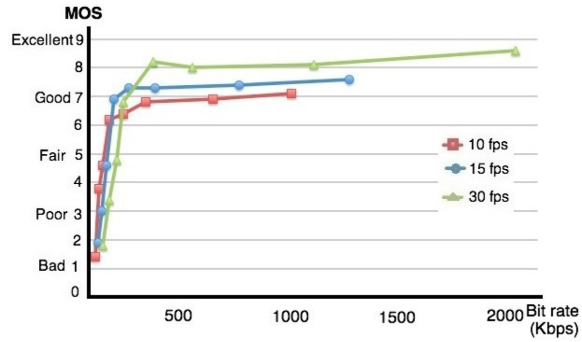
(c) MOS versus frame rate for VGA-size video (Foreman).

Figure 6.4: MOS versus frame rate for different QP values.

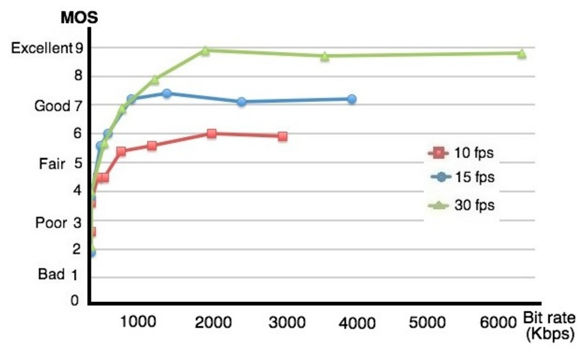
cantly affects acceptability by the end-user. A conservative estimate of the critical point of bit rate for QCIF-size video is 60 Kbps, although bit rates as low as 40 Kbps have been rated as *Fair* by assessors. It can be easily observed in Fig. 6.5-(b) and 6.5-(c) that the relationship between MOS and bit rate for the CIF- and VGA-size videos follows the same pattern as for the QCIF-size videos. However, as opposed to QCIF videos, changing the frame rate noticeably affects perceived quality when the QP is low (i.e., bit rate is high). The critical point of bit rate for CIF-size videos can be 100 Kbps, whereas, 250 Kbps might be the conservative estimate of the critical point of bit rate for VGA-size videos.



(a) MOS versus bit rate for QCIF-size video (Akiyo).



(b) MOS versus bit rate for CIF-size video (Akiyo).



(c) MOS versus bit rate for VGA-size video (Foreman).

Figure 6.5: MOS versus bit rate for different frame rates.

6.6 Transport and quality control

Since video conferencing applications are supposed to control the video parameters based on the network situation, accurate monitoring and measurement of network parameters is vital. Using the bandwidth estimate or TCPF's rate as an upper bound of the streaming bit rate, applications should choose the best video parameters to have the highest quality experienced by users under such conditions.

The control system exploiting measured network situation (e.g., available bandwidth) is sketched in Fig. 6.6. On both sides, the RTP and RTCP protocols are used for the transport of video data and feedback on transport quality [182], including information such as loss ratio and estimated bandwidth. Since it is highly desirable for each application to regulate its outgoing traffic flow so that every flow can have its own fair share of bandwidth, it is recommended to use an efficient and fair congestion control mechanism based on the feedback information. Based on the congestion and the feedback, the control unit will decide how to change the video parameters.

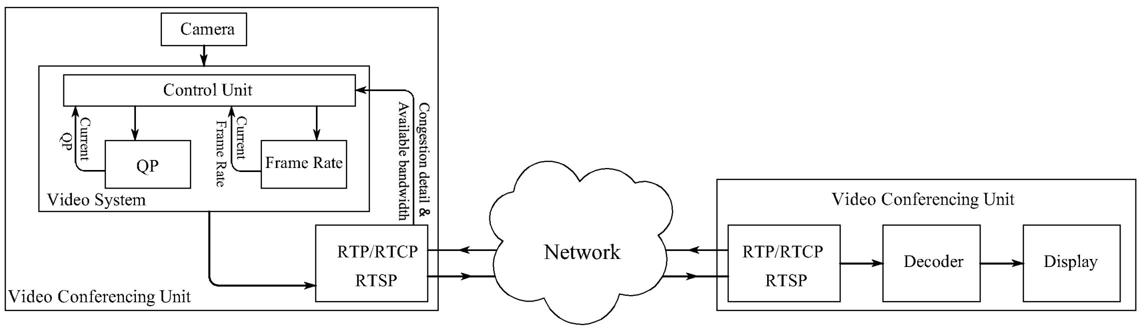


Figure 6.6: Overview of the perceptual quality control system using the available bandwidth estimation.

6.6.1 Congestion control algorithm

As mentioned in Section 6.2.2, extensive studies have been conducted on the subject of congestion control for multimedia applications. In order to harmonize H.264 streaming bit rate (which can vary smoothly by gentle change of frame rate and QP) with the time-varying throughput of heterogeneous networks, a new congestion control algorithm is proposed. It ensures that co-existing TCP flows are not treated unfairly by real time applications (e.g., video conference application), while simultaneously utilizing network bandwidth efficiently.

This algorithm consists of four phases: *normal* (i.e., no loss and no packets interval variation), *pre-congestion* (i.e., packets intervals increase), *congestion* (i.e., loss happens after changing packets intervals), and *occasional loss* (e.g., loss happens because of noise). The algorithm follows (6.1) to calculate the bit rate in *normal* phase. So the bit rate is set as:

$$R'(i+1) = \frac{\overline{M}(i)}{\overline{RTT(i)}\sqrt{\frac{2p}{3}} + 3p(1+32p^2)t_{RTO}\sqrt{\frac{3p}{8}}}$$
$$R(i+1) = min(R'(i+1), R_m) \tag{6.3}$$

where $R(i+1)$ is the suggested bit rate for the next streaming and monitoring interval. $\overline{M}(i)$ and $\overline{RTT}(i)$ denote the mean of packet size and RTT in last streaming and monitoring interval, respectively. R_m is the streaming bit rate for which the video quality is best and the MOS does not improve by increasing the bit rate through either changing the frame rate or the QP value. For instance, based on experimental results (Fig. 6.5), R_m is about 240 Kbps for the QCIF-size video (QP=14 and frame-rate=30 fps).

In the *pre-congestion* phase, the bit rates of TCP flows are not modified; therefore to follow the fairness policy in link capacity sharing, our proposed congestion control algorithm does not change its bit rate although the bandwidth is being estimated (explained in the following). The streaming bit rate is set equal to the estimated bandwidth (eBW) and the most important frames [49] (i.e., I-frames and very first P-frames after I-frames (see Chapter 5)) are resent using the excess of last sending bit rate over the estimated bandwidth (ΔR). Hence,

$$R(i+1) = R(i) = eBW + \Delta R \tag{6.4}$$

where eBW is the estimated bandwidth. Note that, to prevent the adverse effect of delay on perceptual quality, the interval time between re-sent frames and their original frames should not be more than 100 ms [183]. Since about 10 to 20 percent of all loss events in the Internet are preceded by a noticeable increase in delay [184], it is worthwhile to slightly decrease the video quality by reducing the streaming bit rate and resend some recent important frames. By doing so, the probability of a significant quality degradation occurrence due to frame loss remains low.

Our congestion control does not take the *occasional loss* into account. The bit rate is calculated based on (6.3), as if an occasional loss never happened.

Once the *congestion* phase is determined by observing the loss after the *pre-congestion* period, the bit rate is set to

$$R(i+1) = min(R_{TCPF}, eBW) \tag{6.5}$$

where R_{TCPF} is the bit rate calculated with (6.3). Since the streaming bit rate is equal to eBW in the previous phase, the bit rate modification will be very smooth.

6.6.2 Bandwidth estimation

Several methods have been developed for estimating link or bottleneck bandwidth in end-to-end communications. Although measuring the packet loss ratio in the receiver side can bear witness to the existence of a bottleneck or congestion in the middle of path, it cannot give the precise estimate of the available bandwidth. The authors of [185] and [186] use the packet pair technique to measure bottleneck bandwidth. In [104] and [150], analysing a packet's Round Trip Time (RTT) is used to measure the link bandwidth for each hop. All these methods assume that the links between nodes are symmetric. Jiang in [187] introduces an algorithm that can measure each hop link's bandwidth in both directions. Although his method is interesting, it is not suitable for realtime applications. Moreover, most of these methods add an overhead burden to the available bandwidth. Given that the only required data is the minimum hop bandwidth in each direction, we introduce a simple straightforward technique, inspired by the aforementioned methods, to measure the lowest intermediate hop's bandwidth in each path while sending video data.

Fig. 6.7 shows an arbitrary link connecting two end-nodes which are not necessarily time-synchronized. To measure the bottleneck hop bandwidth, node a sends frames/packets

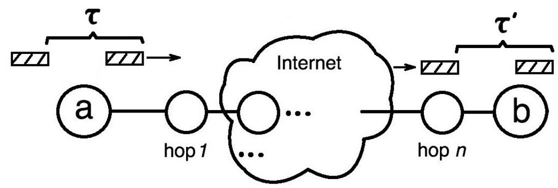


Figure 6.7: Testbed topology of bandwidth estimation method.

to node b with a specific time interval which is set based on the frame rate of the video data (e.g., every $\tau = 100$ ms in case of 10 fps-video). Node b may receive these frames/packets with a different time interval depending on the intermediate nodes' characteristics (e.g., queuing times); i.e., if one or more intermediate hops' bandwidth is less than the sending bit rate, the receiving time interval will be larger than the sending one ($\tau' > \tau$). Otherwise, if all intermediate hops' bandwidth is greater than the sending bit rate, the receiving and sending time intervals of consecutive packets/frames will be equal ($\tau = \tau'$).

The minimum intermediate hop's bandwidth can be calculated from:

$$eBW_{\text{min}} = \frac{1}{M}\sum_{i=1}^{M}\frac{P_i}{\tau_i'} \qquad \text{if } (\tau' \neq \tau) \tag{6.6}$$

where eBW is the estimation of the minimum bandwidth of intermediate hops, P_i is the first packet/frame size of i^{th} pair, and $M = 30$ is chosen based on [188]. If eBW_{min} is less than the current transmission bit rate, it should be chosen as the new bit rate. Otherwise, the bit rate is equal to or less than the available bandwidth.

6.6.3 Frame rate and QP selection

Fig. 6.5 shows that video conferencing applications cannot work properly and meet the users' expectations when the bandwidth is less than 40, 80, or 200 Kbps for QCIF, CIF, or VGA videos, respectively. For a bandwidth greater than the critical point, changing the frame rate and quantization may affect the end user's satisfaction differently depending on their present values. As depicted in Fig. 6.6, our proposed perceptual quality control system uses the current frame rate and the QP value as input data. Furthermore, based on this data and the congestion control algorithm, the perceptual quality control function will decide on the frame rate and QP by which the video data should be sent to the other party. Experimental results reveal that increasing the bit rate through either changing the frame rate or the QP value may not always cause a sensible perceptual quality improvement. Fig. 6.3 and 6.5 demonstrate that for all video resolutions, the frame rate plays the main role in establishing a level of QoE for the compressed videos with QP less than 24, whereas over 24, QP becomes more influential. Consequently, the pseudo code for the proposed QoE control scheme can be presented in the form of Algorithm 1.

As mentioned in the pseudo code, if the QP value is greater than 24, decreasing it will result in better video perceptual quality. But for QP values less than 24, increasing the frame rate is more effective than decreasing the QP in improving the MOS, specially for the videos with higher resolutions. However, in the case of performance-critical situations, e.g., limited bandwidth, it is crucial to change the video parameters so as to keep the perceptual quality above an minimally acceptable level. In this case, if the QP value is less than 24, increasing it will cause bit rate reduction without adverse effect on perceptual quality; otherwise, decreasing the frame rate will be more effective.

6.7 Validation

To investigate the efficiency and applicability of the proposed (i) bandwidth estimation, (ii) congestion control algorithm, and (iii) QoE control algorithm, the network scenario shown in Fig. 6.8 is simulated using the NS-2 software [125].

To show the accuracy of the bandwidth estimation based on (6.6), node 1 streams the QCIF-size video data to node 2. The frame rate and streaming bit rate are set to 30 fps (30

```
if (Available_BW <Critical_Value) then
    Re-Optimize the Network;
end
if (R(i+1) <R(i)) then
    if (Current_QP <24) then
        Increase the QP;
    else
        if (Current_Frame-Rate >Critical_frame-Rate_Point) then
            Decrease the Frame-Rate;
        else
            Increase the QP;
        end
    end
else
    if (Current_QP <24) then
        if (Current_Frame-Rate <30) then
            Increase the Frame-Rate;
        else
            Decrease the QP;
        end
    else
        Decrease the QP;
    end
end
```

Algorithm 1: Pseudo code of the proposed QoE control.

packets per second) and 240 Kbps, respectively. Node 3 generates and sends a constant bit rate flow along with some random flows to node 4. Figure 6.9 shows the total received bit rate at the bottleneck hop and the available bandwidth estimated by node 4. The bottleneck bandwidth is set to 1 Mbps and its hop's buffer is large enough to avoid packet loss. The average error in bandwidth estimation is less than 7% in this test which is rather reasonable for a simple and straightforward estimation method compared to the more complicated ones [104, 150, 185, 186].

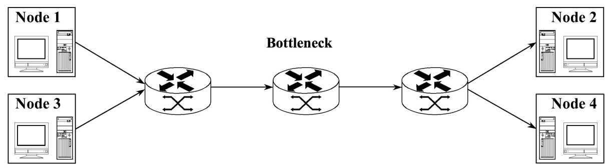


Figure 6.8: Testbed topology for simulation.

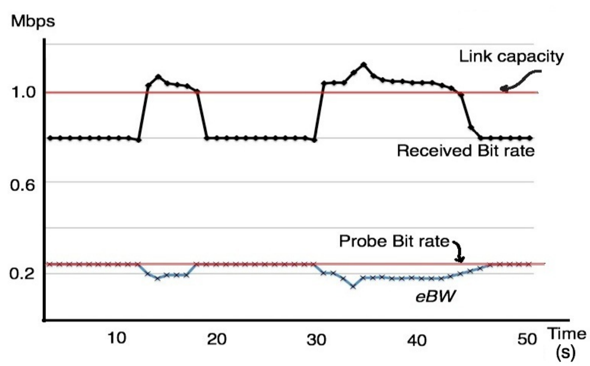


Figure 6.9: Available bandwidth estimation.

Figure 6.11 shows throughput comparison of TCPF and our congestion control algorithm when the occasional packet/frame loss happens in the path between server and client nodes. A random packet loss generator which obeys the uniform distribution is used in the simulations to drop packets at the loss rate of 2%. Figure 6.11 demonstrates the adverse effect of random loss which is not caused by congestion on streaming bit rate smoothness in the TCPF algorithm.

Fig. 6.11 shows the efficiency of our proposed congestion control algorithm compared to the TCPF algorithm; further, we have conducted a simulation using the results of Fig. 6.11 as the final streaming bit rate for our proposed control unit to demonstrate the interaction between our proposed QoE control algorithm and different congestion control techniques. Figure 6.10 shows the superiority of our proposed congestion control algorithm from a perceptual video quality perspective. Indeed, the preference of our proposed congestion control algorithm will be more evident the more frequently congestion occurs.

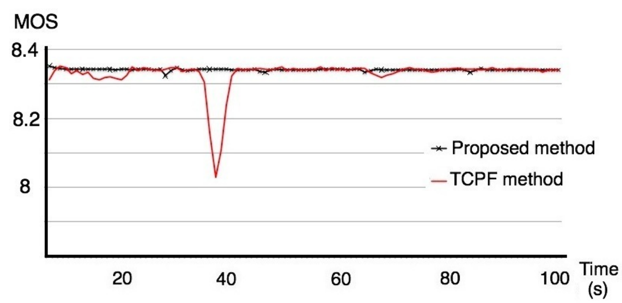


Figure 6.10: Comparison between proposed and TCPF congestion control algorithms from QoE's point of view.

