

**User-Perceived Quality-Aware Adaptation  
of Streamed Multimedia over  
Best-effort IP Networks**

Volume 1 of 2

Nicola Cranley

National University of Ireland

Department of Computer Science  
Faculty of Science  
University College Dublin  
Belfield, Dublin 4

A thesis submitted for the Degree of Doctor of Philosophy

March 2004

Head of Department: G.M.P. O'Hare  
Supervisor: Liam Murphy

## Abstract

*There is an increasing demand for streaming video applications over both the fixed Internet and wireless IP networks. MPEG-4 and H.264 are compression standards targeted at streamed multimedia services over wireless best-effort IP. The dynamic nature of best-effort networks in terms of fluctuating bandwidth and time-varying delays makes it challenging for the application to provide good quality streaming under such constraints.*

*Many adaptive delivery mechanisms have been proposed over recent years; however, most do not explicitly consider user-perceived quality when making adaptations, nor do they define what quality is. This research proposes that an optimal adaptation trajectory through the set of possible encodings exists, and that it indicates how to adapt transmission in response to changes in network conditions to maximize user-perceived quality. Such an optimum adaptation trajectory can be used with any transmission adaptation policy. There have been many studies by various research groups to devise an objective metric for user perceived quality. However, none of these have a good correlation to human visual system. This research describes a subjective methodology that can be used to discover these optimum adaptation trajectories for a number of different MPEG-4 video clips. Using this knowledge of user perceived quality, an end-to-end adaptation algorithm and system architecture have been developed for adaptive streaming to wireless devices which adapts the quality of the stream in such a way so as to provide the maximum user perceived quality.*

# Table of Contents

## Chapter 1

<b>Introduction .....</b>	<b>1</b>
1.1. MOTIVATION .....	1
1.2. PROBLEM STATEMENT .....	3
1.3. SOLUTION .....	3
1.4. CONTRIBUTIONS .....	4
1.5. THESIS OUTLINE.....	5

## Chapter 2

<b>Multimedia Streaming .....</b>	<b>6</b>
2.1. MULTIMEDIA ENCODING .....	6
2.1.1. MPEG-4 .....	8
2.2. TRANSPORT PROTOCOLS .....	9
2.2.1. REAL-TIME TRANSPORT PROTOCOL (RTP) AND REAL-TIME TRANSPORT CONTROL PROTOCOL (RTCP).....	9
2.2.2. REAL-TIME STREAMING PROTOCOL (RTSP) .....	10
2.2.3. SESSION DESCRIPTION PROTOCOL (SDP) .....	10
2.3. STREAMING .....	10
2.3.1. CHALLENGES FOR STREAMING MULTIMEDIA .....	12
2.3.2. COMMERCIAL STREAMING SOLUTIONS.....	16
2.3.3. FUTURE TRENDS IN STREAMING .....	17

## CHAPTER 3

<b>Literature Review.....</b>	<b>18</b>
3.1. INTRODUCTION.....	18
3.2. MULTIMEDIA ADAPTATION TECHNIQUES .....	18
3.2.1. ADAPTATION TECHNIQUES.....	19
3.2.2. RATE CONTROL .....	21
3.2.2.1. SERVER-BASED RATE-BASED CONTROL .....	21
3.2.2.2. RECEIVER-BASED RATE CONTROL.....	28
3.2.2.3. HYBRID RATE CONTROL .....	31
3.2.3. RATE SHAPING .....	33
3.2.4. RATE ADAPTIVE ENCODING.....	35
3.2.5. TRANSCODER-BASED ADAPTATION.....	39

3.2.6. ERROR CONTROL.....	40
3.2.7. CRITIQUE OF ADAPTATION TECHNIQUES .....	44
3.3. SUBJECTIVE TEST METHODOLOGIES .....	49
3.3.1. ABSOLUTE CATEGORY RATING METHOD (ACR).....	49
3.3.2. DEGRADED CATEGORY RATING METHOD (DCR).....	49
3.3.3. PAIR COMPARISON METHOD (PC).....	50
3.3.4. FORCED CHOICE METHODOLOGY .....	51
3.3.5. MULTIPLE STIMULUS HIDDEN REFERENCE AND ANCHORS (MUSHRA)..	52
3.3.6. GRADING SCALES .....	52
3.4. OBJECTIVE METRICS .....	54
3.4.1. VQEG.....	55
3.4.2. REVIEW OF OBJECTIVE METRICS .....	56

## **CHAPTER 4**

<b>Optimum Adaptation Trajectories .....</b>	<b>60</b>
4.1. INTRODUCTION.....	60
4.1.1. ADAPTIVE STREAMING.....	61
4.1.2. OPTIMUM ADAPTATION TRAJECTORIES .....	62
4.1.3. HUMAN PERCEPTION .....	63
4.1.4. LIMITATIONS OF OBJECTIVE METRICS .....	64
4. 2. SUBJECTIVE TESTING CONSIDERATIONS.....	65
4.2.1. CHOICE OF TEST MATERIAL.....	65
4.2.2. SPATIAL AND TEMPORAL CONSIDERATIONS.....	65
4.2.3. SUBJECTIVE TEST METHODOLOGIES.....	67
4.2.4. KNOWN CRITICISMS OF CONVENTIONAL SUBJECTIVE TESTING METHODS .....	68
4.2.5. SUBJECT CONSIDERATIONS .....	68
4.2.6. RELIABILITY AND REPLICATION .....	69
4.2.7. POPULATION SAMPLE.....	69
4.3. TEST SEQUENCE SELECTION .....	70
4.3.1. TEST SEQUENCE PREPARATION.....	72
4.4. SAMPLING THE ADAPTATION SPACE .....	73
4.4.1. WEBER'S LAW OF JUST NOTICEABLE DIFFERENCE (JND).....	73
4.4.2. FRAME RATE PARAMETERS.....	74
4.4.3. SPATIAL RESOLUTION PARAMETERS .....	75
4.4.4. ADAPTATION SPACE.....	76
4.4.5. DETERMINING TEST CASES .....	76

4.5. TEST PROCEDURE.....	78
4.5.1. PARALLEL TESTING.....	78
4.5.2. TEST ENVIRONMENT .....	78
4.5.3. VIEWING DEVICE.....	78
4.5.4. VIEWING DISTANCE.....	79
4.5.5. SUBJECT SCREENING .....	79
4.5.6. TEST METHODOLOGY .....	80
4.6. RESULTS AND STATISTICAL ANALYSIS .....	81
4.6.1. INTER-TESTER RELIABILITY.....	81
4.6.2. STATISTICAL ANALYSIS .....	82
4.6.3. DATA RESULTS .....	90
4.6.4. DISCUSSION .....	95
4.7. INTERPOLATION TESTS .....	96
4.7.1. INTERPOLATION TEST RESULTS .....	96
4.7.2. DISCUSSION .....	99
4.8. ANALYSIS AND DISCUSSION OF RESULTS .....	100
4.8.1. COMPARISON OF PATHS OF MAXIMUM USER PREFERENCE.....	100
4.8.2. COMPARISON OF WEIGHTED PATHS OF PREFERENCE.....	102
4.8.3. GLOBALLY AVERAGED OAT .....	104
4.9. SUMMARY AND CONCLUSIONS.....	108
<b>CHAPTER 5</b>	
<b>Validation of the OAT.....</b>	<b>110</b>
5.1. USING OBJECTIVE METRICS FOR OAT DISCOVERY.....	110
5.1.1. OAT DISCOVERY USING PSNR.....	110
5.1.2. OAT DISCOVERY USING VQM METRICS.....	112
5.1.3. METHODOLOGY FOR OAT DISCOVERY USING VQM METRICS.....	114
5.1.4. PSNR MODEL (VQM-PSNR).....	114
5.1.5. RESULTS .....	115
5.1.6. DISCUSSION .....	116
5.2. ONE-DIMENSIONAL VERSUS TWO-DIMENSIONAL ADAPTATION.....	119
5.2.1. TEST METHODOLOGY .....	119
5.2.2. TEST SEQUENCE PREPARATION.....	120
5.2.3. SSCQE DATA ANALYSIS .....	122
5.2.4. TEST RESULTS .....	123
5.2.5. DISCUSSION .....	130
5.3. CHAPTER CONCLUSIONS AND SUMMARY .....	132

## CHAPTER 6

<b>Adaptive Streaming</b> .....	<b>133</b>
6.1. ADAPTIVE STREAMING OVERVIEW .....	133
6.1.1. SYSTEM OVERVIEW .....	133
6.1.2. HETEROGENEOUS AND SERVER-SPECIFIC CLIENTS .....	135
6.2. SYSTEM ADAPTATION .....	137
6.2.1. SYSTEM ADAPTATION FOR PRE-ENCODED MULTI-TRACKED CONTENT ...	137
6.2.2. MULTI-TRACKED MPEG-4 AND 3GP CONTENT .....	137
6.2.3. TEMPORAL ADAPTATION: FRAME DROPPING .....	140
6.2.4. SPATIAL ADAPTATION: TRACK SWITCHING.....	141
6. 3. USING OATS WITH ANY SENDER-BASED ADAPTATION ALGORITHM .....	143
6.3.1. LDA USING OATS .....	143
6.3.2. LDA SYSTEM ARCHITECTURE .....	145
6.3.3. LDA SIMULATIONS .....	146
6.3.4. DISCUSSION .....	150
6.4. PERCEPTUAL QUALITY ADAPTATION (PQA) .....	151
6.4.1. PQA SYSTEM OVERVIEW .....	151
6.4.2. RTCP-APP: EXPLICIT QUALITY FEEDBACK.....	153
6.4.3. RTCP-RR: INFERRING THE PLAYABLE FRAME RATE.....	154
6.4.4. PQA ALGORITHM.....	158
6.4.5. PQA EXAMPLE OPERATION .....	159
6.4.6. PQA SIMULATION .....	160
6.4.7. DISCUSSION .....	164
6.5. PQA IN WIRELESS NETWORKS.....	166
6.5.1. ISSUES IN WIRELESS STREAMING.....	166
6.5.2. WIRELESS NETWORK TEST SCENARIOS .....	167
6.6. SIMULATION ENVIRONMENTS .....	173
6.6.1. SIMULATION SETUP.....	173
6.6.2. CONTENT PREPARATION FOR SIMULATION TESTS .....	173
6.7. PQA IN WIRELESS NETWORK TEST RESULTS.....	176
6.7.1. CONGESTION BUILD-UP SIMULATION RESULTS.....	176
6.7.2. CONGESTION DECAY SIMULATION RESULTS .....	177
6.7.3. WORST-CASE SCENARIO SIMULATION RESULTS.....	178
6.7.4. MOBILE PATH SIMULATION RESULTS .....	179
6.8. SUMMARY AND CONCLUSIONS.....	181

<b>Chapter 7</b>	
<b>Conclusions and Future Work .....</b>	<b>183</b>
7.1. FUTURE WORK .....	185
<b>Refereed Publications.....</b>	<b>188</b>
<b>References .....</b>	<b>189</b>