To assess the protective effect of the *pre-congestion* phase in our proposed congestion control algorithm, nodes 1 and 2 generate a traffic flow based on either our proposal or the TCPF algorithm to transmit 10-second QCIF-size medium-motion video (i.e., Akiyo sequence). Nodes 3 and 4 are designed to generate the common Internet traffic flow for background traffic and make the aggregated traffic situation similar to that of a real Internet network. The Tmix module in NS-2 is utilized in nodes 3 and 4 in order to generate realistic Internet network traffic [126]. The random loss rate is set to zero and congestions are the only cause for the losses. The *pre-congestion* phase is detected 1.38 s (on the average) before loss happens (i.e., *congestion* phase) in this simulation.

Table 6.2: Simulation detail results for assessing the protective effect of *pre-congestion* phase in our proposed congestion control algorithm

Congestion control method	Overall loss percentage	Resent packets percentage	Average PSNR
TCPF	0.4 %	-	38.06
Our proposal	0.4 %	0.8%	38.35

Table 6.2 shows the detailed results for this simulation. It states that the proposed scheme outperforms TCPF in improving video quality. The average PSNR is degraded by 1.06 dB in the TCPF congestion control algorithm, whereas applying our proposal results in 0.77 dB degradation in the PSNR. Given that the average PSNR in a no-loss case is 39.12 dB, it can be concluded that our proposal improves PSNR degradation more than 27% compared to the TCPF congestion control.

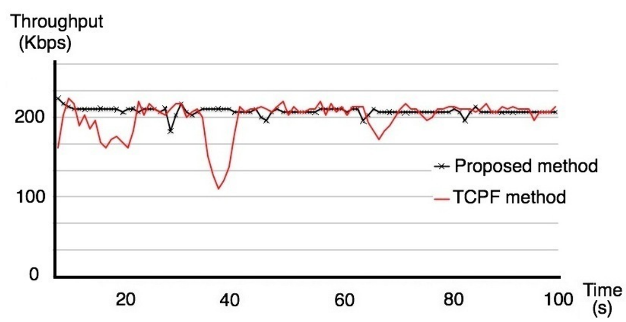


Figure 6.11: Throughput comparison of our proposed congestion control algorithm and TCPF when the probability of occasional loss is 0.02.

Although a congestion and loss control method is also proposed in this chapter, our main contribution is the control of QCIF medium motion video's perceptual quality by the application, aware of the network conditions (e.g., the most effective bit rate). Therefore, to compare the proposed video perceptual quality control algorithm with existing algorithms, a number of simulations were conducted with the assumption that all algorithms adjust their streaming bit rates based on our congestion control method. In these simulations, node 1 (i.e., the server of video streaming) generates a traffic based on the proposed congestion control method and sends data to node 2 (i.e., the client of video streaming). Nodes 3 and 4 generate a random traffic to cause congestion moments at the bottleneck link. The results of these simulations (i.e., streaming bit rate) are used to investigate the performance of proposed video perceptual quality control algorithm and other existing ones.

The MOSs of received QCIF-size videos for cases employing different video quality control algorithms in the server application are shown in Fig. 6.12 and 6.13. In Fig. 6.12, the proposed algorithm is compared to other algorithms in which the streaming bit rate is controlled by adjusting the QP while keeping the frame-rate constant. Since the real-time subjective quality evaluation tests are usually suitable for the long videos [59] and also because their reliability is dubitable [189], the results of our test are achieved based on the linear interpolation of experimental data in Fig. 6.5. Although temporal variation of frame rate and quantization may influence the perceptual quality [190–192], this effect seems to be negligible in this study as the test situations have been almost similar for the two compared algorithms. Moreover, in our proposed algorithm, the bit rate and consequently the video parameters change very gently without big step-sizes.

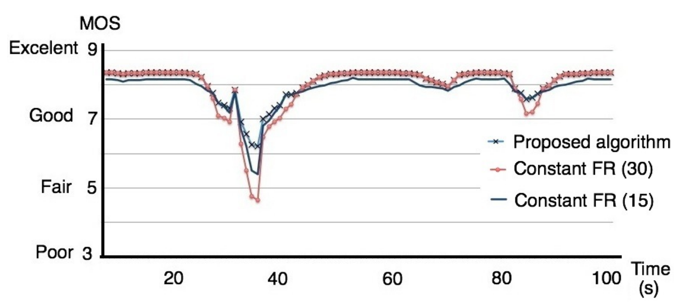


Figure 6.12: Comparison between proposed and constant frame-rate perceptual quality control algorithms.

Figure 6.13 demonstrates the superiority of our algorithm in perceptual quality control over other algorithms in which the frame-rate is changed to control the streaming bit rate.

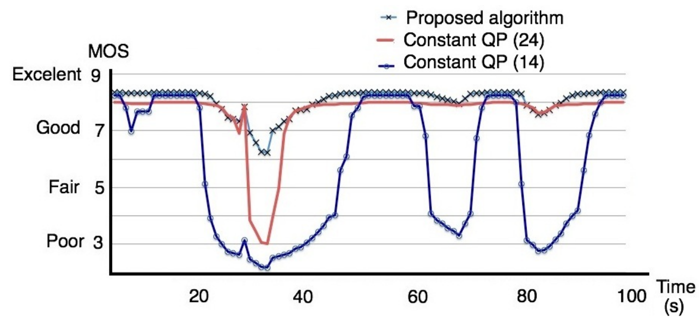


Figure 6.13: Comparison between proposed and constant-QP perceptual quality control algorithms.

Moreover, this figure shows that very low frame-rate videos annoy end users.

6.8 Conclusion

In this chapter, a framework for managing the end user's perceived quality for video coded with H.264 over limited bandwidth networks has been introduced. Unlike other similar studies, we have specifically focused on the medium-motion videos with QCIF, CIF, and VGA resolutions, the most pervasive video formats used by video conferencing applications across the Internet and mobile communication systems. The video streaming bit rate has been adjusted through changing the frame rates and compression levels to manage the perceptual quality.

To quantify the video quality perceived by end users, a measurement study has been conducted through subjective tests. The results demonstrate the relation between the main influential video parameters and the video quality experienced by end users. Simply stated, QP and frame rate play different roles in the perceived quality for medium-motion videos with different resolutions depending on their current values and the video bit rate; the effect of QP on end users' perceived quality is more significant in low bit rate videos with QP more than 24, whereas frame rate affects QoE more noticeably when the bit rate is high and QP is less than 24. The latter is more apparent for the larger-size videos.

Furthermore, after investigating the effect of different frame rates and compression levels on video streaming bit rate and consequently on video quality, we have proposed a perceptual video quality control mechanism for cases of limited-bandwidth.

Chapter 7

General Conclusions and Perspectives

7.1 Introduction

In telecommunications, performance has been assessed in terms of quality of service (QoS). Today, increased access to broadband networks has led to a fast-growing demand for Voice and Video over IP (VVoIP) applications such as Internet telephony (VoIP), video conferencing, and IP television (IPTV). While the evaluation of Video (or Speech) communication systems has been an important field for both academia and industry for decades, the introduction of VVoIP systems has created a new set of issues that require new evaluation methods. Moreover, since we have moved to a unique network for multiple services, it has appeared that traditional QoS measures do not tell a sufficient story and the focus has moved to Quality of Experience (QoE).

We can conclude that QoS and QoE are two interdependent concepts in the modern multimedia transmissions over IP networks and hence, they should be studied and managed with a common understanding, from planning to implementation and engineering (optimization). Moreover, although the QoS research field has been extensively studied, measuring network impairments for enhancing the QoE is nevertheless an open research area which is under consideration in this thesis.

Video streaming is currently commonly employed over the Internet, and it is also expected that video chatting will be one of the key business areas for mobile services through wireless communications (e.g., 3G and 4G). To meet customer expectations, service providers should know the level of quality which is deemed acceptable by customers. Based on this information, service providers need to manage and control resources efficiently. However, managing and deploying more resources not only increases costs but also sometimes is not possible (e.g., in mobile environment, the bandwidth cannot be more than a certain level). Therefore, it seems that designing intelligent applications, which can dynamically adapt themselves with existing networks by managing the video system (e.g., bit rate) without adverse effect on end-users' perceived quality, has become an overwhelmingly important issue. In other words, QoE management by video applications is meant to lead to more efficient and economic deployment of network resources while keeping the end user's satisfaction at an acceptable level.

The main objectives of this research are (i) to investigate the QoS parameters which affect the end user's perceptual multimedia quality and their measurement methods, (ii) to investigate the effects of some network and codec parameters on end user's perceived video quality, and (iii) to develop a QoE control algorithm for video streaming applications to cope the network bandwidth limitation.

This chapter concludes the thesis and highlights the main contributions. Considering the limitations of current work, the future research directions are also suggested.

7.2 Conclusions and Contributions

To conclude the thesis, the main contributions of this research are:

(1) Proposing a new packet-loss probability estimation method.

We have reviewed the existing online packet-loss probability (*plp*) estimation methods and the theory behind the finite buffer overflow probability (tail probability in infinite buffer) estimation. A new approximation for *plp* has been introduced based on the central limit theory by modelling the input traffic of an intermediate high speed node as a Gaussian process. Combining this online approximation with the offline output traffic measurement, we have proposed an accurate *plp* estimator which significantly improves the quality of the estimate compared to the recent proposed *plp* estimators which use similar theoretical basis.

(2) Improving the One-Way Delay estimation.

In this thesis, a recently proposed method for estimating the OWD has been analyzed and improved upon. This method is based on conducting multiple RTT measurements among pairs of nodes and applying the LSE method to find the more reasonable estimate of the OWD between two specific nodes. This is done by measuring all possible independent RTTs between them and one auxiliary third-party node. To secure more accurate estimates, some additional constraints have been introduced to the regular cyclic-path's set of equations. All the new constraints can be easily estimated based on the known behavior of nodes and their connection paths. To measure the transmission delay, which is often an asymmetric constraint, an accurate method has been introduced.

(3) Investigating the effects of frame loss and the position of lost frame on perceived video quality.

Using the average PSNR as a measure of perceptual quality of video, we have investigated the effect of frame loss position relative to I-frames on total distortion for the videos and proposed a model to estimate the PSNR of the received frames impaired by distortion propagation. A linear relationship between the average PSNR and the relative position of the lost frame w.r.t. the last I-frame has been derived from this model. To achieve more accurate estimations of PSNR (closer to the actual values), two new factors have been introduced: an exponential and a conditional attenuation factor accounting for spatial filtering and Intra update of I-frames, respectively. The proposed model may be used to estimate the average quality of a video, if the position (index) of the lost frame is known. We have also proposed a simple packet transmission schedule for improving the performance of noisy channels, and verified its efficiency through calculating the probability of different lengths of burst loss.

(4) Investigating the effects of H.264 video codec parameters on streaming bit rate and end user's perceived video quality and proposing a QoE control algorithm for the video streaming applications.

In this research, we have conducted extensive measurement studies for investigating the effects of different control parameters (i.e., frame rate and quantization) on the bit rates limited by network bandwidth. Unlike other similar studies, we have specifically focused on the medium-motion videos with QCIF, CIF, and VGA resolutions, the most pervasive video formats used by video conferencing applications across the Internet and mobile communication systems. We have presented the results of subjective tests conducted for measuring the end-users' perceived video quality and reported the optimum video parameters based on the given network bandwidth and acceptable perceived quality level. Finally, we have proposed a perceptual quality control algorithm based on the mentioned measurements.

7.3 Current Research Limitations

The study carried out in this thesis has a number of limitations which should be addressed in future research.

(1) Simulation-based performance evaluations.

The proposed algorithms and methods in this research have been assessed in simulated testbeds. This approach benefits from being economical, fast, repeatable, and easy to customize and control. However, real network conditions such as spikes in traffic or power/duration of unexpected noise in the Internet are unpredictable all the time.

(2) Limited QoS parameters.

This study takes the effect of packet loss on perceptual video quality into consideration. However, multimedia applications may also suffer from other IP networks impairments such as delay, jitter, etc. which have not been the focus of this research.

(3) Limited Internet services.

In this thesis, the perceptual quality of video has been studied. However, the QoS parameters may impact differently on the perceived quality of different services [193–195] such as web browsing, audio, and combination of audio and video transmission.

(4) Limited video sizes and contents.

The QCIF video resolution has been considered in investigating the effects of lost packet position on quality degradation and the QoE control algorithm has been proposed for the medium-motion videos. However, the end users may have different perceptions for other video types and resolutions when the network's and codec's parameters change.

(5) Accuracy of the subjective tests.

Although the QCIF resolution was selected due to the exponential growth of video applications over the cellular networks, all the subjective tests have been conducted on a computer which has a much bigger screen to assessors compared to a small handheld device.

7.4 Suggestions for Future Work

Considering the current work limitations and the extent of the QoE application, four main research directions are suggested for the future work.

(1) Other network impairments.

The bandwidth limitation has been considered for our proposed QoE control algorithm, whereas the affects of packet loss, burst loss, and delay have not been explored in this research. Adjusting the streaming bit rate by video conferencing application through changing the codec parameters was our proposed solution for limited bandwidth problem. Investigation of how the end nodes' applications can deal with other types of network impairment while keeping the end users' perceptual quality in the acceptable level would be an interesting area to explore.

(2) Performance validation using real system implementations.

The proposed method for improving the accuracy of the OWD measurement has been validated by simulations. However, cooperation of different research centres in implementing the proposed method using synchronizing hardware such as GPS, would be beneficial. Investigation of estimation of OWD by adding more auxiliary nodes and consequently more equations to the proposed model is also recommended. Investigation of how to implement the proposed OWD estimation method in a real QoE management system in a multimedia service such as video conferencing seems very interesting and worthwhile to be considered as a practical research for academia and industry.

(3) Focus on other video types.

Given that the scope of this study has been restricted w.r.t. the codec, type, and size of videos used for investigation and control algorithm development, exploring other codecs and types of videos (e.g., high-motion or 3D videos) with different resolution levels remains an important avenue for future research. Video conferencing has mainly been considered in this research. However, end users' perception for the video quality may vary for different IP multimedia services. Moreover, since different video contents such as sports, cartoons, movies, etc. have different audience, diverse opinions for the perceived quality are expected. Therefore, study of the QoE from these different points of view helps the service providers, which offer a variety of services, to utilize their resources more efficiently.

(4) Perceived quality of non-silent videos.

Since our research has focused solely on video quality, investigation of the QoS and codec parameters' effects on perceived quality for the non-silent videos is suggested for the future work. End users attach importance to the voice or image differently based on the service type. For instance, video quality might be more important than audio's in sport sequences or on the contrary, given a choice, end users may prefer very good voice quality over very good video quality in a video chat session. Moreover, the video and audio synchronization is another factor that should be considered in assessing the end users' quality perception.

Appendix A

Adaptive Video Streaming Application

A.1 Introduction

Today, increased access to the Internet networks as well as broadband networks have made possible and affordable the deployment of multimedia applications such as Internet telephony, video conferencing, and IP television (IPTV) by academia, industry, and residential communities. Therefore, streaming the multimedia occupies a large portion of the Internet capacity. The "best effort" nature of the Internet makes it a competitive environment for different applications to increase their throughput; hence congestion and consequently loss and delay inevitably happen within the network which adversely affect the perceived multimedia quality. Therefore, it seems that designing flexible (intelligent!) applications, which can dynamically adapt themselves with existing networks by managing the video system (e.g., bit rate) without adverse effect on end-users' perceived quality, has become an overwhelmingly important issue. To do so, we have created a streaming application based on the GStreamer framework [196] and the *mcn streaming* platform [197]. The application provides video streaming platform in which different adaptive H.264 encoding/decoding and streaming algorithms over the Internet network can be implemented. This application is also a complete video streaming software which includes the H.264 encoder/sender and decoder/receiver. A simple adaptive frame rate and QP algorithm has been implemented in this application.

The rest of this Appendix is organized as follows: Section A.2 gives a brief explanation of the GStreamer framework. Section A.3 discusses the application architecture. Sender and receiver pipelines are explained in Section A.4 and A.5, respectively.