## List of Figures

FIGURE 2.1: EVOLUTION OF ENCODING STANDARDS .....	6
FIGURE 2.2: DELAY SOURCES IN A TRANSMISSION SYSTEM .....	14
FIGURE 2.3: RECEIVER JITTER BUFFERING .....	15
FIGURE 3.1: ADAPTATION TECHNIQUES .....	20
FIGURE 3.2: ERROR CONTROL TECHNIQUES .....	20
FIGURE 3.3: AIMD AND MIMD .....	22
FIGURE 3.4: LAYERED MULTICAST .....	29
FIGURE 3.5: 2-LEVEL SNR SCALABLE ENCODER .....	36
FIGURE 3.6: 2-LEVEL SPATIAL SCALABLE ENCODER .....	36
FIGURE 3.7: FGS-FGST IMPLEMENTATION .....	37
FIGURE 3.8: FGST-FGS IMPLEMENTATION .....	38
FIGURE 3.9: MULTIPLE TRANSCODER SYSTEM .....	40
FIGURE 3.10: CHANNEL CODING .....	42
FIGURE 3.11: ACR METHODOLOGY .....	49
FIGURE 3.12: DCR METHODOLOGY .....	50
FIGURE 3.13: PAIR COMPARISON METHODOLOGY .....	51
FIGURE 3.14: FORCED CHOICE METHODOLOGY .....	51
FIGURE 3.15: OBJECTIVE METRIC MEASUREMENT METHODOLOGIES .....	55
FIGURE 4.1: PROBLEM STATEMENT .....	63
FIGURE 4.2: MAPPING CONTENT SPACE TO ADAPTATION SPACE .....	63
FIGURE 4.3: FORCED CHOICE METHODOLOGY .....	67
FIGURE 4.4: SCREEN SHOT OF TEST SEQUENCES .....	70
FIGURE 4.5: TEST SEQUENCE SI-TI VALUES .....	71
FIGURE 4.6: SAMPLING ADAPTATION SPACE .....	76
FIGURE 4.7: INTER-TESTER RELIABILITY FOR C2 .....	81
FIGURE 4.8: INTER-TESTER RELIABILITY FOR C3 .....	82
FIGURE 4.9: NORMAL DISTRIBUTION ABOUT THE MEAN .....	84
FIGURE 4.10: CONFIDENCE LEVELS AND Z-DISTRIBUTION .....	85
FIGURE 4.11: C1 RESULTS .....	91
FIGURE 4.12: C2 RESULTS .....	91
FIGURE 4.13: C3 RESULTS .....	92



FIGURE 4.14: C4 RESULTS .....	92
FIGURE 4.15: C7 RESULTS .....	93
FIGURE 4.16: C9 RESULTS .....	93
FIGURE 4.17: C12 RESULTS.....	94
FIGURE 4.18: INTERPOLATION TESTS FOR C1 .....	97
FIGURE 4.19: PATHS OF MAXIMUM PREFERENCE FOR ALL CONTENT TYPES .....	100
FIGURE 4.20: WEIGHTED PATH OF PREFERENCE FOR ALL CONTENT TYPES .....	103
FIGURE 4.21: GLOBALLY AVERAGED PATHS OF MAXIMUM USER PREFERENCE AND WEIGHTED PATH OF PREFERENCE .....	106
FIGURE 5.1: PLANE OF PSNR .....	112
FIGURE 5.2: PATH OF MAXIMUM PSNR.....	112
FIGURE 5.3: VQM PROCESSING MODEL .....	113
FIGURE 5.4: PLANE OF VQM PSNR.....	115
FIGURE 5.5: PATH OF MAXIMUM VQM PSNR .....	115
FIGURE 5.6: PATHS OF MAXIMUM VQM QUALITY.....	116
FIGURE 5.7: DIFFICULTIES IN OBJECTIVE TEMPORAL ANALYSIS .....	117
FIGURE 5.8: COMPARISON OF PATH OF MAXIMUM USER PREFERENCE AND MAXIMUM VQM PATHS.....	118
FIGURE 5.9: ONE-DIMENSIONAL VERSUS TWO-DIMENSIONAL ADAPTATION .....	119
FIGURE 5.10: SCREEN-SHOT OF SSCQE TESTING APPLICATION .....	120
FIGURE 5.11: BIT RATES OF ONE-DIMENSIONAL VERSUS TWO-DIMENSIONAL ADAPTATION.....	121
FIGURE 5.12: ADAPTIVE TEST SEQUENCE PREPARATION .....	122
FIGURE 5.13: SSCQE DATA ANALYSIS .....	123
FIGURE 5.14: ADAPTING DOWN RESULTS.....	125
FIGURE 5.15: ADAPTING UP RESULTS.....	128
FIGURE 5.16: AIMD ADAPTATION RESULTS .....	130
FIGURE 6.1: BASIC CLIENT-SERVER ARCHITECTURE .....	136
FIGURE 6.2: MULTI-TRACKED .3GP FILE.....	139
FIGURE 6.3: VOP INTER-DEPENDENCIES.....	141
FIGURE 6.4: TRACK SWITCHING.....	142
FIGURE 6.5: LDA_TESTSEQ BIT RATE PLANE .....	144
FIGURE 6.6: LDA_TESTSEQ OAT AND EABR.....	145
FIGURE 6.7: LDA SYSTEM ARCHITECTURE .....	146
FIGURE 6.8: LDA EXAMPLE 1 .....	148

FIGURE 6.9: LDA EXAMPLE 2 .....	149
FIGURE 6.10: PQA SYSTEM ARCHITECTURE .....	153
FIGURE 6.11: RTCP-APP PACKET STRUCTURE.....	154
FIGURE 6.12: PLAYABLE FRAME RATE WITH LOSS.....	156
FIGURE 6.13: PQA ALGORITHM.....	158
FIGURE 6.14: EXAMPLE USAGE OF PQA.....	160
FIGURE 6.15: PQA EXAMPLE 1 .....	162
FIGURE 6.16: PQA EXAMPLE 2 .....	163
FIGURE 6.17: INVERSE RELATIONSHIP BETWEEN BIT RATE AND DELAY .....	167
FIGURE 6.18: WIRELESS CELL.....	169
FIGURE 6.19: WORST-CASE NETWORK CONDITIONS.....	170
FIGURE 6.20: NETWORK CONDITIONS WITH GRADUAL CONGESTION BUILD-UP .....	171
FIGURE 6.21: NETWORK CONDITIONS WITH GRADUAL CONGESTION DECAY .....	172
FIGURE 6.22: NETWORK CONDITIONS OF A SINGLE MOBILE USER .....	172
FIGURE 6.23: TEST CONTENT TRACK STRUCTURE .....	173
FIGURE 6.24: BIT RATE FOR TEST SEQUENCE.....	175
FIGURE 6.25: BIT RATE FLUCTUATIONS OF NON-ADAPTIVE TEST SEQUENCE.....	175
FIGURE 6.26: CONGESTION BUILD-UP SIMULATION RESULTS.....	177
FIGURE 6.27: CONGESTION DECAY SIMULATION RESULTS .....	178
FIGURE 6.28: WORST-CASE SCENARIO SIMULATION RESULTS .....	179
FIGURE 6.29: MOBILE USER SIMULATION RESULTS .....	180
FIGURE 7.1: OATS FOR MULTICAST SESSIONS.....	187

## List of Tables

TABLE 3.1: FIVE POINT QUALITY GRADING SCALE.....	52
TABLE 3.2: FIVE POINT IMPAIRMENT SCALE .....	53
TABLE 4.1: GENERAL TEST SEQUENCE CLASSIFICATION .....	71
TABLE 4.2: LOGARITHMIC RESOLUTION SCALE .....	75
TABLE 4.3: LINEAR RESOLUTION SCALE .....	76
TABLE 4.4: CONFIDENCE INTERVALS FOR POPULATION MEAN ESTIMATION.....	84
TABLE 4.5: ESTIMATED POPULATION SIZE FOR C2.....	86
TABLE 4.6: ESTIMATED POPULATION SIZE FOR C3.....	87
TABLE 4.7: CRITICAL VALUES FOR CHI-SQUARE .....	89
TABLE 4.8: CHI-SQUARE FOR C2 .....	89
TABLE 4.9: CHI-SQUARE FOR C3 .....	90
TABLE 4.10: INTERPOLATION RESULTS FOR CONTENT 1: TEST 1 AND TEST 2.....	98
TABLE 4.11: INTERPOLATION RESULTS FOR CONTENT 1: TEST 3 AND TEST 4.....	99
TABLE 4.12: GLOBALLY AVERAGED OAT RESULTS .....	106
TABLE 5.1: ADAPT DOWN RESULTS.....	124
TABLE 5.2: ADAPT UP RESULTS.....	126
TABLE 5.3: AIMD RESULTS.....	129
TABLE 6.1: DARWIN STREAMING SERVER VERSUS REALPLAYER.....	134
TABLE 6.2: WIRELESS CHANNEL CHARACTERISTICS AT DIFFERENT LOCATIONS.....	169
TABLE 6.3: ENCODING CONFIGURATION FOR TEST SEQUENCE .....	174

# Chapter 1

## Introduction

With the rapid increase of bandwidth and computing power, streaming video over the Internet has been experiencing dramatic growth and has made the transition to mainstream media communications. Cellular systems have evolved to carry IP traffic over the radio links. With recent developments in multimedia compression and wireless network infrastructures, media streaming will soon become a promising revenue generating application. Although there will be more bandwidth available for 3G wireless networks and the latest compression techniques enable very low bit rate streaming, there are still challenges with wireless multimedia streaming. Adaptive streaming has been a much-researched topic, however its focus has been on network level adaptation without any consideration for user perception. The research presented in this thesis gains a better understanding of user perception that can be used to enhance adaptive streaming over wireless networks.

### 1.1. Motivation

Best-effort IP networks are unreliable and unpredictable, particularly in a wireless environment. Losses and excessive delays can be caused by congestion or poor radio channel conditions. There are strict delay constraints imposed by streamed multimedia traffic: if a video packet does not arrive before its playout time, the packet is effectively lost. Packet losses have a particularly devastating effect on the smooth continuous playout of a compressed video sequence due to inter-frame dependencies.

By reducing the bit rate of compressed video, the loss rate can be reduced at the expense of reduced video quality. A slightly lower quality but uncorrupted video stream is less irritating to the user than a corrupted stream. However, rapidly fluctuating quality should also be avoided as the human vision system adapts to a specific quality after a few seconds and it becomes annoying if the viewer has to adjust to a varying quality over short time scales. Therefore, controlled video quality adaptation is needed to reduce the negative effects of congestion whilst providing the highest possible level of stream quality.

Although recent efforts have made some progress in streaming media delivery, today's solutions are proprietary and inflexible and do not consider the end-users perception in the

design and development of streaming technology. In general current streaming video applications:

- Deliver low quality pictures;
- Require large amounts of buffering;
- Do not allow for high user interactivity;
- Do not respond well to the changing conditions of the Internet;
- Do not cooperate with other applications with which they are sharing bandwidth.

One possible approach to the problem of varying network characteristics is to use feedback mechanisms to adapt the output bit rate of the encoders, which in turn adapts the video quality, based on implicit or explicit information received about the state of the network. Several bit rate control mechanisms based on feedback have been proposed in the last few years. For example, Real Time Control Protocol (RTCP) provides network-level Quality of Service (QoS) monitoring and congestion control information such as packet loss, round trip delay, and jitter. Many applications use RTCP to provide control mechanisms for transmission of video over IP networks. However, the network-level QoS parameters provided by RTCP are not video content-based and therefore it is difficult to gauge the quality of the received video stream from this feedback. For example, assume that two video packets are lost during the transmission: one contains important control information, whereas the other contains video data. Obviously, the impact of losing the first packet on the reconstructed video quality is much worse than that of the second packet, but RTCP cannot distinguish the difference since it does not indicate the content of the packets.

In the past few years, there has been much work on *video quality adaptation* and *video quality evaluation*. In general, video quality adaptation indicates how the bit rate of the video should be adjusted in response to changing network conditions. However, this is not addressed in terms of video quality, as for a given bit rate budget there are many ways in which the video quality can be adapted. Video quality evaluation measures the quality of video as perceived by the users, but is not designed for adaptive video transmissions.

The concept of adaptive multimedia streaming is based on the widely accepted maxim that users prefer a reduced bit rate to packet losses. These adaptation policies address the problem of how to adapt only in terms of adjusting the transmission rate, window size or encoding parameters. They do not consider the impact of perception, nor do they indicate how this bit rate adaptation can be achieved in terms of actual parameters used by the encoder.

- Sender-based schemes indicate how to adjust the transmission rate of the sender in response to various network conditions of loss and delay, but, for a given bit rate, there are several ways to encode the content.
- Receiver-based schemes and layered schemes use a policy of join/leave experiments for additional layers of video to improve the perceived quality and make the most of the available bandwidth. The base and enhancement layers can be composed in a large number of different ways.
- Utility-based adaptation schemes adapt video quality using a utility function, where the available network resources and the utility are the basis for adapting video quality. However, the utility is often measured using quality metrics that have not been shown to correlate with human perception.

To assess the user-perceived quality of a sequence, the human vision system (HVS) has to be taken into account. Clearly it would be difficult to produce an adaptive system that accounts for the many facets of the HVS, so some alternative strategy is required if user-perceived quality is to be accommodated.

### 1.2. Problem Statement

The problem this thesis addresses is:

*How to adapt video quality in terms of video encoding parameters and user-perceived quality for streamed video over best-effort IP networks.*

The goal of this work is to gain a better understanding of user perception and integrate this knowledge into video quality adaptation techniques. A key issue addressed in this research is how to adapt in order to maximize the resulting user-perceived quality. This raises the question of how user-perceived quality can be assessed in practice.

### 1.3. Solution

This research proposes that there is an optimal way in which multimedia transmissions should be adapted in response to changing network conditions in order to maximize the user-perceived quality. This is based on the hypothesis that within the set of different ways to achieve a target bit rate, there exists an encoding configuration that maximizes the user-perceived quality.

If a particular multimedia file has  $n$  independent encoding parameters then there exists an adaptation space with  $n$  dimensions. When adapting the transmission from some point

within that space to meet a new target bit rate, the adaptive server should select the encoding configuration that maximizes the user-perceived quality for that target bit rate. When the transmission is adjusted across its full range, the locus of these selected encoding configurations should yield an optimum adaptation trajectory (OAT) within that adaptation space. Although this approach is applicable to any type of multimedia content, the work presented here focuses for concreteness on the adaptation of MPEG-4 video streams within a two dimensional adaptation space defined by frame rate and spatial resolution. These encoding variables were chosen as they most closely map to the spatial and temporal complexities (action and detail) of the video content.

An OAT indicates how the transmission should be adapted (upgraded or downgraded) so as to maximize the user-perceived quality. Extensive subjective testing presented here suggests that an OAT does exist and that it is related to the spatial and temporal characteristics of the content. Knowledge of these OATs can be used as part of an adaptation strategy that aims to maximize the user-perceived quality of the delivered multimedia content.

### 1.4. Contributions

This thesis draws from several different domains in multimedia and networking: video encoding, objective quality metrics, subjective methodologies, human perception, networking and adaptive streaming. It demonstrates and addresses the need for knowledge of user perception in an adaptive streaming system, and makes the following contributions:

1. Proposes the concept of an OAT, which exists in an adaptation space defined by spatial resolution and frame rate.
2. Presents a subjective methodology that can be used for discovering the OAT of any content type. The OATs discovered for several different content types are presented.
3. Suggests that the OAT cannot be discovered using objective metrics.
4. Suggests that 2-dimensional adaptation using an OAT out-performs one-dimensional adaptation in terms of spatial resolution and frame rate.
5. Demonstrates how the OAT can be used to complement any existing sender-based adaptation algorithm.
6. Proposes and develops a new Perceptual Quality Adaptation (PQA) algorithm that directly uses the OAT as a means of making adaptation decisions. An adaptive streaming system was developed for a wireless IP environment and the behaviour of PQA was tested in this environment.

## 1.5. Thesis Outline

The rest of this thesis is organised as follows:

**Chapter 2** provides a high level overview of various aspects of streaming multimedia. MPEG-4 is a popular encoding standard used for streaming multimedia. There are many aspects to this standard, which constrain how the media may be encoded and transmitted, such as the MPEG-4 profiles. The transport protocols used for real-time streaming over IP networks are described. These include transport protocols for media data delivery, feedback information, session description and session control. The chapter concludes by describing the different types of streaming, the characteristics and challenges of streaming over IP networks.

**Chapter 3** contains a literature review of key developments and research pertinent to this thesis. There are three domains described, video quality adaptation, objective video quality analysis and subjective testing methodologies. There has been much research into the effects that network congestion has on the playout of multimedia. In response to this, many adaptation algorithms have been developed. These include rate control, rate shaping, adaptive encoding techniques and transcoder techniques. In addition, there has been much research in the development of objective video metrics. Subjective methods were used to discover the OAT, however there are many aspects to subjective testing that had to be considered.

**Chapter 4** presents the concept of the OAT and how the OAT was discovered using subjective methods.

**Chapter 5** compares the OATs discovered against those discovered using objective metrics. Finally, subjective tests were performed to determine whether two-dimensional adaptation using the OAT was indeed better than one-dimensional adaptation.

**Chapter 6** describes an adaptive streaming system that was implemented to use the OATs in an adaptive streaming environment. The OAT is demonstrated to work with a well-known sender-based adaptation algorithm. Further a new algorithm was developed that directly uses knowledge of the OAT to make adaptation decisions.



## Chapter 2

# Multimedia Streaming

Multimedia is the field concerned with the computer-controlled integration of text, graphics, drawings, still and moving images (video), animation, audio, and any other media where every type of information can be represented, stored, transmitted and processed digitally. A multimedia application is an application, which makes use of a collection of multiple media sources. A multimedia system is a system capable of processing multimedia data and applications. Multimedia systems can be characterised by the processing, storage, generation, manipulation and rendition of multimedia information. One such multimedia system consists of a server, which encodes an audio-visual event. The server transmits the encoded media as a stream to a client application, which will decode the media and play out the event on a player application in a timely manner.

### 2.1. Multimedia Encoding

An important phase in video streaming is how the multimedia should be encoded. For a multimedia system to function, there must be some means to encode and decode the content. There are 2 standardisation bodies in setting video compression standards. These are the ITU (International Telecommunications Union) [1] and the MPEG (Motion Picture Experts Group) [2]. Over the years, both standardisation bodies have produced standards for the encoding and decoding of video content [3] (Figure 2.1).

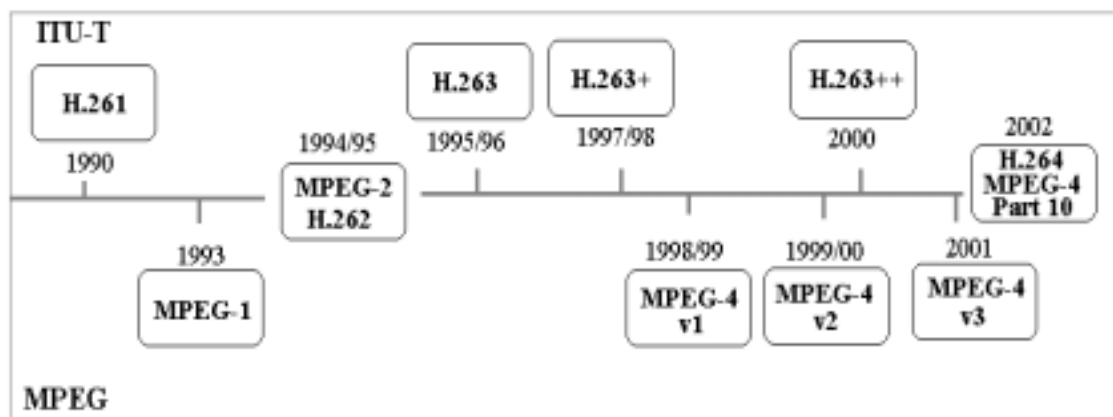


FIGURE 2.1: EVOLUTION OF ENCODING STANDARDS

#### *MPEG*

MPEG is the Motion Picture Experts Group, established in 1988 and is a working group of

ISO/IEC in charge of the development of standards for coded representation of digital audio and video. Each compression standard is designed for a specific target application and bit rate. There are several standards for the encoding of multimedia and others for the description and retrieval of multimedia [4].

- MPEG-1: Designed for up to 1.5 Mbps. Standard for the compression of moving pictures and audio. This was based on CD-ROM video applications, and is a popular standard for video on the Internet, transmitted as .mpg files. In addition, level 3 of MPEG-1 is the most popular standard for digital compression of audio, known as MP3. MPEG-1 is the standard of compression for VideoCD (VCD), the most popular video distribution format throughout much of Asia.
- MPEG-2: Designed for between 1.5 and 15 Mbps. MPEG-2 visual is equivalent to H.262. Standard on which Digital Television set top boxes (SDTV), high definition television (HDTV), video on demand (VoD) and DVD compression is based. It is based on MPEG-1, but designed for the compression and transmission of digital broadcast television. The most significant enhancement from MPEG-1 is its ability to efficiently compress interlaced video.
- MPEG-4: Designed for very low bit rates to very high bit rates. MPEG-4 targets Internet and wireless streaming applications. MPEG-4 is based on object-based compression, which allows individual objects within a scene to be tracked separately and compressed together resulting in very efficient and scalable compression.
- MPEG-7: this standard, currently under development, is also called the Multimedia Content Description Interface. The standard will provide a framework for multimedia content that will include information on content manipulation, filtering and personalization. MPEG-7 will represent information about the content.
- MPEG-21: this standard, currently under development, also called the Multimedia Framework. MPEG-21 will attempt to describe the elements needed to build an infrastructure for the delivery and consumption of multimedia content, and how they will relate to each other.