A.2 GStreamer

"GStreamer is a framework for creating streaming media applications. The fundamental design comes from the video pipeline at Oregon Graduate Institute, as well as some ideas from DirectShow.

GStreamer's development framework makes it possible to write any type of streaming multimedia application. The GStreamer framework is designed to make it easy to write applications that handle audio or video or both. It isn't restricted to audio and video, and can process any kind of data flow. The pipeline design is made to have little overhead above what the applied filters induce. This makes GStreamer a good framework for designing even high-end audio applications which put high demands on latency.

One of the most obvious uses of GStreamer is using it to build a media player. GStreamer already includes components for building a media player that can support a very wide variety of formats, including MP3, Ogg/Vorbis, MPEG-1/2, AVI, Quicktime, mod, and more. GStreamer, however, is much more than just another media player. Its main advantages are that the pluggable components can be mixed and matched into arbitrary pipelines so that it's possible to write a full-fledged video or audio editing application.

The framework is based on plug-ins that will provide the various codec and other functionality. The plug-ins can be linked and arranged in a pipeline. This pipeline defines the flow of the data. Pipelines can also be edited with a GUI editor and saved as XML so that pipeline libraries can be made with a minimum of effort.

The GStreamer core function is to provide a framework for plug-ins, data flow and media type handling/negotiation. It also provides an API to write applications using its various plug-ins.

Specifically, GStreamer provides

- An API for multimedia applications.
- A plug-in architecture.
- A pipeline architecture.
- A mechanism for media type handling/negotiation.
- Over 150 plug-ins.
- A set of tools" [196].

A.3 Application Architecture

The overview of our adaptive video streaming application architecture is shown in Fig. A.1. The video data is sent via RTP packets from server side to client. The client sends information about network condition (i.e., QoS information) as the feedback to the server via RTCP packets. Using the QoS information and utilizing the QoE control algorithm explained in Chapter 6, server's encoder unit adjusts the codec parameters on the fly to meet the network limitations, keeping the perceptual video quality in the acceptable level.

A.4 Server

The video source is a "v4l2src" device which is the captured videos by the webcam. The captured video is encoded by the x264 encoder. A GStreamer RTP bin manages the sending

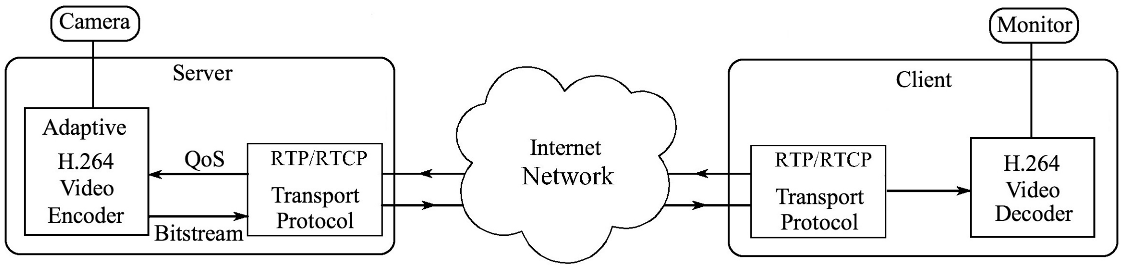


Figure A.1: Video streaming application architecture.

and receiving of RTP and RTCP packets. The pipeline of sender application is shown in Fig. A.2. The codec parameters of "x264enc" element are adjusted according to the feedback information received by RTCP packets from port 5005.

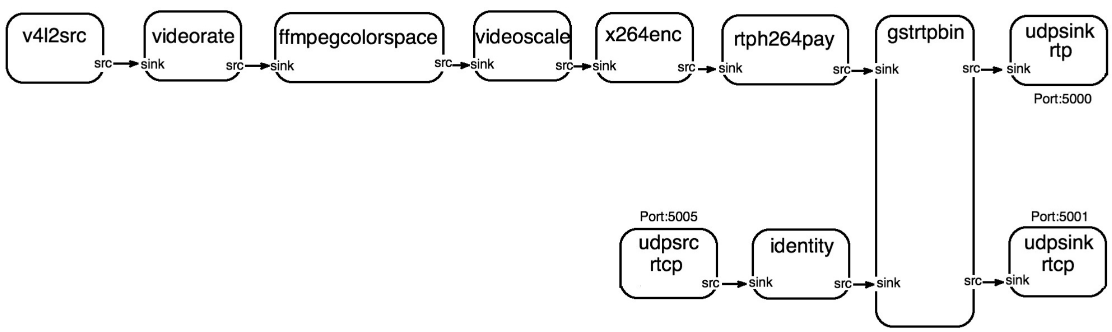


Figure A.2: Server pipeline.

A.5 Client

The received RTP video packets by receiver application are investigated and required information about the network (i.e., lost packets, loss ratio, jitter, etc.) is sent back to the server side through port 5005. The received packets are extracted into coded video data by "rtp264depay" and then the coded video is decoded by "ffdec_h264" and played on the screen. Fig. A.3 shows the receiver application pipeline.

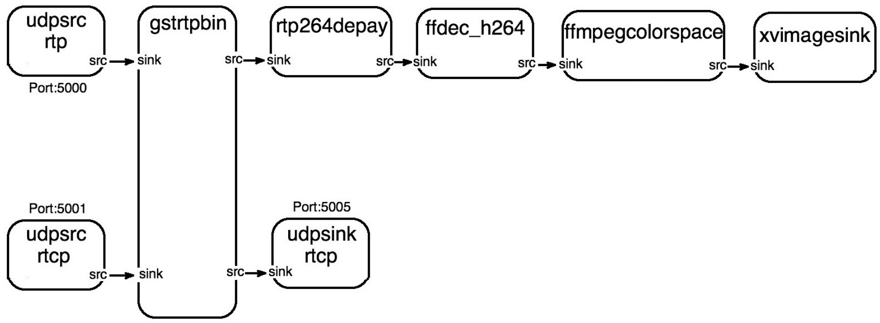


Figure A.3: Client pipeline.

Appendix B

Résumé de la thèse

B.1 Vue d'ensemble

Dans les télécommunications, la performance est évaluée en termes de qualité de service (*Quality of Service*, QoS). Elle est mesurée soit de manière intrinsèque à une technologie (par exemple, pour ATM, la perte de cellules, ou la variation de débit) [1] ou selon certains niveaux de protocole (par exemple, la perte de paquets, le délai, ou la gigue) [2]. Au temps des réseaux dédiés aux applications (p.ex. téléphonie) ou lorsque l'interconnexion de réseaux était le service, ces mesures étaient suffisantes pour caractériser la qualité et l'impact de tout potentiel négatif sur le service, ou alternativement, elles ont été utiles en tant que paramètres pour les ententes de niveau de service (SLA) entre les fournisseurs de service et les clients. Aujourd'hui, l'augmentation en nombre et performance de l'accès aux réseaux à large bande a conduit à une demande croissante pour les applications voix et vidéo sur IP (VVoIP) comme la téléphonie Internet (VoIP), la vidéoconférence et encore la télévision sur IP (IPTV). Bien que l'évaluation de la qualité de la vidéo (ou voix) des systèmes de communication ait été un sujet d'étude important pour les universités et l'industrie pendant des décennies, le déploiement de systèmes VVoIP a amené une nouvelle série de questions qui nécessitent de nouvelles méthodes d'évaluation. De plus, puisque nous sommes passés à un réseau unique pour plusieurs services, basé sur IP, il est apparu que les mesures traditionnelles de qualité de service sont insuffisantes et l'intérêt s'est déplacé vers la notion de qualité d'expérience (QoE). QoE est la performance globale d'un système du point de vue des clients. En d'autres termes, QoE est une mesure de bout en bout des performances au niveau du service pour le client, et une indication de la façon dont le système répond aux besoins du client [3]. Lorsque les clients parlent de qualité, ils tentent de décrire leur réaction ou leur satisfaction face à un ou plusieurs de ces attributs de service, en fonction de la nature de la demande, dont :

- la qualité de connexion ;
- la convivialité de la connexion ;
- la sécurité de la connexion ;
- la robustesse de la connexion ou la déconnexion;
- ... .

Par conséquent, bien que la QoE soit de par sa nature même assez subjective, il est très important qu'une stratégie soit conçue pour la mesurer de façon aussi réaliste que possible. La capacité de mesurer la QoE donnera au fournisseur de service un certain sens de la contribution de la performance du réseau au niveau global de satisfaction de sa clientèle en termes de fiabilité, de disponibilité, d'évolutivité, de vitesse, de précision et d'efficacité. Par conséquent, même si une infrastructure de service a été bien conçue, il faut mesurer la QoE livrée, ce qui implique de nombreuses difficultés pratiques. Le codage de l'information est souvent intrinsèquement un processus de réduction de la qualité, et donc la qualité transmise au client n'est pas optimale, indépendamment de tout incident de transmission. Ce qui est perdu, endommagé ou retardé a également un impact sur la qualité, car tous les éléments d'information sont considérés comme égaux. Pour le transfert unidirectionnel temps réel (par exemple, le streaming vidéo) perte d'information et codage sont les facteurs dominants, mais pour les communications bidirectionnelles en ligne (par exemple, une conversation au cours de la conférence sur Internet ou vidéo) d'autres paramètres tels que le délai et la gigue (variation du délai) peuvent être aussi importants que le codage et la perte [3].

La croissance exponentielle de l'utilisation des applications multimédias sur Internet et l'importance de la QoE pour mesurer la performance des services multimédias ont motivé ce travail de recherche. Dans ce chapitre, nous présentons d'abord les questions de recherche qui motivent ce projet, et ensuite les objectifs de recherche et nos contributions principales. Le reste de ce chapitre est organisé comme suit: les motivations de cette recherche sont présentés à la section 1.2. La section 1.3 présente les objectifs de recherche. Les contributions principales de cette recherche sont résumées à la section 1.4. La section 1.5 décrit brièvement les grandes lignes de la thèse.

B.2 Motivation

La QoE n'est pas un concept nouveau. Elle a sa place depuis longtemps dans la téléphonie où il était important de mesurer la satisfaction des clients du service, et cela a été fait avec des expériences subjectives avec un grand nombre d'utilisateurs et défini en termes de note moyenne d'opinion de qualité, soit le *Mean Opinion Score* (MOS), de valeurs mauvaise à excellente. Cependant, les techniques pour mesurer la qualité subjective ne peuvent pas être utilisées dans les essais à grande échelle en raison de leur mise en œuvre laborieuse, le coût élevé de l'écoute des experts, et leur caractère non-répétitif [4, 5]. En outre, pour le dépannage proactif de la performance des goulets d'étranglement de VVoIP, qui se manifestent comme des déficiences de rendement tels que le gel d'image vidéo et la décrochage de la voix, les opérateurs de réseaux ne peuvent pas compter sur de véritables utilisateurs finaux pour signaler leur qualité subjective de la perception. Par conséquent, des techniques automatisées et objectives qui fournissent en temps réel ou en ligne des estimations perceptives de qualité VVoIP sont essentiels [5].

La QoE combine les paramètres non-techniques tels que la perception du client, l'expérience et ses attentes, ainsi que des paramètres techniques dont la QoS de l'application et du réseau. Du point de vue du client, la partie QoE-technique pour la transmission multimédia sur Internet peut se résumer en une chaîne de valeurs qui comprend les éléments suivants [6] :

- les fournisseurs de contenu multimédia, les serveurs, les applications multimédia en continu, la préparation des données multimédia, etc. ;
- les fournisseurs de réseaux et de services, les déficiences du réseau, etc. ;
- les dispositifs de l'utilisateur, les applications de lecture, etc.

Afin de gérer la qualité perçue par l'utilisateur, il est essentiel de comprendre la relation quantitative entre la QoE et tous ces paramètres techniques de la chaîne de production de la QoE.

La relation entre la qualité de service (application et réseau) et la QoE aide les fournisseurs de réseaux et de services à gérer les paramètres de qualité de service et les fournitures de services efficaces et efficientes afin de fournir une meilleure qualité d'expérience pour les utilisateurs d'une manière rentable, compétitive et efficace. La première étape de ce processus consiste à mesurer le niveau de satisfaction de l'utilisateur final de la qualité de service, tandis que les paramètres de qualité de service de l'application et du réseau sont surveillés et mesurés.

Les techniques contemporaines pour mesurer la qualité perçue sont divisées en mesures subjectives et objectives [4]. Les techniques d'évaluation subjective, donc effectuées par des utilisateurs humains, pour classer la vidéo, l'audio ou la qualité des données peuvent fournir l'évaluation la plus précise de la qualité de sortie du point de vue des clients d'un fournisseur de services. Toutefois, en raison de leurs inconvénients déjà mentionnés au début de cet article, des tests objectifs, plus pratiques [4, 5, 7], sont préférés pour prédire la perception de l'utilisateur final. En règle générale, l'évaluation objective de la qualité perceptive nécessite une comparaison des informations sur la source et la destination [2, 3, 8]. Pour prévoir la qualité, il faudrait connaître la source des données, ainsi que les effets que la propagation du réseau peut avoir sur les données. Par exemple, la majorité des modèles et des systèmes qui existent pour estimer la qualité vidéo dans les réseaux par paquets nécessitent généralement une connaissance détaillée du contenu vidéo et des fonctionnalités, et reposent souvent sur l'inspection approfondie des paquets vidéo [7, 9]. Ces techniques peuvent être appelées des techniques hors ligne car (a) elles nécessitent un alignement temporel et spatial de l'information originale et reconstruite, ce qui prend beaucoup de temps à réaliser, et (b) (pour la transmission vidéo), elles requièrent de nombreux calculs en raison de leur traitement par pixel des séquences vidéo. Une autre série de tests objectifs, appelés tests objectifs indirects, utilisent des mesures des déficiences du réseau (perte, retard, gigue, durée de la perturbation) pour estimer l'impact sur la qualité (vidéo ou audio), et peuvent-potentiellement-être effectués en ligne. Ces techniques sont applicables là où il existe une relation établie entre QoE et QoS [3].

Dans l'ensemble, on peut conclure que QoS et QoE sont deux concepts interdépendants dans les transmissions multimédias modernes sur des réseaux IP et, par conséquent, ils devraient être étudiés et gérés dans une démarche conjointe, de la planification à la mise en œuvre et l'ingénierie (optimisation). En outre, bien que le champ de recherche en qualité de service ait été largement étudié, mesurer les déficiences du réseau pour améliorer la qualité d'expérience n'en est pas moins un domaine de recherche ouvert qui est abordé dans cette thèse.

Les effets des paramètres QoS sur QoE ont été étudiés à partir de perspectives différentes qui sont examinées dans le chapitre 2. Par exemple, les effets de perte de bits / paquets /

trames et la longueur de la perte sur la qualité de la vidéo ont été discutés dans [10-15], et différents modèles de qualité ont été proposés. Toutefois, les modèles existants n'ont pas pris en compte tous les aspects de la perte des effets sur la qualité vidéo perçue. Par exemple, les résultats des modèles de l'impact de la propagation de l'erreur due à une perte de trame sur la qualité perçue de la vidéo compressée transmise n'ont pas une corrélation acceptable avec les résultats expérimentaux pour toutes les variantes de contenu. Notre recherche étudie cette question et se concentre spécifiquement sur l'effet de la position de perte de trame sur les vidéos avec différents types de contenus.

Les techniques de mesure de la QoE peuvent avoir des usages multiples. Elles peuvent être utilisées dans la conception du réseau ou pour choisir des codecs appropriés (par exemple, pour la voix) pour répondre aux exigences minimales de qualité. De même, dans le cas de la vidéo (ex: TV) en service de streaming, les techniques de mesure de la QoE peuvent aider à déterminer dans quelle mesure le réseau prend en charge la prestation d'un certain niveau de qualité. Elles peuvent également être utilisées à des fins de surveillance, soit dans le cadre des accords de niveau de service (SLA) ou tout simplement pour l'évaluation de la qualité. Enfin, elles peuvent être exploitées de manière adaptative pour protéger la qualité d'un service.

La vidéo en continu (live) est actuellement couramment utilisée sur Internet, et il est également prévu que le chat vidéo sera l'une des applications de prédilection pour les services mobiles via des communications sans fil (par exemple, 3G et 4G). Pour répondre aux attentes de leur clientèle, les fournisseurs de services doivent connaître le niveau de qualité qui est jugé acceptable par leurs clients. Sur la base de ces informations, les fournisseurs de services peuvent efficacement gérer et contrôler les ressources de leurs réseaux. Cependant, la gestion et le déploiement de plus de ressources augmente non seulement les coûts, mais n'est aussi parfois pas possible (par exemple, dans un environnement mobile, la bande passante ne peut pas être supérieure à un certain niveau). Par conséquent, il appert que la conception d'applications intelligentes, qui peuvent s'adapter dynamiquement avec les réseaux existants par la gestion du système vidéo (par exemple, le débit binaire) sans effet négatif sur la qualité perçue pour les utilisateurs finaux, est devenue un enjeu extrêmement important. En d'autres termes, la gestion de la QoE par des applications vidéo a pour but de conduire à un déploiement plus efficace et économique des ressources du réseau tout en gardant la satisfaction de l'utilisateur final à un niveau acceptable. En outre, étant donné que la coopération étroite des serveurs / applications avec les fournisseurs de services n'est généralement pas possible, il est nécessaire de développer de nouvelles applications multimédias qui peuvent s'adapter à l'environnement réseau existant, au comportement du meilleur effort (best effort) ainsi que la nature concurrentielle de l'Internet pour offrir la meilleure qualité de perception pour les utilisateurs finaux. Pour notre contribution sur ce thème, nous avons étudié comment les paramètres vidéo affectent le débit et la qualité vidéo. Notre étude porte spécifiquement sur les applications de vidéo en continu qui utilisent la taille QCIF et sont codées en H.264 (le type de vidéo le plus omniprésente utilisé dans les applications de visioconférence) sur les réseaux à bande passante limitée.

B.3 Objectifs

Les méthodes de mesure et de contrôle de qualité de service existantes dans les réseaux IP, selon les recommandations de l'ISI ne reflètent pas la satisfaction des usagers des services. Ainsi, pour améliorer la qualité de perception de l'utilisateur final, nous avons l'intention de concevoir un système de contrôle de l'interfonctionnement entre la qualité expérimentée, la transmission des paramètres de qualité de service, et la couche application sur le serveur et le client (par exemple, les spécifications vidéo, le codage). Ce système de contrôle est utilisé dans l'application de vidéo en continu pour optimiser la qualité de la vidéo perçue selon les conditions de réseau et pour gérer l'utilisation des ressources disponibles. En raison de l'ampleur du projet, nous considérons que les scénarios où la QoS et la QoE surveillées par les communications se produisent entre un client et un serveur.

Comme première étape, nous cherchons à étudier les facteurs qui affectent la qualité multimédia perçue par les utilisateurs finaux, la façon dont ces facteurs sont mesurés ou prédits, et comment les différents modes explicites ou implicites de l'échange d'informations (par exemple, le protocole RT(C)P) peuvent être utilisés pour détecter les variations de la qualité de perception. En d'autres termes, nous voulons définir les caractéristiques qui devraient être suivies et mesurées, et ensuite, nous avons l'intention de trouver la corrélation entre les caractéristiques mesurées et la perception des utilisateurs à comprendre le rôle de chaque élément d'information (comportement du réseau et ses caractéristiques) de la qualité expérimentée. En particulier, notre objectif est de contrôler et de mesurer la perte de paquets et le délai uni-directionnel (one-way delay, OWD) comme paramètres de qualité de service qui affectent la QoE.

Comme la plupart des recherches récentes et des méthodes proposées pour l'évaluation de la qualité perçue (par exemple, le E-Model) sont insuffisantes pour capturer les compromis parmi les facteurs qui ont une incidence [2] sur la qualité de service, notre projet de recherche vise à évaluer la mesure dans laquelle chaque caractéristique mesurée affecte la perception des clients en premier lieu. L'une des contraintes les plus importantes de notre recherche est que toutes les procédures doivent être en ligne et nous devons trouver des méthodes pour surveiller et mesurer les facteurs nécessaires et puis estimer la qualité perçue, en temps réel. En d'autres termes, nous avons l'intention d'expliquer la qualité perçue en fonction de certains paramètres de qualité de service en temps réel, dans des contextes spécifiques.

En raison du large domaine de variation de qualité de service, le suivi des fluctuations QoE en fonction des paramètres de qualité de service serait impossible ou très compliqué. Par conséquent, notre objectif est d'explorer la relation entre la QoE et certains paramètres spécifiques de QoS (par exemple, la perte et la bande passante). Dans la mesure où l'emploi de l'Internet comme infrastructure de transmission pour les vidéos est un axe de développement majeur dans l'industrie et joue un rôle important dans les réseaux de prochaines générations (NGN), nous nous concentrons d'abord sur la détermination de la qualité perçue en fonction du modèle de perte pour les services vidéo tels que les services de vidéoconférence. Par conséquent, nous avons l'intention d'enquêter sur la façon dont le modèle de perte affecte la qualité de la vidéo et de façon plus explicite, si la position du paquet / perte de trame est important. En répondant à cette question, nous proposons un modèle de qualité vidéo en fonction de la position de perte de trame pour différents types de vidéo. Sur base de notre étude sur l'effet des différents schémas de pertes sur la qualité de la vidéo, nous allons ensuite comprendre comment il est possible pour les applications des utilisateurs finaux de minimiser la dégradation de la qualité.

Étant donné que le débit binaire vidéo varie en raison de différentes caractéristiques telles que le débit de trame vidéo, la résolution, le niveau de compression, le contenu, etc., des conditions de réseau identiques peuvent amener les utilisateurs finaux à percevoir différents niveaux de qualité pour différentes vidéos. A ce stade, cette recherche vise à répondre à deux questions principales : ≪quelle est la qualité de la vidéo réelle perçue lorsque les paramètres vidéo sont modifiés pour répondre aux limites en bande passante ? ≫, et ≪Quels sont les meilleurs paramètres vidéos pour un débit binaire vidéo spécifique étant donné la qualité subjective perçue par les utilisateurs finaux ? ≫. Cette thèse se concentre sur l'étude de l'effet de différents facteurs tels que le taux de trame et de quantification (QP) sur le taux de bit des données de vidéo et la qualité de vidéo perçue et, par conséquent, sur le contrôle de la qualité d'expérience (QoE) avec des paramètres vidéos en fonction des limitations de bande passante imposées par le réseau. Comme mentionné précédemment, un de nos objectifs est d'étudier les différentes méthodes de mesure de bande passante. Nous procédons ensuite à lier ces mesures à des actions au niveau du serveur pour maintenir la qualité perceptive des vidéos transmises à l'intérieur des limites acceptables (garanties de performance). En effet, un système de contrôle, en rétroaction sur l'état du réseau (par exemple, la bande passante estimée), contrôle et vérifie le débit en continu en utilisant une combinaison de caractéristiques des mécanismes de contrôle de codec.

Pour conclure, nos objectifs sont les suivants:

- la définition de la QoE et étude sur ses méthodes de mesures ;
- l'étude et l'amélioration des méthodes d'estimation de probabilité de perte de paquets ;
- l'étude et l'amélioration des méthodes d'estimation du délai de communication unidirectionnel ;
- l'étude des effets de la perte de paquet ou de trame sur la qualité multimédia / vidéo en fonction des caractéristiques du codec ;
- la mesure robuste (subjective) de la qualité perçue des vidéos avec différentes caractéristiques (paramètre de quantification et de taux de trame) ;
- l'étude des différents mécanismes de contrôle de taux binaire vidéo / modèles de gestion de la qualité perçue par l'utilisateur final.

B.4 Contributions

Les contributions importantes de cette recherche sont les suivantes.

- Après avoir examiné les méthodes d'estimation en ligne de la probabilité de perte de paquets (*plp*) existantes, nous proposons une nouvelle méthode d'approximation précise de la *plp* à un nœud relais intermédiaire, où un grand nombre de sources devraient être agrégées. Suivant le principe de grandes déviations, en modélisant le trafic entrant d'un intermédiaire à grande vitesse comme un processus gaussien, nous avons introduit une nouvelle approximation pour *plp*. Par la combinaison de cette approximation en ligne avec la mesure du trafic de sortie hors ligne, nous avons proposé un estimateur *plp* qui améliore considérablement la qualité de l'estimation par rapport aux estimateurs *plp* proposés les plus récents, qui reposent sur des bases théoriques similaires.

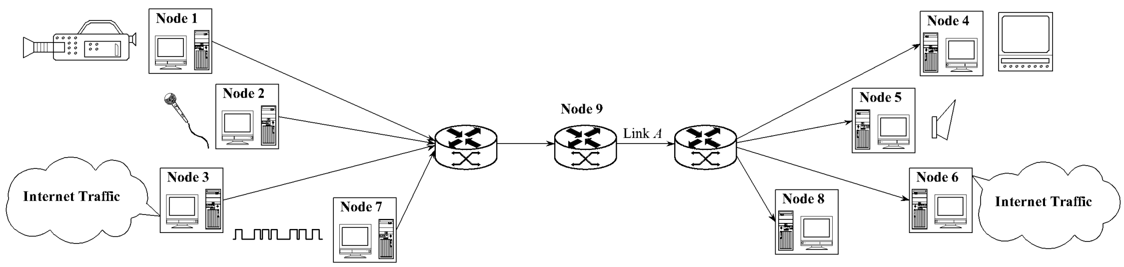


Figure B.1: Topologie du banc d'essai.

Pour étudier la précision des estimations, nous avons utilisé le simulateur NS-2 avec un trafic d'entrée est très similaire au trafic Internet au niveau du nœud de mesure. La topologie de réseau qui est simulée est représentée sur la figure B.1.

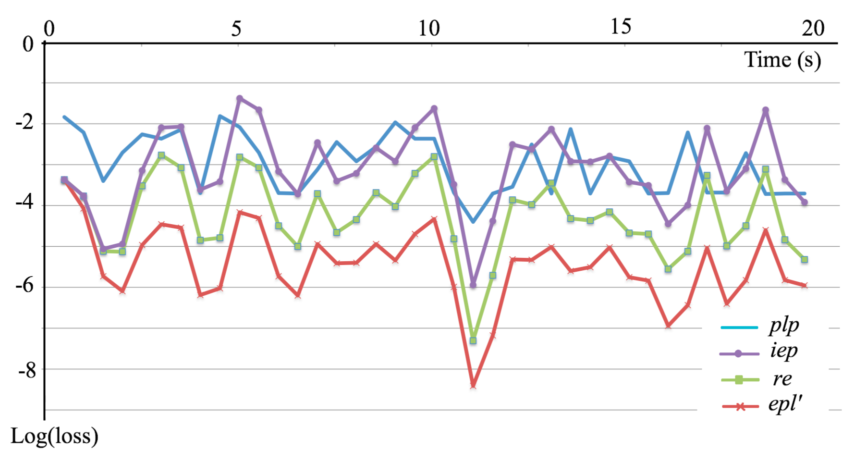


Figure B.2: Mesure et estimation de la probabilité de perte pour un plp de -2,5.

Dans l'ensemble, les résultats des simulations illustrent bien l'effet des configurations différentes, telles que la taille du tampon, sur les estimations (figures B.2 et

B.3). L'analyse des résultats montre l'amélioration de la précision dans l'estimation *plp* réalisée par notre nouvelle méthode de calcul.

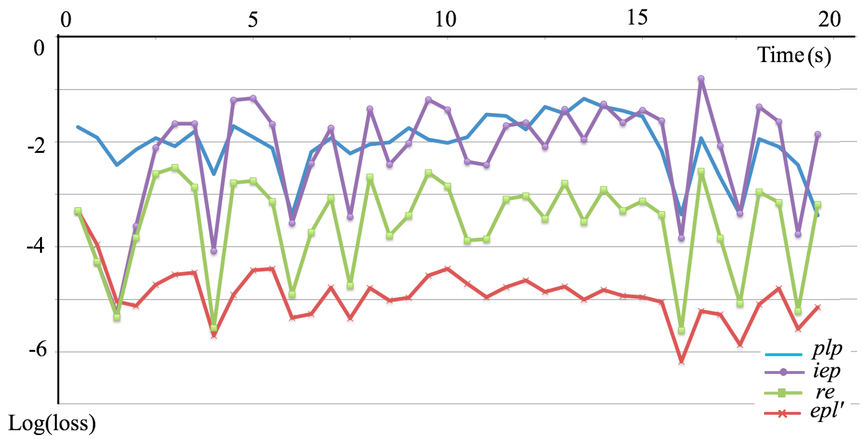


Figure B.3: Mesure et estimation de la probabilité de perte pour un plp de -1,5.

Pour conclure, les avantages de notre estimateur sont les suivants : 1) une augmentation de la précision de l'estimation en utilisant les paramètres mesurés de façon correcte, 2) la flexibilité de la durée de la mesure de l'intervalle de temps, et 3) une estimation assez précise de *plp* dans le cas d'un petit tampon.

(Les publications associées sont [16, 17].)

- Différentes méthodes d'estimation / mesure du délai unidirectionnel (OWD) ont été étudiées. Une méthode récemment proposée pour estimer l'OWD a été analysée et améliorée. Cette méthode est basée sur la réalisation de plusieurs mesures de RTT parmi les paires de nœuds et en appliquant la méthode d'estimation des moindres carrés pour trouver l'approximation la plus raisonnable de l'OWD entre deux nœuds spécifiques. Cela se fait par la mesure de tous les RTT possibles entre les deux nœuds et un nœud tiers auxiliaire (figure B.4). Pour obtenir des estimations plus précises, certaines contraintes supplémentaires ont été ajoutées aux équations régulières du chemin cyclique.

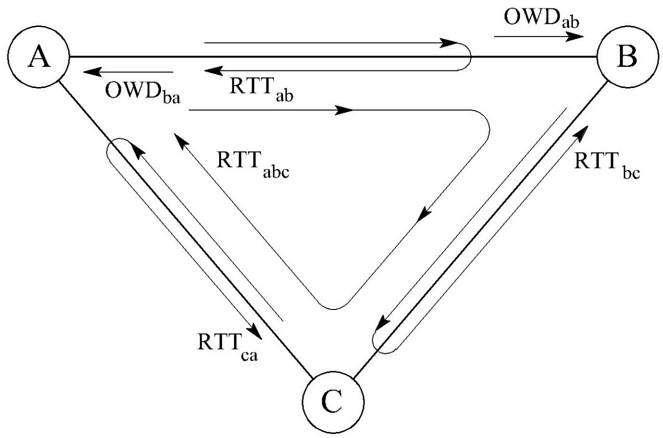


Figure B.4: Modèle de mesure avec tiers.

Toutes les nouvelles contraintes peuvent être facilement estimées sur la base du comportement connu de nœuds et de leurs chemins de connexion. Pour mesurer avec précision le délai de transmission, qui est souvent une contrainte asymétrique, une méthode simple a été mise en place. Il a été démontré que toutes les méthodes proposées sont exemptes d'effets dus au décalage de l'horloge. Pour comparer la précision des résultats obtenus par la méthode proposée, avec les modèles du chemin cyclique et la division traditionnelle du RTT, un réseau simple à trois nœuds a été simulé et étudié dans des situations différentes (figure B.5). Les résultats ont confirmé l'amélioration des erreurs d'estimation dans le modèle proposé. En outre, l'influence de différents types de contraintes ajoutées au modèle du chemin cyclique a été examinée. Il a été démontré que les contraintes asymétriques sont plus efficaces dans l'amélioration des résultats que les symétriques et comment elles peuvent être efficacement estimées.

Pour examiner l'exactitude de notre méthode proposée pour mesurer le retard de transmission, une topologie du réseau qui est montrée dans la figure B.6 a été simulée.