### *ITU Standards*

There are 2 different standards set by the ITU. Within the ITU, there are several study groups; the ITU-T study group 16 is focused on multimedia services, systems and terminals.

- H.261: This is a low complexity, low latency video standard for bit rates of  $nx64$  kbps. This is targeted at teleconferencing applications and intended to carry video over ISDN.
- H.263: This is very similar to H.261 and has overtaken H.261. It is targeted at videoconferencing applications. It achieves very high compression ratios. H.263+ and

H.263++ support more bit rates with options such as error resilience and scalability.

- H.264: The official ITU-T term is H.264 [5], however it is also known as MPEG-4 part 10, MPEG-4 AVC, JVT and H.26L. H.264 extends MPEG-4 coding by increasing the compression ratios further making it suitable for target applications such as mobile streaming to videophones and mobile devices.

### 2.1.1. MPEG-4

MPEG-4 dramatically advances audio and video compression, enabling the distribution of content and services from low bandwidths to high-definition quality across broadcast, broadband, wireless and packaged media. The MPEG-4 standard provides a set of technologies to satisfy the needs of content providers, service providers and end users [6]. There are a number of elements in the MPEG-4 standard. There are four elements to the MPEG-4 standard [7],

1. MPEG-4 Systems
2. MPEG-4 Visual
3. MPEG-4 Audio
4. Delivery Multimedia Integration Framework (DMIF)

In MPEG-4, as the audio and visual media can be transmitted and stored separately, the client needs to be able to compose these individual media elements to recreate the actual multimedia presentation. The systems part of the MPEG-4 describes the relationship between the audio-visual components that constitute a multimedia presentation [8].

Error resilience allows multimedia to be accessed over a wide range of storage and transmission media, in particular, in error-prone environments at low bit-rates (i.e., less than 64kbps). This is particularly important for wireless networks. Error resilience techniques are not unique to MPEG-4, but are employed and used by MPEG-4 [9][10].

Previous MPEG standards did not have an explicit file format. In these files, absolute timestamps were embedded and the data stream was fragmented for a specific transport media (e.g. MPEG-2 Systems transport system). These characteristics made random access, editing or reuse of the streams difficult without decoding, multiplexing and then rebuilding the stream after editing. MPEG-4 was designed for a generic network medium and as such there is no preferred transport protocol, so fragmenting the data to suit the specific protocol was not acceptable. MPEG-4 media data is stored in its natural or base state, not preferring any one transport protocol or system. When the data needs to be streamed, special

instructions or “hints” for the protocol optionally stored in the file, instruct streaming servers in the process of fragmenting and timestamping the data.

## 2.2. Transport Protocols

There are many transport protocols that can be used for streaming multimedia over best-effort networks. These protocols are designed to enhance the services of the underlying transport protocol (i.e. TCP/IP or UDP/IP). The Real-time Transport Protocol (RTP) is used for facilitating the streaming of multimedia data packets whilst the Real-time Transport Control Protocol (RTCP) is used to relay feedback information between client and server. Feedback messages are used as a basis for quality of service monitoring and multimedia adaptation. The Real-time Streaming Protocol (RTSP) is used to relay control information. Such control information allows for connection establishment and negotiation of session parameters. The session parameters are typically conveyed through a session description using the Session Description Protocol (SDP).

### **2.2.1. Real-time Transport Protocol (RTP) and Real-time Transport Control Protocol (RTCP)**

The main problem with UDP/IP as a transport mechanism is that there is no guarantee that the packets will arrive, and once lost or delayed past their playtime they are discarded. However using another transport mechanism such as TCP/IP would be wasteful, as it has a larger header overhead and requests retransmission of all lost or delayed packets. Retransmission of all lost packets is usually unsuitable for real-time applications, as this would cause undue traffic on the network and the retransmitted packets may be too late in any case. To solve this, RTP was designed for the transport of real-time data, including audio and video [11][12].

RTP consists of two closely linked parts:

- The real-time transport protocol (RTP), to carry data that has real-time properties.
- The RTP control protocol (RTCP), to monitor the quality of service and to convey information about the participants in an on-going session.

RTP is a transport protocol for real-time applications and provides end-to-end network transport functions suitable for applications transmitting real-time data, such as audio, video etc. over multicast or unicast network networks. RTP can be used with other network protocols such as AAL5/IP, UDP/IP and TCP/IP [13].

The RTP control protocol (RTCP) is based on the quasi-periodic transmission of control packets to all participants within a session. The primary function of RTCP is to provide feedback on the quality of the data distribution. This is an integral part of RTP's role as a transport protocol and is related to the flow and congestion control functions of other transport protocols. It is important to get feedback from the receivers to diagnose faults in the distribution. Sending reception feedback reports to all participants allows the server to observe problems and evaluate whether those problems are local or global. This feedback function is performed by the RTCP sender and receiver reports.

### **2.2.2. Real-Time Streaming Protocol (RTSP)**

The Real-Time Streaming Protocol (RTSP) establishes and controls either one or several multimedia streams [14]. It is not concerned with how the media stream itself is delivered. RTSP acts as a “network remote control” for multimedia servers. RTSP is an application-level protocol for control over the delivery of data with real-time properties. The main control methods in RTSP are, DESCRIBE, SETUP, PLAY, PAUSE and TEARDOWN. Others include RECORD, ANNOUNCE, GET\_PARAMETER, REDIRECT and OPTIONS. It has an extensible framework to enable controlled, on-demand delivery of real-time data or pre-encoded data, such as audio and video. The set of streams to be controlled is defined by a presentation description, for example, Session Description Protocol (SDP). The streams controlled by RTSP can use any transport mechanism, RTP etc. to carry the media stream.

### **2.2.3. Session Description Protocol (SDP)**

SDP is intended to describe multimedia sessions for the purposes of session announcement, session invitation, and other forms of multimedia session initiation [15]. SDP is not intended for negotiation of media encodings. SDP is purely a format for session description. It does not incorporate a transport protocol, and is intended to use different transport protocols as appropriate including the Session Announcement Protocol (SAP), Session Initiation Protocol (SIP), and Real-Time Streaming Protocol (RTSP). The purpose of SDP is to convey information about media streams in multimedia sessions to allow the recipients of a session description to participate in the session.

## **2.3. Streaming**

Streaming is a server/client technology that allows multimedia data to be transmitted and consumed. Streaming applications include e-learning, video conferencing, video on demand

etc. The main goal of streaming is that the stream should arrive and play out continuously without interruption. In general, streaming involves sending multimedia (e.g., audio and video) from a server to a client over a packet-based network such as the Internet. There are two main types of streaming, progressive streaming and real-time streaming.

Progressive streaming is often called progressive download. In progressive streaming a compressed video file is transferred to the hard disk of the client progressively. This streaming method is used typically when the movie size is relatively short (i.e. less than three minutes), for example, movie trailers, short movie clips, advertisements etc. However, depending on the format of the video, some progressive files require that the entire movie be downloaded before it can be played (e.g. Real). Progressive streaming is not a good solution for long movies or material where the user may want random access. Progressive streaming is not suited for “live” on-demand multimedia. The client has no interaction with the progressively streamed multimedia and cannot fast-forward or rewind to portions of the stream. Progressive streaming does not adapt to fluctuations in the clients’ bandwidth.

In real-time streaming the multimedia file is transmitted and consumed by the client in real-time or near real-time. When the client connects to the server, the stream will begin to play out on the clients’ machine automatically or after a short delay of 1 or 2 seconds. Real-time streaming is suited for longer videos, for example, live broadcasts, presentations, training videos and lectures. However, real-time streaming is constrained by fluctuations in network conditions. An adaptive streaming server keeps track of the network conditions and adapts the quality of the stream to minimize interruptions and stalling. Typically, the Real-time Transport Protocol (RTP) is used break the encoded media data into a series of time-stamped packets called a stream whilst the server monitors the state of the network using the Real-time Transport Control Protocol (RTCP). The client can interact with the real-time streaming server using Real Time Streaming Protocol, RTSP. If the requested file is pre-encoded, the client can jump to any location in the video clip using the RTSP protocol as a network remote control for the stream. Real-time streaming can be delivered by either peer-to-peer (unicast) or broadcast (multicast).

There are two types of real-time streaming services [16][17], on-demand or live streaming.

- On-demand streaming: pre-compressed, streamable multimedia content is archived on a storage system such as a hard disk drive and transmitted to the client on demand by the streaming server. Client requests are asynchronous, in that, clients can watch different parts of the media at the same time.
- Live streaming: the multimedia stream comes from a live source such as a video

camera and is compressed in real-time by the encoder. There is no notion of duration as these streams are live. Clients cannot avail of fast-forward, rewind functions.

### **2.3.1. Challenges for Streaming Multimedia**

There are many factors, which affect the delivery of multimedia traffic over a best-effort wired and wireless IP networks. These are:

1. Heterogeneity
2. Congestion
3. Bandwidth fluctuations due to congestion and/or mobility.
4. Delays due to congestion, packet loss and/or retransmissions.
5. Loss due to congestion is the biggest factor affecting video quality.
6. Noise and interference, which is particularly evident in wireless networks.

#### *Heterogeneity*

There are two kinds of heterogeneity, namely, network heterogeneity and receiver heterogeneity. Network heterogeneity refers to the subnets in the network having different capacities and resources (e.g., processing, bandwidth, storage and congestion control policies). Network heterogeneity can cause each user to have different packet loss/delay characteristics. This is a problem for designing or predicting behavior of transmission over the network, as the application cannot make any assumptions about how the network will treat a particular packet. For example, if an application used prioritization of packets by using the Type Of Service (ToS) field in the IP header, various subnets in the network will interpret this ToS field differently.

Receiver heterogeneity refers to clients having different delay requirements, visual quality requirement, end device (PDA, laptop, mobile phone) and/or processing capability. In multicast sessions, this heterogeneity can cause a big problem, as all the receivers have to receive the same content with the same quality. Multicast can achieve high bandwidth efficiency while unicast can be inefficient if there is more than one user requesting the same content. The efficiency of multicast is achieved at the cost of losing the service flexibility of unicast.

#### *Congestion*

Congestion occurs when the amount of data in the network exceeds the capacity of the network. As traffic increases, routers are no longer able to cope with the load and this results

in lost packets. Congestion can be caused by several factors.

- Queues can build up caused by high data rate applications. If there is not enough memory to hold all the data at the router, packets will be lost. But even if queues had an infinite length, this cannot eliminate congestion, as by time a packet is at the top of the queue, the packet has already timed out and duplicates have already been sent. All these packets are then forwarded onto the next router, increasing the load all the way to the destination.
- Slow processors can also cause congestion. If the routers CPU are slow at performing its tasks of queuing buffers, updating tables etc. queues can build up even though there is excess capacity.
- Bottleneck bandwidth links cause congestion.

Congestion tends to be cyclic, in that, if a router has no free buffers, it will ignore and drop arriving packets. If the packet is TCP, when a packet is discarded, the sending routers may timeout and retransmit the packet, as the packet cannot be dropped until it is acknowledged resulting in a backup of congestion.