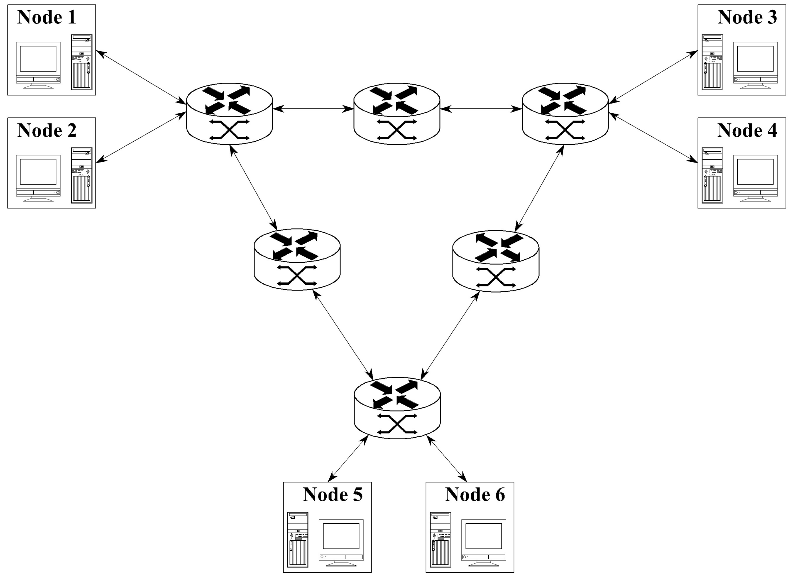


Figure B.5: Topologie de test pour la simulation du chemin cyclique.

La figure B.7 montre la précision de la méthode proposée pour calculer le délai de transmission des paquets avec le codec G.711 (taille du paquet de 250 octets) entre les nœuds 1 et 2, pour des temps de calcul différentes. La figure montre que l'erreur de calcul est inférieure à 1 pour cent, et diminue avec le temps.

(La publication associée est [18].)

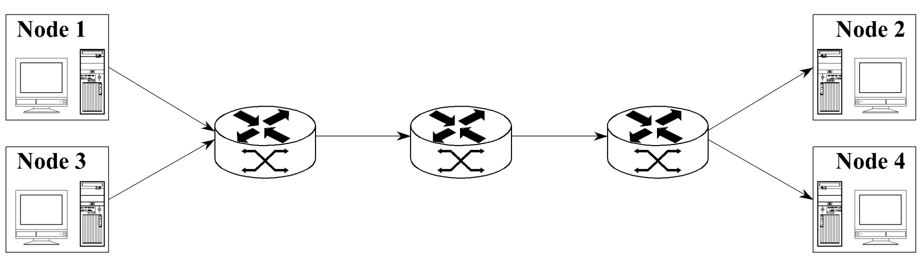
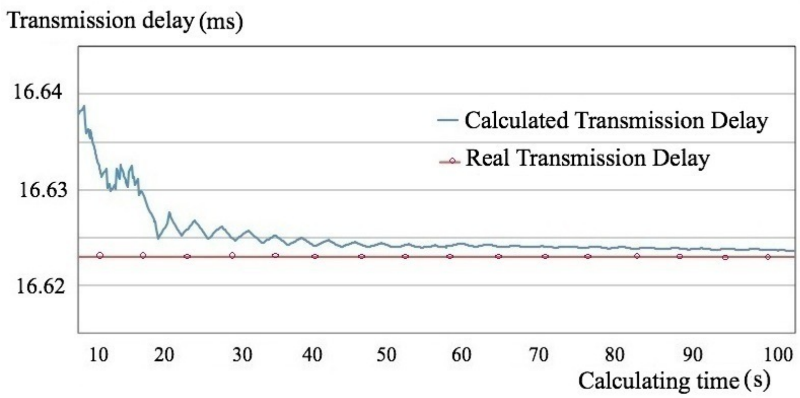
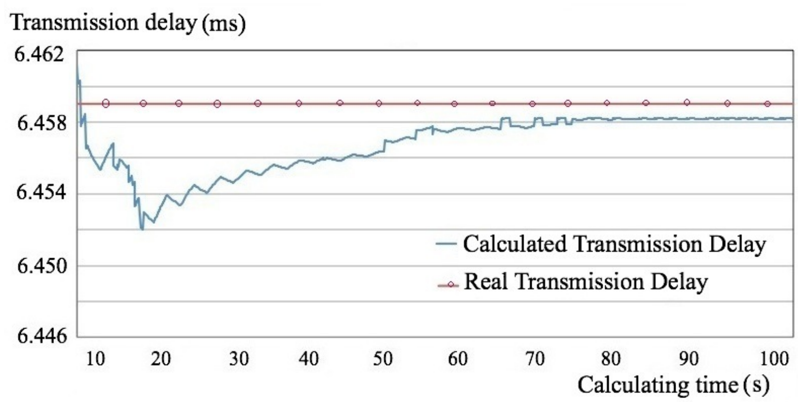


Figure B.6: Topologie de test pour la simulation de la mesure du délai de transmission.



(a) Transmission delay from node 1 to node 2;



(b) Transmission delay from node 2 to node 1;

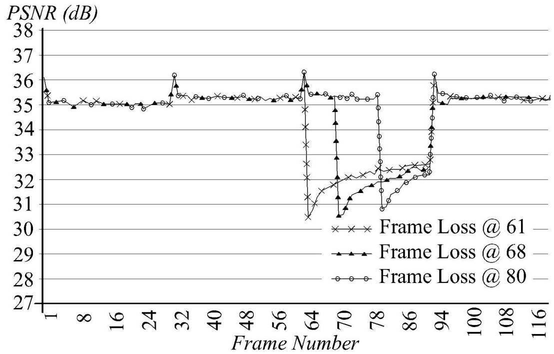
Figure B.7: Délais de transmission aller et retour calculés pour des paquets G.711 (i.e., 250 bytes).

- Cette recherche porte sur la question de savoir si ou non un paquet ou trame perdu(e), et en particulier sa position par rapport aux trames de type I (intra), influence la qualité de la vidéo codée transmise. En utilisant comme mesure le rapport signal-bruit de crête (PSNR) de la vidéo reçue pour mesurer le degré de distorsion, nous étudions l'effet de la position relative de la perte de trames I sur la distorsion totale pour les vidéos. Sur la base de nos résultats empiriques (figure B.8), nous avons proposé un modèle pour estimer le PSNR des trames reçues avec une qualité dégradée par la propagation de la distorsion. Pour obtenir des estimations plus exactes de PSNR (plus proches des valeurs réelles), deux nouveaux facteurs ont été utilisés : une fonction exponentielle et un facteur d'atténuation conditionnel qui suivent le filtrage spatial et la mise à jour des trames I, respectivement. Selon les simulations effectuées pour trois types de vidéos (pour des mouvements faible, moyen et élevé), nous pouvons conclure que la propagation de distorsion estimée par le modèle proposé est beaucoup plus précise que ceux estimés par d'autres méthodes récentes (figures B.9 et B.10). En outre, le modèle proposé peut être utilisé pour estimer la qualité moyenne de la vidéo, si la position (l'index) de la trame perdue est connue (figure B.11).

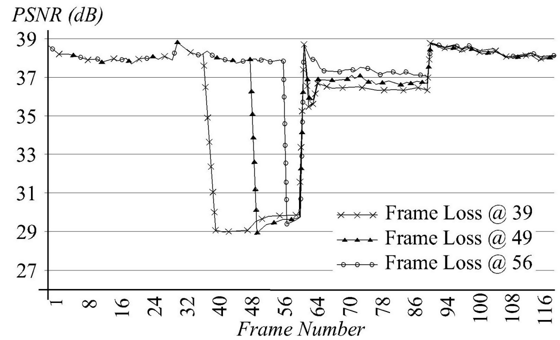
  En outre, après avoir étudié des séquences de perte dans des environnements bruités où la durée de la perte de données est presque constante (figure B.12), nous proposons une méthode pour améliorer la performance de la vidéo en continu sur des canaux bruités basés sur l'ordonnancement des paquets (figure B.13), sans augmentation de la vitesse de transmission.

  (Les publications associées sont [19, 20].)

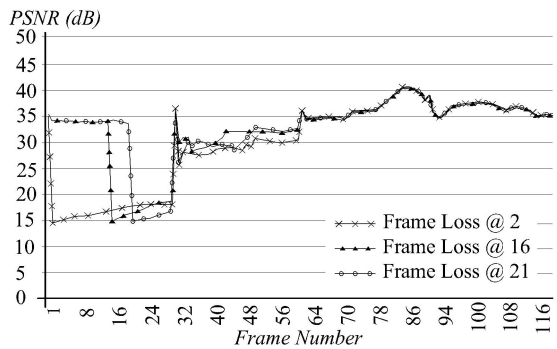
- Pour atteindre notre dernier objectif, nous avons réalisé des mesures approfondies pour étudier l'effet des différents paramètres de contrôle (taux de trame-key rate-et facteur de quantification QP) du taux de bits limité par la bande passante du réseau



(a) $PSNR$ of *Bridge-Close* video which encounters a single loss at frame 62, 68 and 80.



(b) $PSNR$ of *News* video which encounters a single loss at frame 39, 49 and 56.



(c) $PSNR$ of *Football Game* video which encounters a single loss at frame 2, 16 and 21.

Figure B.8: Calcul du PSNR pour trois types de vidéo avec une perte de trame unique.

(figures B.14 et B.15). En outre, pour quantifier la qualité de vidéo perçue par les utilisateurs finaux, une étude basée sur des tests subjectifs a été réalisée. Les résultats démontrent la relation entre les paramètres principaux influant la vidéo et la qualité vidéo vécue par les utilisateurs finaux. En termes simples, le QP et le taux de trame agissent différemment dans la qualité perçue des vidéos de mouvement moyen (figure B.16). L'effet du QP sur la qualité perçue par les utilisateurs finaux est plus significatif à faible débit avec un QP de plus de 24, alors que le taux de trame affecte le QoE de façon plus marquée lorsque le débit binaire est élevé et que le QP est inférieur à 24. Ce phénomène est plus apparent pour les vidéos de plus grande taille.

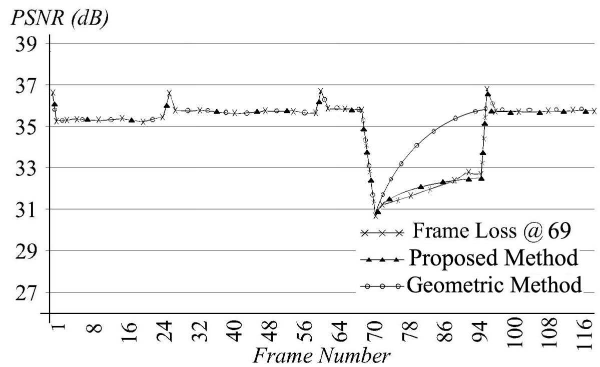


Figure B.9: Comparaison des méthodes proposées et géométriques pour l'estimation du PSNR (*Bridge-Close*)

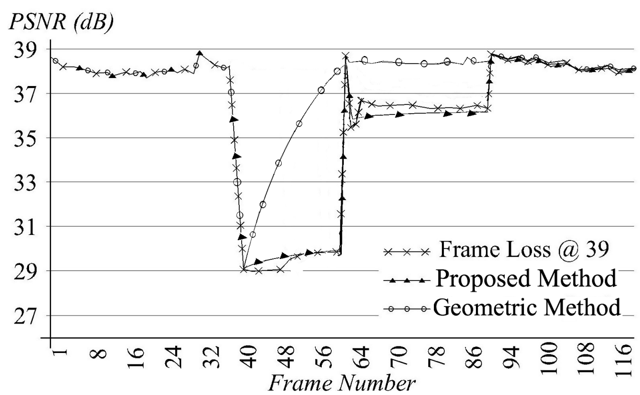


Figure B.10: Comparaison des méthodes proposées et géométriques pour l'estimation du PSNR (*News*)

Nous avons utilisé les résultats de tests subjectifs pour trouver les paramètres optimaux de vidéo basés sur la bande passante du réseau donné et un niveau acceptable de qualité de perception et, enfin, nous proposons un algorithme de contrôle de la qualité vidéo perceptuelle basé sur les mesures mentionnées. Le pseudo-code pour le système proposé de contrôle QoE est présenté sous la forme de l'algorithme 2, en annexe de cette thèse.

Contrairement à d'autres études similaires, nous nous sommes spécifiquement con-

```
if (Available_BW <Critical_Value) then
    Re-Optimize the Network;
end
if (R(i+1) <R(i)) then
    if (Current_QP <24) then
        Increase the QP;
    else
        if (Current_Frame-Rate >Critical_frame-Rate_Point) then
            Decrease the Frame-Rate;
        else
            Increase the QP;
        end
    end
else
    if (Current_QP <24) then
        if (Current_Frame-Rate <30) then
            Increase the Frame-Rate;
        else
            Decrease the QP;
        end
    else
        Decrease the QP;
    end
end
```

Algorithm 2: Pseudo code of the proposed QoE control.

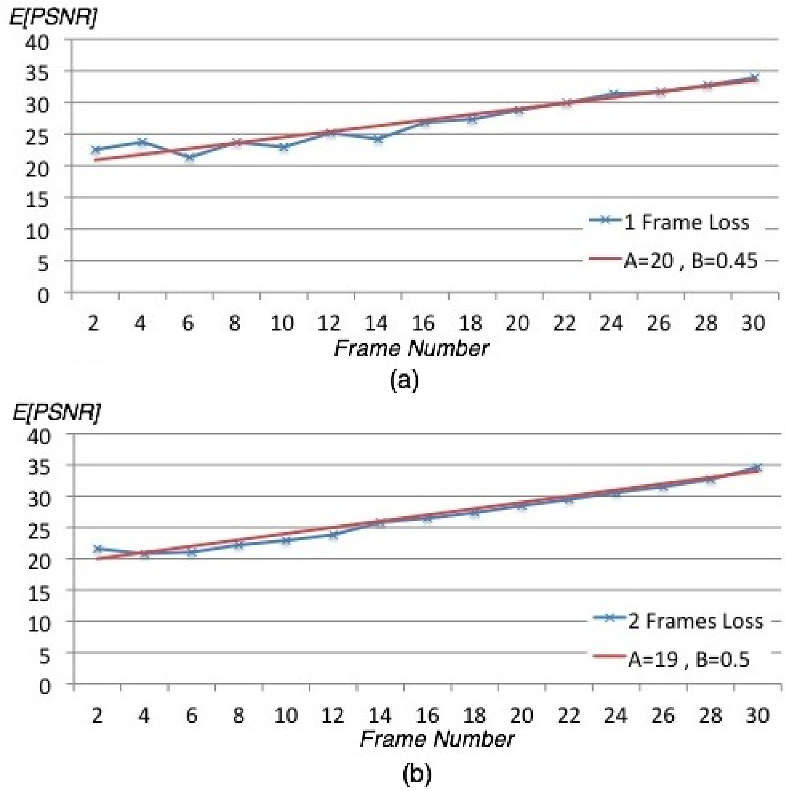


Figure B.11: PSNR moyen pour différentes positions de perte par rapport à la trame intra précédente.

centrés sur les vidéos de mouvement moyen avec les formats vidéo QCIF, CIF, et VGA, formats les plus répandus utilisés par les applications de visioconférence sur Internet et les systèmes de communication mobiles.

(Les publications associées sont [21, 22].)

B.5 Plan du document

Cette thèse est structurée comme suit. Le chapitre 2 donne une définition exhaustive de la QoE et quelques informations sur ses méthodes de mesure. La section 2.2 décrit le concept

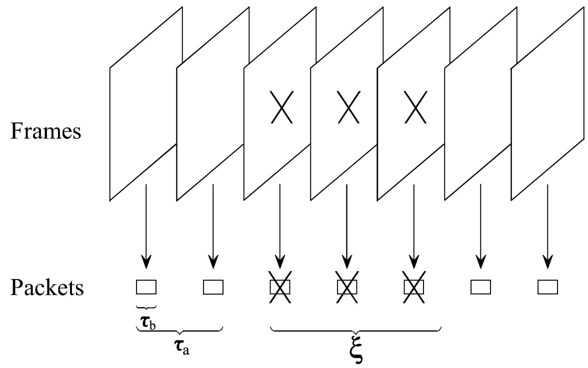


Figure B.12: Modèle de perte de paquet.

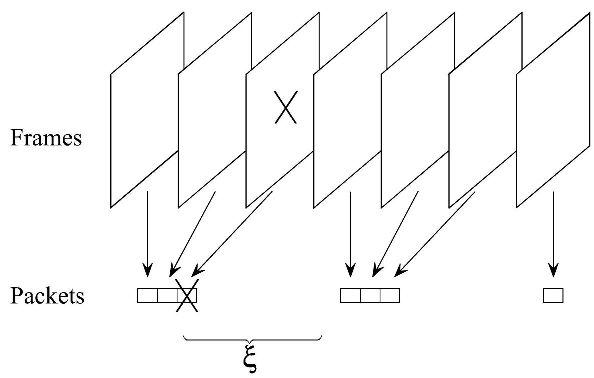
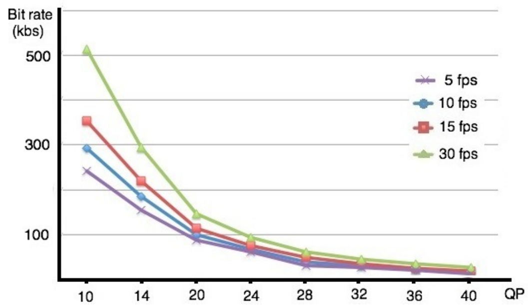


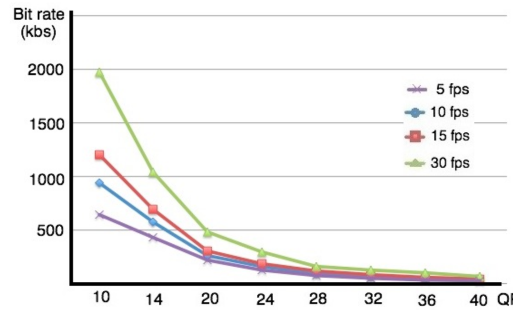
Figure B.13: Modèle de transmission en rafale en environnement bruité.

de deux types de mesures de la qualité de perception: subjectif et objectif. Dans la section 2.3 les paramètres du réseau et leurs effets sur la qualité multimédia perçue par l'utilisateur final sont décrits.

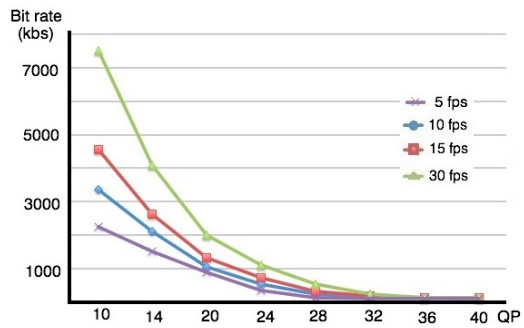
Le chapitre 3 examine comment estimer avec précision la probabilité de perte de paquets au niveau d'un nœud à haute vitesse intermédiaire dans le réseau Internet en temps réel. Ce chapitre se poursuit dans la section 3.2 en examinant les travaux antérieurs sur la mesure ou l'estimation de la probabilité de perte de paquets. Dans la section 3.4, nous développons un nouvel estimateur de la *plp*. Les sections 3.5 et 3.6 présentent les simula-



(a) Bit rate versus QP for QCIF-size video (Akiyo).



(b) Bit rate versus QP for CIF-size video (Akiyo).

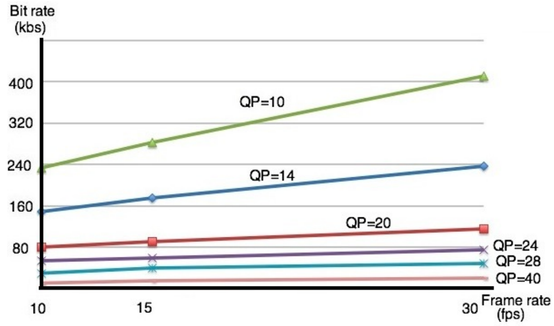


(c) Bit rate versus QP for VGA-size video (Foreman).

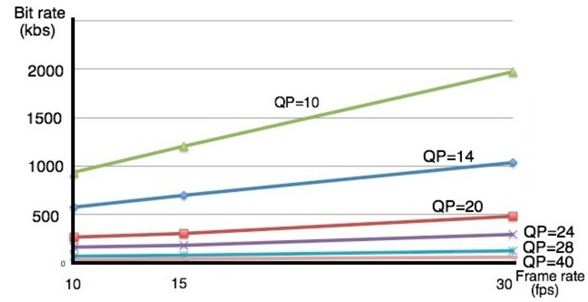
Figure B.14: Débit pour différentes valeurs de QP pour formats QCIF, CIF, VGA.

tions et leurs résultats numériques pour démontrer l'efficacité de notre estimateur.

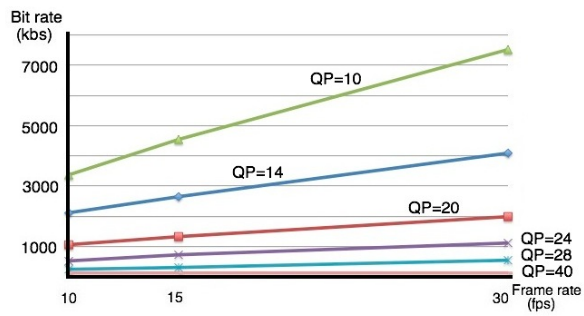
Le chapitre 4 résume l'état de l'art des méthodes de mesure du délai unidirectionnel. La section 4.2 passe en revue les travaux précédents sur la mesure ou l'estimation du délai unidirectionnel. Dans la section 4.3, le modèle à trois nœuds, la méthode du chemin cyclique/LSE, et les améliorations proposées sont expliquées. Dans la section 4.4, une méthode pour mesurer le délai de transmission entre deux nœuds d'extrémité est introduite. Des simulations numériques et les résultats démontrent l'amélioration du modèle proposé par rapport à d'autres modèles dans la section 4.5. Le niveau de précision de la méthode proposée pour mesurer le retard de transmission est également démontré dans cet



(a) Bit rate versus frame rate for QCIF-size video (Akiyo).



(b) Bit rate versus frame rate for CIF-size video (Akiyo).



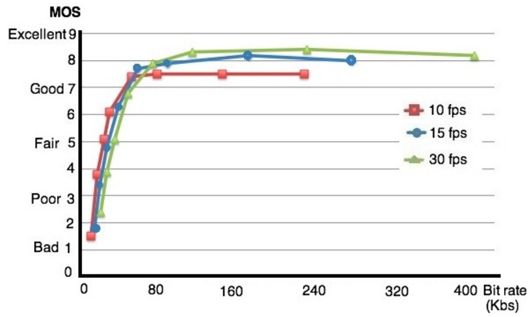
(c) Bit rate versus frame rate for VGA-size video (Foreman).

Figure B.15: Débit contre taux de trame pour différentes valeurs de QP.

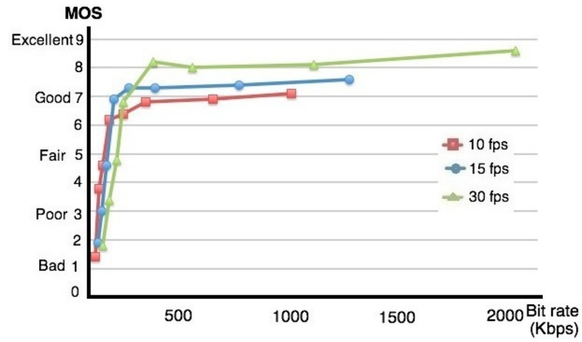
article.

Le chapitre 5 présente un modèle pour estimer la dégradation de la qualité de vidéo en fonction de la position de perte de trame. Les modèles précédents pour estimer la distorsion produite par la perte de paquets sont examinés à la section 5.2.

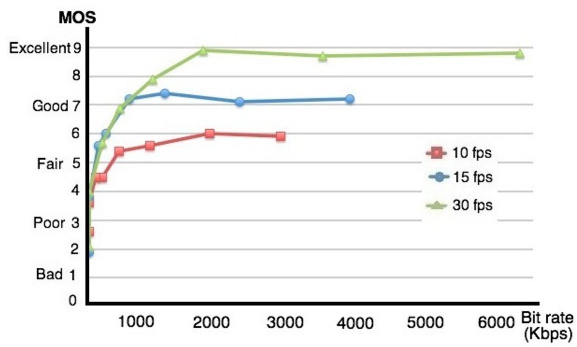
Dans la section 5.3, nous décrivons l'effet de la position de trame perdue par rapport à des trames intra (I) sur le PSNR moyen et nous en tirons un modèle qui estime la distorsion totale propagée. La précision des estimations du modèle est démontrée par des simulations dans la section 5.4. L'effet de la planification de transmission par paquets sur la performance de canal bruité est examiné à la section 5.5.



(a) MOS versus bit rate for QCIF-size video (Akiyo).



(b) MOS versus bit rate for CIF-size video (Akiyo).



(c) MOS versus bit rate for VGA-size video (Foreman).

Figure B.16: MOS contre débit pour différents taux de trame.

Le chapitre 6 étudie le rapport entre le codage vidéo et les paramètres de qualité de vidéo perçue. La section 6.2 présente les différentes méthodes de contrôle de congestion en temps réel de la transmission multimédia ainsi que des études récentes concernant l'effet des paramètres vidéo sur la qualité perçue. La section 6.3 présente les résultats pour les différents paramètres de codage vidéo. Les détails des tests de mesure de la qualité vidéo subjectives et leurs résultats sont présentés dans les sections 6.4 et 6.5. Dans la section 6.6 notre algorithme de contrôle de la qualité perceptive est proposé. Des simulations numériques et les résultats démontrent l'efficacité de l'algorithme proposé par rapport aux alternatives publiées dans la section 6.7.

Enfin, nous présentons les conclusions de nos travaux et proposons des pistes de recherches futures dans le chapitre 7.

Bibliography

[1] D. McDysan, *QoS & traffic management in IP & ATM networks*. New York: McGraw Hill, 2000.

[2] W. C. Hardy, *VoIP service quality: measuring and evaluation packet-switched voice*. McGraw-Hill, 2003.

[3] T. Rahrer, R. Fiandra, and S. Wright, *Triple-play services Quality of Experience (QoE) requirements and mechanisms - for architecture & transport*, DSL Forum, 2006.

[4] B. Sat and B. W. Wah, "Analyzing voice quality in popular VoIP applications," *IEEE MultiMedia*, vol. 16, no. 1, pp. 46–59, Jan.-Mar. 2009.

[5] P. Calyam, E. Ekici, C. Lee, M. Haffner, and N. Howes, "A GAP-Model based framework for online VVoIP QoE measurement," *Communications and Networks*, 2007.

[6] D. Soldani, M. Li, and R. Cuny, *QoS and QoE management in UMTS cellular systems*. John Wiley & Sons, 2006.

[7] S. Tao, J. Apostolopoulos, and R. Guérin, "Real-time monitoring of video quality in IP networks," *IEEE/ACM Transactions on Networking*, vol. 16, no. 5, pp. 1052–1065, Oct. 2008.

[8] D. A. Rosenthal, "Analyses of selected variables effecting video streamed over IP," *International Journal of Network Management*, vol. 14, no. 3, pp. 193–211, May 2004.

[9] A. Reibman, V. Vaishampayan, and Y. Sermadevi, "Quality monitoring of video over a packet network," *IEEE Transactions on Multimedia*, vol. 6, no. 2, pp. 327–334, Apr. 2004.

[10] S. Tao and R. Guerin, "On-line estimation of Internet path performance: an application perspective," in *Proc. of Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, vol. 3, Mar. 2004, pp. 1774–1785.

[11] R. Cáceres, N. Duffield, J. Horowitz, D. Towlsey, and T. Bu, "Multicast-based inference of network-internal characteristics: Accuracy of packet loss estimation," in *Proc. of Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, vol. 1. IEEE, 1999, pp. 371–379.

[12] N. Duffield, F. Lo Presti, V. Paxson, and D. Towsley, "Inferring link loss using striped unicast probes," in *Proc. of Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, vol. 2. IEEE, 2001, pp. 915–923.

[13] M. Yajnikk, S. Moon, J. Kurose, and D. Towsley, "Measurement and modeling of the temporal dependence in packet loss," in *Proc. of Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, New York, Mar. 1999.

[14] V. Paxson, "End-to-end Internet packet dynamics," *IEEE/ACM Transactions on Networking*, vol. 7, no. 3, pp. 277–292, 1999.

[15] J. Bolot, "End-to-end packet delay and loss behavior in the Internet," *ACM SIGCOMM Computer Communication Review*, vol. 23, no. 4, pp. 289–298, 1993.

[16] D. Anick, D. Mitra, and M. Sondhi, "Stochastic theory of a data-handling system with multiple sources," *Bell System Technical Journal*, vol. 61, no. 8, pp. 1871–1894, 1982.

[17] A. Elwalid and D. Mitra, "Effective bandwidth of general markovian traffic sources and admission control of high speed networks," *IEEE/ACM Transactions on Networking*, vol. 1, no. 3, pp. 329–343, 1993.

[18] T. Stern and A. Elwalid, "Analysis of separable markov-modulated rate models for information-handling systems," *Advances in Applied Probability*, pp. 105–139, 1991.

[19] D. Zhang and D. Ionescu, "A new method for measuring packet loss probability using a kalman filter," *IEEE Transactions on Instrumentation and Measurement*, vol. 58, no. 2, Feb. 2009.

[20] ——, "Reactive estimation of packet loss probability for IP-based video services," *IEEE Transactions on Broadcasting*, vol. 55, no. 2, June 2009.

[21] ——, "Online packet loss measurement and estimation for VPN-based services," *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 8, pp. 2154–2166, 2010.

[22] ——, "On packet loss estimation for virtual private networks services," in *Proc. of 13th International Conference on Computer Communications and Networks (IC-CCN)*. IEEE, 2004, pp. 175–180.

[23] M. Allman, S. Dawkins, D. Glover, J. Griner, D. Tran, T. Henderson, J. Heidemann, J. Touch, H. Kruse, S. Ostermann *et al.*, *Ongoing TCP research related to satellites*, RFC 2760, 2000.

[24] T. Henderson and R. Katz, "Transport protocols for Internet-compatible satellite networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 2, pp. 326–344, 1999.

[25] H. Balakrishnan, V. Padmanabhan, and R. Katz, "The effects of asymmetry on TCP performance," *Mobile Networks and Applications*, vol. 4, no. 3, pp. 219–241, 1999.

[26] D. Mills, *Network Time Protocol (NTP)*, ITU-RFC Std., 1992.

[27] How does GPS work. [Online]. Available: www.mikrotik.com/testdocs/ros/2.9/system/gps.pdf

[28] *IEEE Standard for a precision clock synchronization protocol for networked measurement and control systems*, IEC 61588:2009(E), 2009.

[29] L. De Vito, S. Rapuano, and L. Tomaciello, "One-way delay measurement: State of the art," *IEEE Transactions on Instrumentation and Measurement*, vol. 57, no. 12, pp. 2742–2750, 2008.

[30] G. Wei-Xuan and Y. Shun-Zheng, "Inference of one-way queuing delay distribution using packet-pair probes without clock synchronization," in *Proc. of International Conference on Network and Parallel Computing Workshops*. IEEE, 2007, pp. 169–175.

[31] B. Ngamwongwattana and R. Thompson, "Measuring one-way delay of VoIP packets without clock synchronization," in *Proc. Instrumentation and Measurement Technology Conference*. IEEE, 2009, pp. 532–535.