### *Bandwidth Fluctuations*

To achieve acceptable presentation quality, transmission of real-time multimedia typically has minimum bandwidth requirement. However, due to the best-effort nature of the Internet and wireless IP networks, there is no bandwidth reservation to meet such a requirement. Generally, routers do not actively participate in congestion control [18], excessive traffic can cause congestion collapse, which can further degrade the throughput of real-time multimedia.

In wireless networks there may be several reasons for bandwidth fluctuations:

1. When a mobile terminal moves between different networks (e.g., from wireless local area network (LAN) to wireless wide area network (WAN)), the available bandwidth may vary drastically (e.g., from a few megabits per second (Mbps) to a few kilobits per second (kbps)).
2. When a handover happens, a base station may not have enough unused radio resource to meet the demand of a newly joined mobile host.
3. The throughput of a wireless channel may be reduced due to multipath fading, co-channel interference, and noise disturbances.
4. The capacity of a wireless channel may fluctuate with the changing distance between the base station and the mobile host.



### Delay

There are many sources of delay in any transmission system (Figure 2.2). In the network itself, in addition to propagation delays, further delays are incurred at each router along its path due to queuing and switching at various routers. At the end-points, delays are incurred in obtaining the data to be transmitted and packetising it. Real-time multimedia is particularly sensitive to delay, as multimedia packets require a strict bounded end-to-end delay. That is, every multimedia packet must arrive at the client before its playout time, with enough time to decode and display the packet. If the multimedia packet does not arrive on time, the playout process will pause, or the packet is effectively lost. Congestion in a best-effort IP network can incur excessive delay, which exceeds the delay requirement of real-time multimedia. In a wireless network, there are additional sources of delay such as retransmissions on the radio link layer.

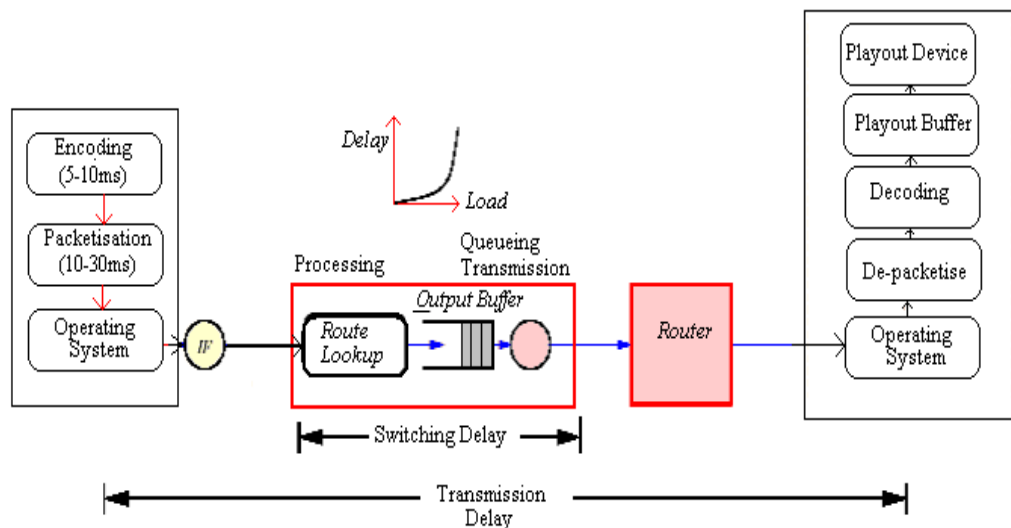


FIGURE 2.2: DELAY SOURCES IN A TRANSMISSION SYSTEM

### Jitter

A variable delay on the IP networks is known as jitter. Jitter is mainly due to queuing and contention of packets at intermediate routers, but can also happen when packets take a different path to the destination. To combat jitter, playout buffering is used at the receiver. Jitter does not have a huge impact on streaming multimedia transmission, which is in general not a highly interactive application and as the delay bounds are less stringent than those for interactive sessions such as VoIP [19]. The trade off is that for higher interactivity, there is less receiver buffering which means that the effects of network jitter increases. However, for applications with lower interactivity, there is greater receiver buffering and so the effects of network jitter become less. In receiver buffering, packets that arrive faster than expected are placed in a buffer for playout at a later time (Figure 2.3). Smooth quality of a received multimedia signal depends on appropriate buffering. The receiver buffer must be

large enough to account for network jitter, and if enough data are buffered, also enables the receiver to sustain momentary drops in the sending rate by playing out of its buffer at a higher rate than the sender is currently sending. The buffer accumulates when the sender is transmitting faster than the receiver is playing out, by not aggressively increasing the sending bandwidth when the available bandwidth capacity is increased.

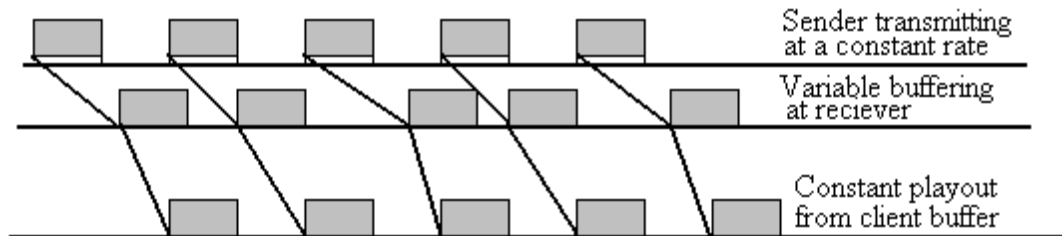


FIGURE 2.3: RECEIVER JITTER BUFFERING

### *Loss*

For streamed multimedia applications, loss of packets can potentially make the presentation displeasing to the users, or, in some cases, make continuous playout impossible. Multimedia applications typically impose some packet loss requirements. Specifically, the packet loss ratio is required to be kept below a threshold (say, 1%) to achieve acceptable visual quality. Despite the loss requirement, the current Internet does not provide any loss guarantee. In particular, the packet loss ratio could be very high during network congestion, causing severe degradation of multimedia quality. Even if the network allows for re-transmission of lost packets, as is the case for wireless IP networks, the retransmitted packet must arrive before its playout time. If the packet arrives too late for its playout time, the packet is useless and effectively lost.

Congestion at routers in the network often results in queue overflow, which generally results in packets being dropped from the queue. Often consecutive packets in a queue are from the same source and belong to the same media stream. So, when packets are being dropped, they often belong to the same stream. Thus, this behavior of packet dropping in the network can be seen as bursty losses. Packet loss and delay can exhibit temporal dependency or burstiness [20]. For instance, if packet  $n$  has a large delay, packet  $(n + 1)$  is also likely to do so. This translates to burstiness in network losses and late losses, which may worsen the perceptual quality compared to random losses at the same average loss rate. There has been much work on modeling temporal loss dependency. It can be approximated by a Bernoulli model [21],[22],[23] but a Markov model is better suited to capture this temporal loss dependency. A simplified 2-state Markov model, known as the Gilbert model is often used and shows that when a packet is lost, there is a higher probability that the next packet will

be lost. Bursty losses can affect the efficacy of error resilience and error concealment schemes. It affects performance of error concealment techniques such as Forward Error Correction (FEC). FEC can recover a lost packet only if other necessary packets belonging to the same block are received. In this way, the loss pattern affects the effectiveness of loss concealment. Even though bursty losses are likely, packet interleaving reduces this effect [24]. If the audio and video packets are interleaved streams, then the likelihood of consecutive video packets being lost is reduced.

### *Noise*

Compared with wired links, wireless channels are typically much more noisy and have both small-scale (multipath) and large-scale (shadowing) fades [25], making the Bit Error Rates (BER) very high. The resulting bit errors can have devastating effect on multimedia presentation quality [26].

### **2.3.2. Commercial Streaming Solutions**

The dominant streaming solutions are Apple's QuickTime, RealNetworks' RealMedia, and Microsoft Windows Media Player [27]. The main problem with so many streaming technologies is lack of compatibility between platforms.

#### *Apple's QuickTime*

Developed by Apple, QuickTime is a simple cross-platform architecture (for Macintosh, Unix, and PC). Its native video format is “.mov”. The Quicktime file format is “.mov”. This file format was flexible and general enough to be adopted as a framework for the MP4 file format [28]. QuickTime 6.3 enables users to share high-quality video, audio and text on a new generation of wireless devices including cell phones and PDAs. It delivers extensive support for the 3GPP standard, including Advanced Audio Coding (AAC) and Adaptive Multi-Rate (AMR) audio, MPEG-4 and H.263 video, 3G Text (TX3G) and native .3gp file format support. 3GP is the industry standard for streaming to wireless devices, such as videophones and PDAs.

#### *RealNetworks' RealVideo*

RealMedia (RealVideo or Real for short) is developed by RealNetworks. It is a network oriented, streaming media platform. Real's playing software, RealPlayer is widespread all over the Internet. For compression and decompression, RealMedia uses its proprietary Real G2 Codec technology. RealSystems Architecture (includes RealSystem Proxy and RealSystem Server Proxy) delivers live content to the end user either by unicast or by

multicast. For on-demand media, proxy supports streaming from local cache and pass-through delivery. Real supports both RTP and RDT (RealNetworks proprietary transport protocol, similar to RTP) data packet formats. RealSystems does not follow ISMA guidelines and instead uses its own proprietary encoding (i.e. RealAudio, RealVideo and RealMedia) and a proprietary transport protocol, RDT, which is very similar to RTP.

### *Microsoft's Windows Media*

Windows Media (formerly called "NetShow") is Microsoft's solution to the development of streaming media architectures. Windows runs its own proprietary server protocol: Microsoft Media Server Protocol (MMS) instead of the standard RTSP, which is freely available and runs on the widely available Windows Server Platforms. Like Real, Windows Media concentrates on network delivery of video and audio. Windows Media allows server-based streaming as well as serverless, HTTP streaming. The standard file for Windows Media is Active Streaming Format (ASF). In Windows Media software package, there are Windows Media Tools for creating streaming media, Windows Media Services for hosting and delivering streaming media, and Windows Media Player for playing streaming media. Windows Media servers support what is called "intelligent streaming": delivering the most suitable version of content to users based on the available Internet bandwidth between the server and the users who need the file. If the bandwidth decreases, the media server will send less video data accordingly. ASF files provide more than one versions of the video track for both high- and low-bandwidth connections. Windows Media Video supports MPEG-4. However powerful Windows Media might be, Windows Media's lack of support for the current version of SMIL represents a shortcoming for this good technology.

### **2.3.3. Future trends in streaming**

Although Windows Media, Quicktime and Real Media are the main players, an open format called MPEG-4 is becoming the de facto video codec standard. MPEG-4 is the preferred format for streaming for several reasons:

- Ubiquity: Streaming video is being applied to diverse range of devices, appliances, platforms and computers, for example, desktop computers, PDA's, mobile phones, Personal Video Recorders. MPEG-4 has the flexibility and diversity to support streaming to a range of devices over various networks, i.e. wired and wireless.
- Unified Standard: MPEG-4 is an open standard that is being integrated into most codecs. This means that the content can be played out on various different players.
- Quality: MPEG-4 is a highly efficient compressor and can encode high-quality video from low to high bit rate ranges.