[32] Y.-C. Chang, C.-C. Huang, H.-C. Chang, H.-C. Fang, and L.-G. Chen, "Error-propagation analysis and concealment strategy for MPEG-4 video bitstream with data partitioning," in *IEEE International Conference on Multimedia and Expo (ICME)*, Aug. 2001, pp. 353–356.

[33] N. Färber, K. Stuhlmüller, and B. Girod, "Analysis of error propagation in hybrid video coding with application to error resilience," in *Proc. of Conf. on Image Processing*, vol. 2, Kobe, Japan, 1999, pp. 550–554.

[34] K. Stuhlmüller, N. Färber, M. Link, and B. Girod, "Analysis of video transmission over lossy channels," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 6, pp. 1012–1032, June 2000.

[35] W. Tan, B. Shen, A. Patti, and G. Cheung, "Temporal propagation analysis for small errors in a single-frame in H.264 video," in *Proc. of 15th IEEE International Conference on Image Processing (ICIP)*. IEEE, 2008, pp. 2864–2867.

[36] X. Yang, C. Zhu, Z. Li, X. Lin, and N. Ling, "An unequal packet loss resilience scheme for video over the Internet," *IEEE Transactions on Multimedia*, vol. 7, no. 4, pp. 753–765, 2005.

[37] Y. Liang, J. Apostolopoulos, and B. Girod, "Analysis of packet loss for compressed video: does burst-length matter?" in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 5, Apr. 2003, pp. 684–687.

[38] T. Zinner, O. Hohlfeld, O. Abboud, and T. Hoßfeld, "Impact of frame rate and resolution on objective QoE metrics," in *Proc. of Second International Workshop on Quality of Multimedia Experience (QoMEX)*. IEEE, 2010, pp. 29–34.

[39] Y. Pitrey, M. Barkowsky, P. Le Callet, and R. Pepion, "Subjective quality assessment of MPEG-4 scalable video coding in a mobile scenario," in *2nd European Workshop on Visual Information Processing (EUVIP)*. IEEE, 2010, pp. 86–91.

[40] J. McCarthy, M. Sasse, and D. Miras, "Sharp or smooth?: comparing the effects of quantization vs. frame rate for streamed video," in *Proc. of the conference on Human factors in computing systems (SIGCHI)*. ACM, 2004, pp. 535–542.

[41] A. Khan, L. Sun, and E. Ifeachor, "QoE prediction model and its application in video quality adaptation over UMTS networks," *IEEE Transactions on Multimedia*, no. 99, 2011.

[42] A. Khan, I. Mkwawa, L. Sun, and E. Ifeachor, "QoE-driven sender bitrate adaptation scheme for video applications over IP multimedia subsystem," in *IEEE International Conference on Communications (ICC)*. IEEE, 2011, pp. 1–6.

[43] *Opinion model for video-telephony applications*, ITU-T Recommendation G.1070, 2007.

[44] T. Liu, N. Narvekar, B. Wang, R. Ding, D. Zou, G. Cash, S. Bhagavathy, and J. Bloom, "Real-time video quality monitoring," *European Association for Signal Processing (EURASIP)*, 2011.

[45] A. Vakili and J. C. Grégoire, "Estimation of packet loss probability from traffic parameters for multimedia over IP," in *Seventh International Conference on Networking and Services (ICNS)*, 2011, pp. 44–48.

[46] ——, "Real-time packet loss probability estimates from IP traffic parameters," *International Journal on Advances in Networks and Services (IARIA)*, 2012.

[47] ——, "Accurate one-way delay estimation: limitations and improvements," *IEEE Transactions on Instrumentation and Measurement*, no. 99, pp. 1 –8, 2012.

[48] ——, "Modelling the impact of the position of frame loss on transmitted video quality," *Journal of Convergance (JoC)*, vol. 2, no. 2, pp. 43–48, 2011.

[49] ——, "Impact of frame loss position on transmitted video quality: models and improvements," in *Proc. of the Fifth FTRA International Conference on Multimedia and Ubiquitous Engineering*. IEEE, 2011, pp. 113–118.

[50] ——, "QoE management in a video conferencing application," in *Proc. of the Sixth FTRA International Conference on Multimedia and Ubiquitous Engineering*. Springer, 2012.

[51] ——, "QoE management for video conferencing applications," Aug. 2012, submitted to Computer Netwoks Journal.

[52] *ITU-T SG 12: Definition of Quality-of-Experience*, COM12 - LS 62 -E, TD 109rev2 (PLEN/12), Geneva, Switzerland, Jan. 2007.

[53] M. Fiedler, K. Kilkki, and P. Reichl, "Executive summary – from quality of service to quality of experience," in *From Quality of Service to Quality of Experience*, ser. Dagstuhl Seminar Proceedings, M. Fiedler, K. Kilkki, and P. Reichl, Eds., no. 09192. Dagstuhl, Germany: Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany, 2009.

[54] M. Varela, "Some musings on QoE definitions philosophy and practice," VTT Technical Research Center of Finland, Tech. Rep., 2012.

[55] A. Bouch, M. Sasse, and H. DeMeer, "Of packets and people: a user-centered approach to quality of service," in *Proc. of 8th International Workshop on Quality of Service (IWQOS)*. IEEE, 2000, pp. 189–197.

[56] A. Bouch and M. Sasse, “Why value is everything: A user-centered approach to Internet quality of service and pricing.” Springer, 2001, pp. 59–72.

[57] S. Jumisko-Pyykkö, J. Häkkinen, and G. Nyman, “Experienced quality factors-qualitative evaluation approach to audiovisual quality,” *Multimedia on Mobile Devices 2007*, vol. 6507, pp. 6507–21, 2007.

[58] *Subjective video quality assessment methods for multimedia applications*, ITU-T Recommendation P.910, 2008.

[59] *Methodology for the subjective assessment of the quality of television picture*, ITU-T Recommendation BT.500-11, 2002.

[60] *Methods for subjective determination of transmission quality*, ITU-T Recommendation P.800, 1996.

[61] *Interactive test methods for audiovisual communications*, ITU-T Recommendation P.920, 2000.

[62] *Objective quality measurement of telephone-band (300-3400 Hz) speech codecs*, ITU-T Recommendation P.861, 02/1998.

[63] *In-service, Non-intrusive measurement device-voice service measurements*, ITU-T Recommendation P.561, Feb. 1996.

[64] *Telephone transmission quality, telephone installations, local line networks*, ITU-T P-Series Recommendations. [Online]. Available: http://www.itu.int/rec/T-REC-P/en.

[65] N. Kiatawaki and K. Itoh, “Pure delay effect on speech quality in telecommunications,” *Selected Areas of Communication*, vol. 9, no. 4, pp. 586–593, 1991.

[66] *Subjective evaluation of conversational quality*, ITU-T Recommendation P.805, 04/2007.

[67] Video Quality Experts Group (VQEG), FRTV, RRNR-TV and HDTV working groups. [Online]. Available: www.vqeg.org,www.its.bldrdoc.gov/vqeg/projects/projects.php

[68] D. A. Rosenthal, “Analyses of selected variables effecting video streamed over IP,” *International Journal of Network Management*, vol. 14, no. 3, pp. 193–211, 2004.

[69] *PQA-300 Picture Quality Analysis system 071-0909-00*, Tektronix, Inc., Beaverton, Oregon.

[70] J. Klaue, B. Rathke, and A. Wolisz, "Evalvid - a framework for video transmission and quality evaluation," in *Conference on Modeling Techniques and Tools for Computer Performance Evaluation*, 2003.

[71] *Objective perceptual video quality measurement techniques for digital cable television in the presence of a full reference*, ITU-T Recommendation J.144, 2001.

[72] M. Venkataraman, S. Sengupta, M. Chatterjee, and R. Neogi, "Towards a video QoE definition in converged networks," in *Proc. of Second International conference on Digital Communications (ICDT)*, Silicon Valley, CA, July 2007, p. 16.

[73] J. Boyce and R. Gaglianello, "Packet loss effects on MPEG video sent over the public Internet," in *ACM Multimedia*, Bristol, UK, September 1998.

[74] P. Calyam, M. Sridharan, W. Mandrawa, , and P. Schopis, "Performance measurement and analysis of H.323 traffic," in *Passive and Active Measurement Workshop (PAM), Springer Lecture Notes in Computer Science*, 2004.

[75] *Perceptual evaluation of speech quality (PESQ), an objective method for end-to-end speech quality assessment of narrowband telephone networks and speech codecs*, ITU-T Recommendation P.862, Nov 2007.

[76] *Perceptual objective listening quality assessment (POLQA)*, ITU-T Recommendation P.863, Jan 2011.

[77] *The E-model, a computational model for use in transmission planning*, ITU-T Recommendation G.107, 2005.

[78] A. Takahashit, H. Yoshino, , and N. Kitawaki, "Perceptual QoS assessment technologies for VoIP," *IEEE Communication*, vol. 42, no. 7, pp. 28–35, July 2004.

[79] *Single-ended method for objective speech quality assessment in narrow-band telephony applications*, ITU-T Recommendation P.563, May 2004.

[80] C. Chang Chih and L. C. J. LIBSVM: a library for support vector machines. [Online]. Available: http://www.csie.ntu.edu.tw/~cjlin/libsvm

[81] A. Markopoulou, F. Tobagi, , and M. Karam, "Assessment of VoIP quality over Internet backbones," in *Proc. of Conference of the IEEE Computer and Communications Societies (INFOCOM)*, New York, NY, June 2002, pp. 150–159.

[82] Y. Liang, J. Apostolopoulos, and B. Girod, "Analysis of packet loss for compressed video: effect of burst losses and correlation between error frames," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 7, pp. 861–874, July 2008.

[83] T. L. Lin, P. Cosman, and A. Reibman, "Perceptual impact of bursty versus isolated packet losses in h.264 compressed video," in *IEEE International Conference on Image Processing*, 2008.

[84] J. E. Voldhaug, E. Hellerud, and U. Svensson, "How much is too much? on the acceptability of packet loss distorted audio," in *Proc. of International Conference on Signal Processing and Communications (ICSPC)*, Dubai, UAE, Nov. 2007.

[85] G. Cermak and V. Laboratories, "Subjective video quality as a function of bit rate, frame rate, packet loss and codec," in *International Workshop on Quality of Multimedia Experience*, 2009.

[86] S. Iai, T. Kurita, and N. Kitawaki, "Quality requirements for multimedia communication services and terminals-interaction of speech and video delays," in *Proc. of Conference on Global Telecommunications (GLOBECOM)*, 1993, pp. 394–398.

[87] R. Coleand and J. Rosenbluth, "Voice over IP performance monitoring," *ACM Computer Communication Review*, vol. 31, no. 2, Apr. 2001.

[88] J. G. Beerends, A. P. Hekstra, A. W. Rix, and M. P. Hollier, "Perceptual evaluation of speech quality (PESQ), the new ITU standard for end-to-end speech quality assessment-part II: psychoacoustic model," *Audio Engineering Society*, vol. 50, no. 10, pp. 765–778, 2002.

[89] *One-way transmission time*, ITU-T Recommendation G.114, 2003.

[90] M. Guéguin, R. Le Bouquin-Jeannés, V. Gautier-Turbin, G. Faucon, and V. Barriac, "On the evaluation of the conversational speech quality in telecommunications," *European Association for Signal Processing (EURASIP)*, Jan. 2008.

[91] S. Tallak, R. Kubichek, and J. Shroeder, "Time delay estimation for objective speech quality," in *Communications, Computers and Signal Processing*, Pacific Rim, 1993.

[92] S. Tallak, R. Kubichek, and J. Schroederl, "Time delay estimation for objective quality evaluation of low bit-rate coded speech with noisy channel conditions," in *Signals, Systems and Computers*, vol. 2, Nov. 1993, pp. 1216–1219.

[93] L. Zheng, L. Zhang, and D. Xu, "Characteristics of network delay and delay jitter and its eeffect on Voice over IP (VoIP)," in *Proc. of International Conference on Communications (ICC)*, 2001, pp. 122–126.

[94] M. Claypool and J. Tanner, "The effects of jitter on the perceptual quality of video," in *ACM Multimedia*, Orlando, FL, 1999, pp. 115–118.

[95] B. Oklander and M. Sidi, “Jitter buffer analysis,” in *Proc. of International Conference on Computer Communications and Networks (ICCCN)*, Virgin Islands, Aug. 2008.

[96] H. Hata, “Playout buffering algorithm using of randomwalk in VoIP,” in *Communications and Information Technology*, Tokyo, 2004, pp. 457–460.

[97] M. Liu, J. Jeffrey Rodriguez, and M. Kevin McNeill, “An adaptive jitter buffer playout scheme to improve VoIP quality in wireless networks,” in *Proc. of Conference on Military Communications(MILCOM)*, Washington, D.C., Oct. 2006.

[98] E. Kelly, “Quality of service in Internet protocol (IP) networks,” in *International Communications Industries Association (Infocomm)*, 2002.

[99] R. Serral-Gracia, A. Cabellos-Aparicio, and J. Domingo-Pascual, “Packet loss estimation using distributed adaptive sampling,” in *IEEE Workshop on End-to-End Monitoring Techniques and Services*, Apr. 2008, pp. 124–131.

[100] R. Van de Meent and M. Mandjes, “Evaluation of user-oriented and black-box traffic models for link provisioning,” *Proc. of 1st EuroNGI Conference on Next Generation Internet Networks Traffic*, pp. 380–387, Apr. 2005.

[101] J. Kilpi and I. Norros, “Testing the gaussian approximation of aggregate traffic,” in *2nd ACM SIGCOMM Workshop on Internet measurement*, Marseille, France, Nov. 2002.

[102] Y. Zhang, V. Paxson, S. Shenker, and L. Breslau, “The stationarity of Internet path properties: routing, loss, and throughput,” *ACIRI Techinical Report*, 2000.

[103] Y. Zhang and N. Duffield, “On the constancy of Internet path properties,” in *Proc. of the 1st ACM SIGCOMM Workshop on Internet Measurement*. ACM, 2001, pp. 197–211.

[104] V. Jacobson, “Pathchar: A tool to infer characteristics of Internet paths,” 1997.

[105] M. Jain and C. Dovrolis, “End-to-end available bandwidth: Measurement methodology, dynamics, and relation with TCP throughput,” in *Proc. of ACM SIGCOMM*. ACM, 2002, pp. 295–308.

[106] G. He and J. Hou, “On exploiting long range dependence of network traffic in measuring cross traffic on an end-to-end basis,” in *Proc. of Twenty-Second Annual Joint Conference of the IEEE Computer and Communications (INFOCOM)*, vol. 3. IEEE, 2003, pp. 1858–1868.

[107] K. Salamatian and S. Vaton, "Hidden markov modeling for network communication channels," in *ACM SIGMETRICS Performance Evaluation Review*, vol. 29. ACM, 2001, pp. 92–101.

[108] C. Chang, *Performance guarantees in communication networks*. New York, NY: Springer-Verlag, 2000.

[109] N. Likhanov and R. Mazumdar, "Cell loss asymptotics in buffers fed with a large number of independent stationary sources," in *Proc. of Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, vol. 1. IEEE, 1998, pp. 339–346.

[110] C. Lambiri, "On the estimation and control of packet loss for VPN services," Ph.D. dissertation, University of Ottawa, 2003.

[111] R. Bahadur and R. Rao, "On deviations of the sample mean," *Annals of Mathematical Statistics*, vol. 31, no. 4, pp. 1015–1027, 1960.

[112] H. Kim and N. Shroff, "Loss probability calculations and asymptotic analysis for finite buffer multiplexers," *IEEE/ACM Transactions on Networking*, vol. 9, no. 6, pp. 755–768, 2001.

[113] J. Choe and N. Shroff, "A central-limit-theorem-based approach for analyzing queue behavior in high-speed networks," *IEEE/ACM Transactions on Networking*, vol. 6, no. 5, pp. 659–671, 1998.