# Chapter 3

## Literature Review

### 3.1. Introduction

Within the scope of this work, there are three different research areas, multimedia adaptation techniques, objective video quality metrics and subjective methodologies. This chapter contains a literature review of key research and developments in each of these domains.

### 3.2. Multimedia Adaptation Techniques

Given the seriousness of congestion on the smooth continuous playout of multimedia, there is a need for strong need for adaptation. The primary goal of adapting multimedia is to ensure graceful quality adaptation and maintain a smooth continuous playout. Multimedia servers should be able to adapt quality to bandwidth variations. If the available bandwidth becomes smaller than the sending rate of the multimedia, the server should tailor multimedia quality intelligently rather than arbitrarily drop packets for example, to match the available network resources.

Adaptation should behave fairly in a shared network. When there is excess bandwidth, the competing traffic flows share the excess bandwidth in a fair manner. A TCP rate control mechanism is not practical for real-time multimedia traffic. High bandwidth multimedia flows that do not follow TCP congestion control mechanisms can starve peer TCP flows in periods of congestion.

Multimedia flows should not use TCP:

- TCP multiplicatively decreases the window size on encountering a packet loss. A sudden decrease of the window size may cause a sharp bit rate variation, which can seriously degrade the perceived quality by the receiver.
- Multimedia applications are loss tolerant and time sensitive so retransmission of lost packets is not very efficient.
- Per-packet acknowledgements impose a large overhead.

Instead, multimedia flows should behave in a TCP-friendly manner. TCP friendly means that non-TCP flows receive the same share of the bandwidth as TCP flows over the same link [29].

In the development of adaptation algorithms, there are a number of design and system issues, which must be considered [30].

1. Any signaling or feedback mechanisms needed to convey congestion information between client and server.
2. Frequency of feedback.
3. The specific rate control mechanism used in response to feedback.
4. The responsiveness of the congestion control scheme in detecting and reacting to network congestion, for example upon detecting congestion, does the adaptation scale back gradually or dramatically.
5. The capability of the adaptation algorithm to accommodate heterogeneous receivers with differing network connectivity, the amount of congestion on their delivery paths, and their need for transmission quality.
6. The scalability of the adaptation algorithm in a multicast session with a large number of receivers.
7. Fair sharing of bandwidth with competing connections, particularly TCP connections.
8. The perceived quality of received multimedia streams.

### 3.2.1. Adaptation Techniques

Broadly speaking, adaptation techniques attempt to reduce network congestion by matching the rate of the multimedia stream to the available network bandwidth. Without some sort of rate control, any data transmitted exceeding the available bandwidth would be discarded, lost or corrupted in the network. Adaptation techniques can be classified by the place where adaptation occurs, i.e. receiver-based, sender-based, hybrid schemes. Equally, they can be classified by the method in which the adaptation occurs (Figure 3.1). These are [31][32]:

- Rate Control: Transport level
  - Sender-based
  - Client-based
  - Hybrid
- Rate Shaping: Transport and encoding level
  - Sender-based
  - Client-based
  - Hybrid
- Encoder-based adaptation: Encoding level
  - Sender-based
- Transcoder: Network level.

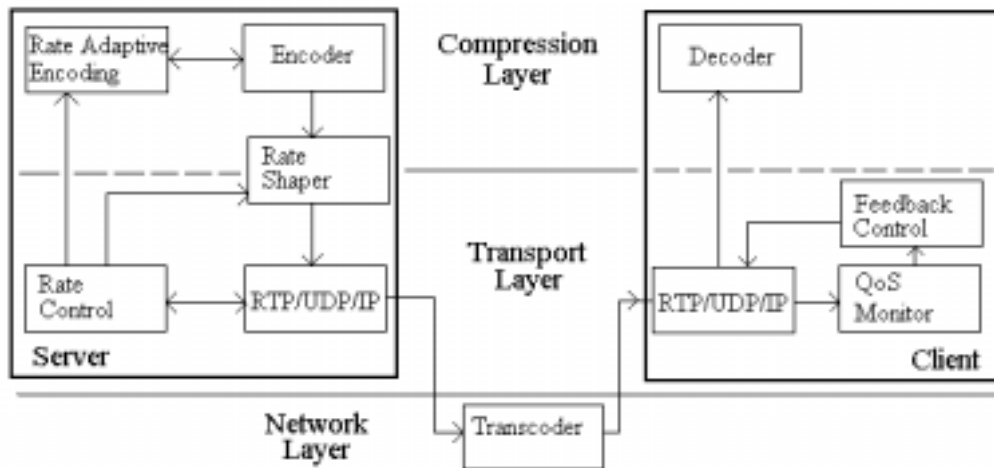


FIGURE 3.1: ADAPTATION TECHNIQUES

One of the main objectives of congestion control is to prevent or minimize packet loss. However, packet loss is often unavoidable in the best-effort IP networks and may have significant impact on perceptual quality. Error control mechanisms are used to maximize multimedia presentation quality in presence of packet loss (Figure 3.2.). These mechanisms can be classified as:

- Forward error correction (FEC).
- Retransmission.
- Error-resilience.
- Error concealment.

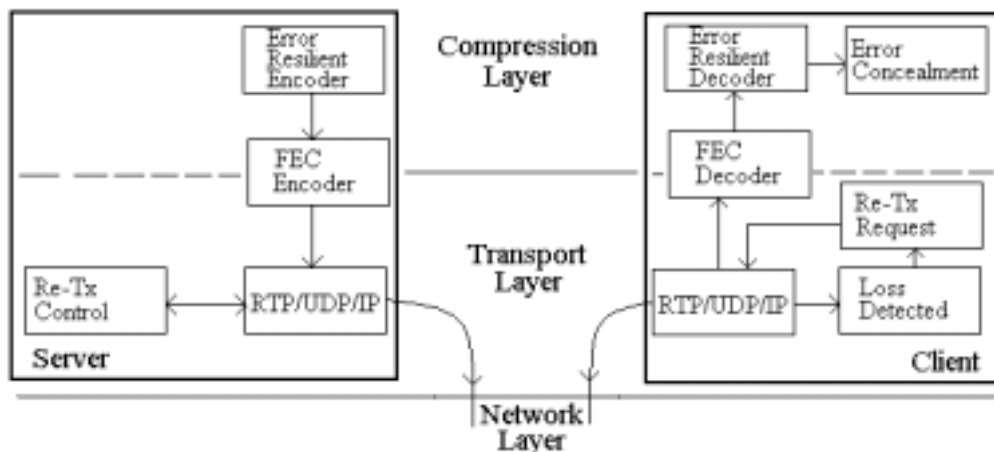


FIGURE 3.2: ERROR CONTROL TECHNIQUES

In general, end-to-end schemes, either sender-based or client-based, have been and are still a popular topic of research. In particular much of the research has focused on rate control solutions especially sender-based rate control schemes. End-to-end schemes are easy to deploy and can be applied to most systems. Encoder-based and rate shaping solutions are

constrained by the functionality of the codec being used. Transcoder based schemes require assumptions and interactions with the network. Currently transcoder based schemes are not very popular and there is very little new research in this area. Despite the efficacy of an adaptation algorithm, there may be situations in which there is unavoidable loss of data. For this reason, often adaptation in a system is employed with some error control techniques. The next few sections will give an overview of several important and key research developments for each of these adaptation schemes. Chapter 3 Section 2.7 contains a discussion and critique of these schemes.

### 3.2.2. Rate Control

TCP retransmission and acknowledgements introduce unnecessary delays and overhead in the network that are not beneficial for real-time multimedia applications. UDP is usually employed as the transport protocol for real-time multimedia streams. However, UDP is not able to provide congestion control and overcome the lack of service guarantees in the network. Therefore, it is necessary to implement control mechanisms above the UDP layer to prevent congestion and adapt accordingly.

There are two types of control for congestion prevention:

1. **Window-based control** [33]: probes for the available network bandwidth by slowly increasing a congestion window; when congestion is detected (indicated by the loss of one or more packets), the protocol reduces the size of the congestion window. The rapid reduction of the window size in response to congestion is essential to avoid network collapse.
2. **Rate-based control** [31][34]: sets the sending rate based on the estimated available bandwidth in the network; if the estimation of the available network bandwidth is relatively accurate, the rate-based control could also prevent network collapse. Rate-based control is usually employed for transporting real-time multimedia. Rate control can take place at the server, the receiver or a hybrid scheme.

#### 3.2.2.1. Server-Based Rate-Based Control

The server adapts the transmission rate of the multimedia stream being transmitted. The server minimizes the levels of packet loss at the client by matching the transmission rate of the multimedia stream to the available network bandwidth. Without any rate control, the data transmitted exceeding the available bandwidth would be discarded in the network. Server-based rate control schemes require a feedback mechanism to convey the state of the network to the server. Using the feedback, the server adapts the rate of the multimedia



stream. Rate control can be applied to both unicast [35] and multicast streams. For unicast streamed multimedia, server-based rate control mechanisms can be classified into two approaches, the probe-based approach and the model-based approach.

1. **Probe-based:** The server probes for the available network bandwidth by adjusting the sending rate so that some QoS requirements are met, for example, loss rates are below a certain level.
2. **Model-based:** The server models its transmission based on the throughput of a TCP connection. The server uses the formula for TCP throughput to determine the sending rate of the multimedia stream. In this fashion, the server can avoid congestion and compete fairly with TCP flows.

### *X-Increase Y-Decrease Algorithms*

All adaptation algorithms behave in an X-Increase/Y-Decrease manner (Figure 3.3). When there is no congestion, the server increases its transmission rate by X, and when there is congestion it decreases its transmission rate by Y. There are many ways to adjust the sending rate, for example,

- Additive Increase/Multiplicative Decrease (AIMD)
- Additive Increase/Additive Decrease (AIAD),
- Additive Increase/Proportional Decrease (AIPD) [36],
- Multiplicative Increase/Multiplicative Decrease (MIMD) [34].

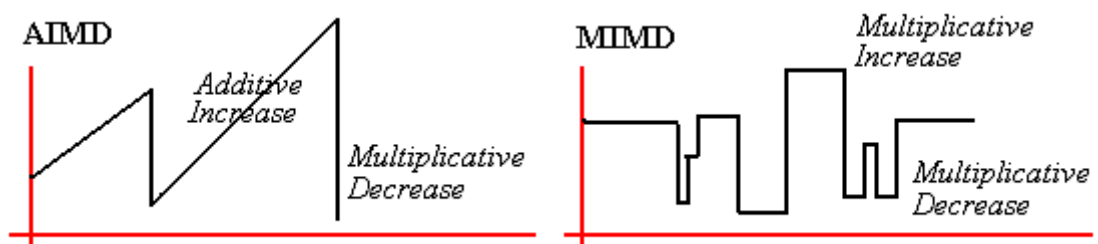


FIGURE 3.3: AIMD AND MIMD

For example, the classical AIMD algorithm operates as follows. When the sender receives loss feedback from the client (or from the network using an explicit congestion notification (ECN) mechanism [37]). The sender adjusts its sending rate based on the loss feedback as follows [38]:

$W_i$  = the window size at time of feedback message  $i$ .