[114] K. Debicki and M. Mandjes, "Exact overflow asymptotics for queues with many gaussian inputs," *Journal of applied probability*, vol. 40, no. 3, pp. 704–720, 2003.

[115] F. Kelly, "Notes on effective bandwidths," *in Stochastic Networks: Theory and Applications, Oxford University Press*, pp. 141–168, 1996.

[116] J. Gärtner, "On large deviations from the invariant measure," *Theory of Probability and its Applications*, vol. 22, 1977.

[117] R. Ellis, "Large deviations for a general class of random vectors," *The Annals of Probability*, vol. 12, no. 1, pp. 1–12, 1984.

[118] W. Whitt, "Tail probabilities with statistical multiplexing and effective bandwidths in multi-class queues," *Telecommunication Systems*, vol. 2, no. 1, pp. 71–107, 1993.

[119] P. Glynn and W. Whitt, "Logarithmic asymptotics for steady-state tail probabilities in a single-server queue," *Journal of Applied Probability*, pp. 131–156, 1994.

[120] S. Tao, K. Xu, A. Estepa, T. Gao, R. Guerin, J. Kurose, D. Towsley, and Z. Zhang, "Improving VoIP quality through path switching," in *Proc. of 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, vol. 4, Mar. 2005, pp. 2268–2278.

[121] C. Lambiri, D. Ionescu, and V. Groza, "A new method for the estimation and measurement of traffic packet loss for virtual private networks services," in *Instrumentation and Measurement Technology Conference*, vol. 1, May 2004, pp. 401–406.

[122] D. P. Heyman and T. Lakshman, "What are the implications of long-range dependence for vbr-video traffic engineering," *IEEE/ACM Transactions Networking*, vol. 4, no. 3, pp. 301–317, June 1996.

[123] B. Ryu and A. Elwalid, "The importance of long-range dependence of vbr video traffic in atm traffic engineering: myths and realities," *ACM SIGCOMM Computer Communication Review*, vol. 26, no. 4, pp. 3–14, Oct. 1996.

[124] X. Yu, I. Thng, and Y. Jiang, "Measurement-based effective bandwidth estimation for long range dependent traffic," in *Proc. of IEEE Region 10 International Conference on Electrical and Electronic Technology (TENCON)*, vol. 1. IEEE, 2001, pp. 359–365.

[125] U. Berkeley, LBL, USC/ISI, and X. PARC. Network simulator ns-2. [Online]. Available: http://www.isi.edu/nsnam/ns

[126] M. Weigle, P. Adurthi, F. Hernández-Campos, K. Jeffay, and F. Smith, "Tmix: a tool for generating realistic TCP application workloads in ns-2," *ACM SIGCOMM Computer Communication Review*, vol. 36, no. 3, pp. 65–76, 2006.

[127] P. Greenwood and M. Nikulin, "A guide to chi-squared testing," Wiley-Interscience, Tech. Rep., 1996.

[128] O. Gurewitz and M. Sidi, "Estimating one-way delays from cyclic-path delay measurements," in *Proc. of Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, vol. 2. IEEE, 2001, pp. 1038–1044.

[129] O. Gurewitz, I. Cidon, and M. Sidi, "One-way delay estimation using network-wide measurements," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2710–2724, june 2006.

[130] G. Almes, S. Kalidindi, and M. Zekauskas, *A One-way delay metric for IPPM*, RFC 2679, 1999.

[131] T. Zseby, S. Zander, and G. Carle, "Evaluation of building blocks for passive one-way-delay measurements," in *Proc. of PAM Workshop*, Amsterdam, Netherlands, 2001.

[132] V. Smotlacha, "One-way delay measurement using NTP synchronization," in *Proc. TERENA Networking Conference*, 2003.

[133] J. Wang, M. Zhou, and Y. Li, "Survey on the end-to-end Internet delay measurements," *High Speed Networks and Multimedia Communications*, pp. 155–166, 2004.

[134] H. Weibel, "High precision clock synchronization according to IEEE 1588 implementation and performance issues," *Proc. Embedded World*, pp. 22–24, 2005.

[135] D. Luong and J. Biro, "Partial method versus end-to-end measurement," High Speed Networks Laboratory, Budapest University of Technology and Economics, Tech. Rep. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.32.5805

[136] M. Tsuru, T. Takine, and Y. Oie, "Estimation of clock offset from one-way delay measurement on asymmetric paths," in *Proc. of Conference on Applications and the Internet (SAINT)*. IEEE, 2002, pp. 126–133.

[137] H. Khlifi and J. C. Grégoire, "Estimation and removal of clock skew from delay measures," in *Proc. of 29th Annual Local Computer Networks*. IEEE, 2004, pp. 144–151.

[138] V. Paxson, "On calibrating measurements of packet transit times," in *ACM SIGMETRICS Performance Evaluation Review*, vol. 26, no. 1. ACM, 1998, pp. 11–21.

[139] S. Moon, P. Skelly, and D. Towsley, "Estimation and removal of clock skew from network delay measurements," in *Proc. of Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, vol. 1. IEEE, 1999, pp. 227–234.

[140] L. Zhang, Z. Liu, and C. Honghui Xia, "Clock synchronization algorithms for network measurements," in *Proc. of Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, vol. 1. IEEE, 2002, pp. 160–169.

[141] Y. Lin, H. Wang, G. Kuo, S. Cheng, and Q. Li, "A fuzzy-based approach to remove clock skew and reset from one-way delay measurement," *IEEE Transactions on Neural Networks*, vol. 16, no. 5, pp. 1125–1135, 2005.

[142] O. Hodson, C. Perkins, and V. Hardman, "Skew detection and compensation for Internet audio applications," in *Proc. of International Conference on Multimedia and Expo (ICME)*, vol. 3. IEEE, 2000, pp. 1687–1690.

[143] Q. Chaudhari, E. Serpedin, and K. Qaraqe, "On maximum likelihood estimation of clock offset and skew in networks with exponential delays," *IEEE Transactions on Signal Processing*, vol. 56, no. 4, pp. 1685–1697, 2008.

[144] T. Kohno, A. Broido, and K. Claffy, "Remote physical device fingerprinting," *IEEE Transactions on Dependable and Secure Computing*, vol. 2, no. 2, pp. 93–108, 2005.

[145] J. Choi and C. Yoo, "Analytic end-to-end estimation for the one-way delay and its variation," in *Second IEEE Consumer Communications and Networking Conference (CCNC)*. IEEE, 2005, pp. 527–532.

[146] A. Hernandez and E. Magafia, "One-way delay measurement and characterization," in *Proc. of Third International Conference on Networking and Services (ICNS)*. IEEE, 2007.

[147] *Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s*, ITU-T Recommendation G.723.1, 03/1996.

[148] R. L. Carter and Crovella, "Measuring bottleneck link in packet-switched networks," Boston University, Tech. Rep. BU-CS-96-006, 1996.

[149] V. Paxon, "Measurements and analysis of end-to-end Internet dynamics," Ph.D. dissertation, University of California at Berkeley, 1997.

[150] P. Beyssac. Bing: A point-to-point bandwidth measurement tool based on ping. [Online]. Available: http://fgouget.free.fr/bing/bing_src-readme-1st.shtml

[151] W. Jiang, "Detecting and measuring asymmetric links in an IP network," Columbia University, Tech. Rep. CUCS009-99, 1999.

[152] *Advanced video coding for generic audiovisual services*, ITU-T Recommendation H.264/ISO/IEC 11496-10, Mar. 2010.

[153] Y. Liang, J. Apostolopoulos, and B. Girod, "Analysis of packet loss for compressed video: effect of burst losses and correlation between error frames," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 7, pp. 861–874, July 2008.

[154] W. Tan and A. Zakhor, "Video multicast using layered FEC and scalable compression," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 3, pp. 373–386, 2001.

[155] Y. Liang, E. Setton, and B. Girod, "Network-adaptive video communication using packet path diversity and rate-distortion optimized reference picture selection," *Journal of VLSI Signal Processing*, vol. 41, no. 3, pp. 345–354, 2005.

[156] J. Apostolopoulos, W. Tan, S. Wee, and G. Wornell, "Modeling path diversity for multiple description video communication," in *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 3. IEEE, 2002.

[157] R. Zhang, S. Regunathan, and K. Rose, "Video coding with optimal inter/intra-mode switching for packet loss resilience," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 6, pp. 966–976, 2000.

[158] T. Maugey, T. André, B. Pesquet-Popescu, and J. Farah, "Analysis of error propagation due to frame losses in a distributed video coding system," in *Proc. of European signal processing conference*. Citeseer, 2008.

[159] G. Sun, W. Xing, and D. Lu, "An novel loss protection scheme for H.264 video stream based on frame error propagation index," in *Proc. of Third International Conference on Communications and Networking*. China: IEEE, 2008.

[160] H. Lee, T. Chiang, and Y. Zhang, "Scalable rate control for MPEG-4 video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, no. 6, pp. 878–894, 2000.

[161] Z. Li, J. Chakareski, X. Niu, and Y. Zhang, "Modeling and analysis of distortion caused by markov-model burst packet losses in video transmission," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 7, pp. 917–931, July 2009.

[162] ——, "Modeling of distortion caused by markov-model burst packet losses in video transmission," in *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, Oct. 2009.

[163] C.-H. Ke, C.-K. Shieh, W.-S. Hwang, and A. Ziviani, "An evaluation framework for more realistic simulations of MPEG video transmission," *Information Science and Engineering*, vol. 24, pp. 425–440, 2008.

[164] Y. Liao and J. Gibson, "Enhanced error resilience of video communications for burst losses using an extended rope algorithm," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2009, pp. 1853–1856.

[165] ——, “Rate-distortion based mode selection for video coding over wireless networkswith burst losses,” in *17th International Packet Video Workshop*, May 2009, pp. 1–12.

[166] C.-Y. Hsu, A. Ortega, and M. Khansari, “Rate control for robust video transmission over burst-error wireless channels,” *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 5, pp. 756–773, May 1999.

[167] V. Varsa, M. Karczewicz, G. Roth, R. Sjöberg, G. Liebl, and T. Stockhammer, *Common test conditions for RTP/IP over 3GPP/3GPP2*, ITU-T SG.16/VCEG-L77, 2001.

[168] H. Schwarz, D. Marpe, and T. Wiegand, “Overview of the scalable video coding extension of the H.264/AVC standard,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1103–1120, 2007.

[169] H. Knoche and M. Sasse, “The big picture on small screens delivering acceptable video quality in mobile TV,” *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)*, vol. 5, no. 3, p. 20, 2009.

[170] R. Feghali, F. Speranza, D. Wang, and A. Vincent, “Video quality metric for bit rate control via joint adjustment of quantization and frame rate,” *IEEE Transactions on Broadcasting*, vol. 53, no. 1, pp. 441–446, 2007.

[171] D. Wang, F. Speranza, A. Vincent, T. Martin, and P. Blanchfield, “Toward optimal rate control: a study of the impact of spatial resolution, frame rate, and quantization on subjective video quality and bit rate,” in *Proc. of Visual Communications and Image Processing*, vol. 5150, 2003, p. 198.

[172] K. Reddy and L. Reddy, “A survey on congestion control mechanisms in high speed networks,” *International Journal of Computer Science and Network Security (IJCSNS)*, vol. 8, no. 1, pp. 187–195, 2008.

[173] M. Allman, V. Paxson, and W. Stevens, *TCP congestion control*, RFC 2581, Apr. 1999. [Online]. Available: http://www.ietf.org/rfc/rfc2581

[174] M. Handley, S. Floyd, J. Padhey, and J. Widmer, *TCP Friendly Rate Control (TFRC): protocol specification*, IETF RFC 3448, Jan. 2003.

[175] I. Rhee, V. Ozdemir, and Y. Yi, “Tear: TCP emulation at receivers - flow control for multimedia streaming,” NCSU, Tech. Rep., Apr. 2000.

[176] E. Kohler, M. Handley, and S. Floyd, *Datagram Congestion Control Protocol (DCCP)*, IETF RFC 4340, Mar. 2006.

[177] R. Rejaie, M. Handley, , and D. Estrin, "Rap: An end-to-end rate-based congestion control mechanism for realtime streams in the Internet," in *Proc. of Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*. IEEE, Mar. 1998.

[178] S. H. Choi and M. Handley, "Designing TCP-friendly window-based congestion control for real-time multimedia applications," in *PFLDNet 2009*, Feb. 2009.

[179] G. Yang, T. Sun, M. Gerla, M. Sanadidi, and L. Chen, "Smooth and efficient real-time video transport in the presence of wireless errors," *ACM transactions on multimedia computing, communications, and applications (TOMCCAP)*, vol. 2, no. 2, pp. 109–126, 2006.

[180] S. Cen, P. Cosman, and G. Voelker, "End-to-end differentiation of congestion and wireless losses," *IEEE/ACM Transactions on Networking*, vol. 11, no. 5, pp. 703–717, 2003.

[181] T. Kim and M. Ammar, "Optimal quality adaptation for scalable encoded video," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 2, pp. 344–356, 2005.

[182] H. Schulzrinne, S. Casner, R. Fredrick, and V. Jacobson, *RTP: A transport protocol for real-time applications Internet engineering task force*, RFC 3550, 2003.

[183] *End-user multimedia QoS categories*, ITU-T Recommendation G.1010, 2001.

[184] J. Martin, A. Nilsson, and I. Rhee, "Delay-based congestion avoidance for TCP," *IEEE/ACM Transactions on Networking*, vol. 11, no. 3, pp. 356–369, 2003.

[185] R. L. Carter and Crovella, "Measuring bottleneck link in packet-switched networks," Boston University, Tech. Rep. BU-CS-96-006, 1996.

[186] V. Paxon, "Measurements and analysis of end-to-end Internet dynamics," Ph.D. dissertation, University of California at Berkeley, 1997.

[187] W. Jiang, "Detecting and measuring asymmetric links in an IP network," Columbia University, New York, Tech. Rep. CUCS009-99, Jan. 1999.

[188] D. Nguyen and J. Ostermann, "Congestion control for scalable video streaming using the scalability extension of H.264/avc," *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 2, pp. 246–253, 2007.

[189] M. Pinson and S. Wolf, "Comparing subjective video quality testing methodologies," in *SPIE Video Communications and Image Processing Conference*, 2003, pp. 573–582.

[190] Y. Ou, Z. Ma, and Y. Wang, "Modeling the impact of frame rate and quantization stepsizes and their temporal variations on perceptual video quality: A review of recent works," in *Proc. of 44th Annual Conference on Information Sciences and Systems (CISS)*. IEEE, 2010.

[191] Y. Ou, Y. Zhou, and Y. Wang, "Perceptual quality of video with frame rate variation: a subjective study," in *Proc. of IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*. IEEE, 2010, pp. 2446–2449.

[192] Z. Ma, M. Xu, Y. Ou, and Y. Wang, "Modeling of rate and perceptual quality of compressed video as functions of frame rate and quantization stepsize and its applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 5, pp. 671–682, 2012.

[193] P. Reichl, B. Tuffin, and R. Schatz, "Logarithmic laws in service quality perception: Where microeconomics meets psychophysics and quality of experience," *Telecommunication Systems*, pp. 1–14, 2011.

[194] S. Egger, P. Reichl, T. Hoßfeld, and R. Schatz, "Time is bandwidth? narrowing the gap between subjective time perception and quality of experience," in *Proc. IEEE International Conference on Communications (ICC 2012)-Communication QoS, Reliability and Modeling Symposium, Ottawa, Canada (June 2012)*, 2012.

[195] T. Hobfeld, R. Schatz, M. Varela, and C. Timmerer, "Challenges of qoe management for cloud applications," *Communications Magazine, IEEE*, vol. 50, no. 4, pp. 28–36, 2012.

[196] [Online]. Available: http://gstreamer.freedesktop.org/documentation/

[197] "MCN streaming: An adaptive video streaming platform," Dept. of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611, USA, Tech. Rep. [Online]. Available: http://www.wu.ece.ufl.edu/projects/wirelessVideo/project/mcn_streaming