$\alpha$  = the increase constant, where ( $\alpha > 0$ ).

$\beta$  = the decrease constant, where ( $0 < \beta < 1$ ).

$p$  = the fraction of packet loss.

If ( $p=0$ )

$$\{ W_{i+1} = W_i + \alpha \}$$

Else

$$\{ W_{i+1} = W_i * (1 - \beta) \}$$

In TCP, typically  $\alpha=1$  and  $\beta=0.5$ . Upon packet loss, the AIMD sender decreases its window by  $\beta$  conforming to the congestion control mechanism of TCP.

### *Server-based Rate Control*

The server-based rate control is an extension of the AIMD algorithm. It attempts to reduce the packet loss ratio,  $p$ , below a certain threshold level,  $P_{TH}$ . The value of  $P_{TH}$  is the maximum tolerated loss in the video stream at the receiver.

$p$  = packet loss ratio,

$P_{TH}$  = the threshold for the packet loss ratio,

$r_i$  = the sending rate at the server,

$AIR$  = the additive increase rate,

$MaxR$  and  $MinR$  are the maximum rate and the minimum transmission rate of the sender respectively,

$\beta$  = the multiplicative decrease factor.

If ( $p \leq P_{TH}$ )

$$\{ r_{i+1} = \text{Min} [(r_i + AIR), MaxR] \}$$

Else

$$\{ r_{i+1} = \text{Max} [(\beta * r_i), MinR] \}$$

The packet loss ratio,  $p$ , is measured by the receiver and conveyed back to the sender. When the server receives this loss ratio value from the client, it adapts its sending rate as per the AIMD algorithm and chooses the minimum value for the transmission rate, either  $r_{i+1} = (r_i + AIR)$  or  $r_{i+1} = MaxR$ . Similarly, when increasing the transmission rate, the server chooses the maximum value either  $r_{i+1} = \beta * r_i$  or  $r_{i+1} = MinR$ .

### *Rate Adaptation Protocol (RAP)*

RAP [39] is sender-based AIMD rate adaptation with TCP friendliness. Servers send data packets with sequence numbers and receivers acknowledge each packet containing the corresponding sequence number. The server estimates the round trip time called  $RTT$  and timeout. The server maintains a record, called transmission history, of all outstanding transmitted packets, containing the sequence number, departure time, transmission rate and status flag. Before sending a new packet the server checks for timeouts of packets in the

transmission history with provisions made for detecting ACK losses. Losses are indicated by either gaps in the sequence numbers of the acknowledged packets or timeouts. The transmission rate ( $r_i$ ) is controlled by adjusting the inter-packet gap ( $IPG$ ).

$r_i$  = transmission rate.

$p$  = loss rate;

$\alpha$  = additive increase factor.

$\beta$  = decrease factor.

If ( $p=0$ )

$$\{ r_{i+1} = r_i + \alpha \}$$

Else

$$\{ r_{i+1} = \beta * r_i \text{ where } \beta = 0.5 \}$$

### General AIMD (GAIMD)

GAIMD [40] is a window-based rate adaptation scheme. In GAIMD, the servers' transmission rate is increased by  $\alpha$  if there is no packet loss in a  $RTT$ . During periods of loss, the transmission rate is decreased by a factor of  $\beta$ . The parameters  $\alpha$  and  $\beta$  are adjusted such that the GAIMD flows are TCP friendly. The relationship between  $\alpha$  and  $\beta$  for a GAIMD flow to be TCP-friendly, where the GAIMD flow has approximately the same sending rate as a TCP flow when,

$\alpha$  = increase factor, where ( $\alpha > 0$ )

$\beta$  = decrease factor, where ( $0 < \beta < 1$ )

$$\alpha = \frac{4(1 - \beta^2)}{3}$$

$p$  = loss rate.

$W_i$  = window size

If ( $p=0$ )

$$\{ W_{i+1} = W_i + \alpha / W_i \}$$

The sender increases its transmission rate by  $\alpha/W$  for each new ACK received.

Else

$$\{ W_{i+1} = W_i + \alpha / W_i \}$$

If congestion is detected by triple duplicate ACKS, it reduces the congestion window to  $\beta * W$ . If congestion is due to timeout then the window size is reduced to 1. GAIMD calculates a mean sending rate using  $p$ ,  $RTT$ ,  $t_{RTO}$ ,  $num_{ACKed}$  (number of packets acknowledged by each ACK),  $\alpha$  and  $\beta$ . This mean sending rate has been compared with the actual sending rate and is reasonably accurate for large ranges of  $\alpha$  and  $\beta$  when packet loss rate is less than 20%.

*Binomial Congestion Control*

Binomial control [41] generalizes the TCP's increase/decrease rules to derive a family of TCP-friendly congestion control algorithms with a varying degree of oscillation. This is achieved using the following equations:

$k$  and  $l$  = parameters of binomial controls

$k+l = 1$  and  $l < 1$  for suitable  $\alpha$  and  $\beta$ .

$w_t$  = instantaneous window value, which governs the transmission rate.

$$\text{Increase: } w_{t+1} = w_t + \frac{\alpha}{w_t^k}$$

$$\text{Decrease: } w_{t+1} = w_t - \beta w_t^l$$

Where ( $\alpha > 0$ ) and ( $0 < \beta < 1$ )

For  $k = 0; l = 1$ , the behavior is AIMD used by TCP; most aggressive probing scheme.

For  $k = -1; l = 1$ , the behavior is MIMD used by slow start in TCP);

For  $k = -1; l = 0$ , the behavior is MIAD;

For  $k = 0; l = 0$  the behavior is AIAD.

If the rule ( $k+l=1$ ) is satisfied, the flow is TCP-friendly.

*Model based approaches estimating TCP throughput*

The model-based approaches attempt to estimate the available network bandwidth explicitly by using a stochastic TCP model for deriving a sending rate equation, which is characterized by the following formula [42]:

$\lambda$  = the throughput of a TCP connection.

$MTU$  = (Maximum Transit Unit) is the maximum packet size used by the connection.

$RTT$  = the round trip time for the connection.

$p$  = the packet loss ratio.

$$\lambda = \frac{1.22 * MTU}{RTT * \sqrt{p}}$$

EQN: TCP THROUGHPUT ESTIMATION

This equation is used to determine the sending rate of the video stream. To compute the sending rate  $\lambda$ , the server needs to obtain the  $MTU$ ,  $RTT$ , and packet loss ratio  $p$ . If unknown, the  $MTU$  can be found through the mechanism proposed in [43]. If the  $MTU$  information is unavailable, the default  $MTU$ , i.e. 576 bytes, is used. The parameter  $RTT$  can be obtained through feedback of timing information. In addition, the receiver can periodically send the packet loss ratio,  $p$ , to the server. Upon feedback, the server estimates the sending rate  $\lambda$  and rate control can be applied. There is a drawback with using the throughput equation above as it does not consider timeouts which is a common phenomenon

when error rates are high. Another analytical model used for estimating the average bandwidth share of a TCP connection is [44] which takes into consideration these timeouts.

$t_{RTO}$  = TCP retransmission timeout value.

$D$  = the number of acknowledged TCP packets by each acknowledgment packet.

$\lambda$  = the throughput of a TCP connection.

$MTU$  = (Maximum Transit Unit) is the maximum packet size used by the connection.

$RTT$  = the round trip time for the connection.

$p$  = the packet loss ratio.

$$\lambda = \frac{MTU}{RTT * \sqrt{2Dp/3} + t_{RTO} \text{Min}(1, 3\sqrt{3Dp/8}) p(1 + 32p^2)}$$

EQN: TCP THROUGHPUT

Using this throughput equation, the video flow gets its bandwidth share like a TCP connection and can avoid congestion in a way similar to that of TCP, and can co-exist with TCP flows in a “friendly” manner [45]. Without rate control during periods of network congestion, the video flow lead to the possible starvation of competing TCP flows due to the rapid reduction of the TCP window size in the TCP flows in response to congestion. Different versions of TCP have different algorithms for the window based congestion control scheme, with different throughput characteristics [46]. However, in either equation, it is not clear which version of TCP it is friendly with.

#### *TCP-Friendly Rate Control (TFRC) and TFRC Protocol (TFRCP)*

TFRC [47] is a congestion control algorithm for unicast traffic, which explicitly adjusts its sending rate as a function the measured loss event rate, where a loss event consists of one or more packet dropped within a  $RTT$ . Receivers calculate the loss event rate  $p_e$  and  $RTT$  time and send feedback to the sender at least once per  $RTT$ . The server does not reduce the sending rate in half in response to a single loss event, but reduces the sending rate in half in response to several successive loss events. If the sender has not received feedback after several  $RTT$ , then the sender should reduce its sending rate. Otherwise, senders increase or decrease their sending rate according to the TCP throughput equation [44].

TFRCP presented in [48] is a rate-adjustment congestion control protocol that is based on another function of TCP modeling demonstrated in [44]. The TFRCP sender works in rounds of  $M$  time units. At the beginning of each round, the sender computes a TCP-friendly rate and sends packets at this rate. Each packet carries a sequence number and sending

timestamp. The receiver acknowledges each packet, by sending an ACK that carries the sequence number of the packets received correctly and receiving timestamp. From this timestamp, the sender can calculate the  $RTT$  and  $t_{RTO}$ , and obtain the number of packets dropped,  $num_{drop}$  and the number of acknowledged packets  $num_{ACKed}$  of this round based on the sequence numbers. During periods of loss, the sender reduces its transmission rate to the equivalent TCP rate calculated using the TCP throughput equation, and during periods of no loss, the rate is doubled.

$W_{MAX}$  = the receiver's declared window size.

$t_{RTO}$  = TCP retransmission timeout value.

$r$  = the transmission rate where  $r$  is a function of  $W_{MAX}$ ,  $p_e$ , and  $t_{RTO}$ .

$num_{drop}$  = the number dropped.

$num_{ACKed}$  = the number of correctly acknowledged packets.

If ( $drop = 0$ )

$$\{ r_{i+1} = 2 * r_i \}$$

Else

$$\{ p = num_{drop} / (num_{drop} + num_{ACKed})$$

$$r_{i+1} = fn(W_{MAX}, p, t_{RTO})$$

$$\}$$

This scheme behaves in a TCP-friendly manner during periods of loss. However, its increase behaviour may result in unequal bandwidth distribution due to the possibility of increasing the transmission rate faster than competing TCP connections. TFRC smoothes the reduction of sending rate in response to packet loss and avoids the oscillation of rate when they are used for real-time multimedia applications. However, the model relies on a single TCP connection with a steady-loss ratio. But is not as effective, if the  $RTT$  is affected by queuing delays or when the bottleneck router is shared among competing connections, especially when the dropping rate is large.

#### *Enhanced Loss Delay Adjustment (LDA+)*

LDA+ [49] is an enhanced version of LDA [50][51], which adapts the transmission rate of UDP-based multimedia flows to the congestion situation in the network in a TCP-friendly manner. LDA+ controls the transmission rate of a sender based on end-to-end feedback information about losses ( $p$ ), delays and bandwidth capacities measured by the receiver. LDA+ estimates the flows bandwidth share to be minimally the bandwidth share determined by [44] which is the theoretical TCP-friendly bandwidth share.

$p$  = packet loss rate.

$r$  = the transmission rate.

$b$  = the bottleneck bandwidth.

$AIR$  = the additive increase rate.

$AIR_R$  =  $AIR$  as a function of the transmission rate  $r$ , the sender is sending relative to the bottleneck bandwidth,  $b$ .

$AIR_{MAX}$  = the maximum increase rate limited to the bottleneck bandwidth,  $b$ .

$AIR_{TCP}$  = the increase rate of a TCP connection sharing the same link.

If ( $p=0$ )

The sender estimates its additive increase rate ( $AIR$ ). To allow for a smooth increase of  $AIR$  and to allow for flows of smaller bandwidth shares to increase their transmission rates faster than competing flows with higher shares,  $AIR_R$  is determined.

$$AIR_R = AIR \left(2 - \frac{r}{b}\right)$$

$$AIR_{MAX} = r \left(1 - \exp\left(-\left(1 - \frac{r}{b}\right)\right)\right)$$

$AIR_{TCP}$  is estimated, as an RTP flow should not increase its bandwidth share faster than a TCP connection sharing the same link. With an average value of  $T$  seconds between RTCP-RR reports and a round trip time  $RTT$ , a TCP connection increases its transmission window by  $P$  packets. The window size is increased by one packet each  $RTT$ . The receiver estimates its bandwidth share averaged over  $T$ .

$$P = \sum_{p=0}^{T/RTT} p = \frac{T}{RTT} \left(\frac{T}{RTT} + 1\right) / 2$$

$$AIR_{TCP} = \frac{P}{T} \rightarrow \frac{\left(\frac{T}{RTT} + 1\right)}{2RTT}$$

The  $AIR$  value is then set to

$$AIR_i = \text{Min}(AIR_R, AIR_{MAX}, AIR_{TCP})$$

The new transmission rate for the sender is then:

$$r_i = r + AIR_i$$

Else ( $p>0$ )

In situations of loss, the sender reduces its transmission rate to  $r_i$ , where  $r_{TCP}$  is determined using [52].

$$r_i = \max\left(r(1 - \sqrt{p}), r_{TCP}\right)$$

### 3.2.2.2. Receiver-based Rate Control

In receiver-based rate control, the clients control the receiving rate of video streams by adding/dropping channels or layers. Generally, receiver-based rate control is applied to layered multicast video. In layered multicast, the video sequence is compressed into

multiple layers: a base layer and one or more enhancement layers. The base layer can be independently decoded and provides basic video quality; the enhancement layers can only be decoded together with the base layer and they enhance the quality of the base layer (Figure 3.4). The base layer consumes the least bandwidth; the higher the layer is, the more bandwidth the layer consumes. Each video layer is sent to a separate IP multicast group. Each client subscribes to a certain set of video layers by joining the corresponding IP multicast group. Each client attempts to achieve the highest subscription level of video layers without incurring congestion [53][54][55].

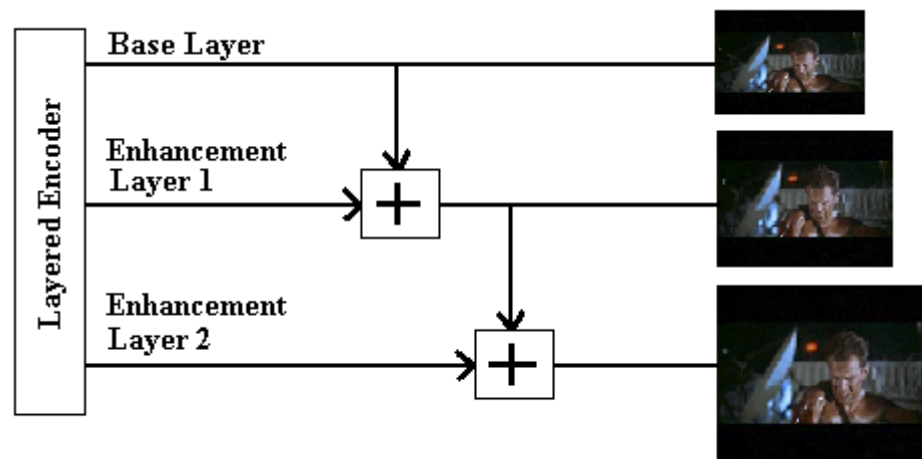


FIGURE 3.4: LAYERED MULTICAST

#### *Model-based Layered Multicast*

The model-based approach [56] attempts to explicitly estimate the available network bandwidth and is based on the throughput model of a TCP connection. In the algorithm, it is assumed that each receiver knows the transmission rate of all the layers.

$r_i$  = the transmission rate of Layer  $i$ .

$\lambda$  = the target transmission rate calculated using the TCP throughput equation.

$L$  = the highest layer currently subscribed i.e.  $L=0$  is the base layer only.

A client starts by subscribing to the base layer (i.e.,  $L=0$ ). The client estimates  $MTU$ ,  $RTT$ , and  $p$  for a given period using a feedback mechanism such as RTCP. The client estimates the target transmission rate,  $\lambda$ , using the TCP throughput equation.

If  $\lambda$  is less than  $r_0$ , the target transmission rate is less than the transmission rate of the base layer, the base layer is dropped as there is congestion on the network and the client can no longer able receive the base layer.

If  $\lambda$  is greater than  $r_0$ , the algorithm determines the layer,  $L'$ , whose transmission rate is closest to, but not greater than  $\lambda$ . If  $L'$  is greater than  $L$ , then the appropriate layers are added, if  $L'$  is less than  $L$ , then the appropriate layers are dropped.



In a multicast environment the aggregate frequency of join-experiments increases with the number of clients. Failed join-experiments can cause congestion in the network. One solution to overcome this is to coordinate the joining/leaving actions of the clients using synchronization points. This will reduce the frequency and duration of join-experiments, resulting in a lower possibility of congestion.

#### *TCP Emulation At Receivers (TEAR)*

TEAR [57] is a receiver-driven adaptation scheme. The receiver detects congestion using packet losses and timeouts as indicators and determines its own appropriate receiving rate. The receiving rate is determined by emulating the congestion window evolution of a TCP sender and maintains an exponentially weighted moving average (EWMA) of the congestion window. This value is divided by the estimated round trip time to obtain a TCP-friendly sending rate. The receiver asks the sender to transmit data at the estimated TCP-friendly sending rate.

#### *Quality Adaptation by receiver buffering*

Quality adaptation [58] and buffering at the receiver can be used to smooth short-term oscillations caused by AIMD-controlled transmission and playback of hierarchically or layered encoded video. Receiver buffering can sustain momentary drops in the sending rate if enough data is buffered and playing out of its buffer at a higher rate than the server is currently sending. The buffer fills up when the sender is transmitting faster than the receiver is playing out. AIMD is used to react to congestion by adding and dropping layers of the video stream for long-term adaptation. A new layer is added only when the two conditions hold:

$r$  = the current transmission rate.

$n_a$  = number of currently active layers.

$C$  = bandwidth/layer.

$buf_i$  = amount of data buffered for layer  $i$ .

$\alpha$  = rate of linear increase in bandwidth.

$RTT$  = round trip time.

$$r > (n_a + 1)C$$

$$\sum_{i=0}^{n_a-1} buf_i \geq RTT(C(n_a + 1) - 0.5r)^2 / 2\alpha$$

This ensures that a new layer is added only when the instantaneous available bandwidth is greater than the consumption rate of the existing layers plus the new layer, and, there is enough buffered at the receiver to survive an immediate back off and continue playing all

the existing layers plus the new layer. Similarly, layers are dropped when these conditions are not satisfied. That is, when the total amount of buffering falls below the amount required for a drop from a particular rate, then the highest layer is dropped. Buffer space is allocated between layers so as to place a greater importance on lower layers, protecting these layers against a reduction in the transmission rate. The conditions for the adding and dropping of layers and inter-layer buffer allocation for AIMD and hierarchical encoding are described in [59].

### 3.2.2.3. Hybrid Rate Control

Hybrid rate control consists of rate control at both the sender and a receiver. The hybrid rate control is targeted at multicast video and is applicable to both layered video [60] and non-layered video [61]. Typically, clients regulate the receiving rate of video streams by adding or dropping channels while the sender also adjusts the transmission rate of each channel based on feedback information from the receivers. Unlike server-based schemes, the server uses multiple channels, and the rate of each channel may vary due to the hybrid approach of adapting both at the server and receiver.

#### *SCP Model based*

SCP [62] uses a hybrid approach of rate adaptation by combining both rate-based and window-based schemes. It uses window-based rate adaptation schemes like slow start to discover available network bandwidth and multiplicative decrease (MD) to adapt when congestion occurs making it somewhat TCP-friendly. SCP has a number of states:

SlowStart: probing for available bandwidth.

Steady: Available bandwidth fully utilized.

Congested: Network congested as indicated by timeouts or gap in ACK's.

Paused: No packets in the network.

Unlike TCP, when the network is not congested (during steady state,  $W_{ss}$ ) SCP invokes a policy that maintains smooth streaming with maximum throughput by calculating the ACK rate,  $r_{ACK}$ , which is used to adjust the window:

$W_l$  = the congestion window size,

$W_{ss}$  = the threshold of  $W_l$  for switching from slow start to steady state,

$T_{brtt}$  = the base round trip time

$W\Delta$  = the window size incremental coefficient.

$W_{ss} = W_l = r_{ACK} * T_{brtt} + W\Delta$

On congestion, SCP backs off multiplicatively by reducing its congestion window size by half and enters the congestion state. If the network is seriously congested, SCP backs off

exponentially.

$$W_{ss} = W_l = 0.5(W_l)$$

#### *Adaptive Layered Transmission (ALT)*

In ALT [63], the sender monitors loss information for each layer through periodic RTCP receiver reports. The transmission rate of each layer is adapted by the server using the AIMD model. If a client experiences packet loss above a certain loss threshold, it drops a layer to avoid driving the transmission rate of the layer down too low. If the client determines that it has excess capacity, it adds a layer. If all the receivers drop the current highest layer, or if the transmission rate of the highest layer is reduced below the minimum transmission rate, the server may choose to temporarily drop the layer.

#### *Adaptive Multicast of Multi-Layered video (AMML)*

AMML [64] is a rate-based adaptation schemes, as well as a credit-based scheme. In response to receiver feedback, the sender decides the number of layers to encode, and the rate at which to transmit each layer. The network is assumed to provide prioritized service giving the base video layer the highest priority, and successive enhancement layers decreasing priority. AMML is based on congestion control mechanisms used in ATM networks. In the rate-based method, the sender receives feedback explicitly in the form of the desired transmission rate for each layer. The sender initiates the feedback process by multicasting a “forward feedback packet”. At each intermediate node, the ERICA algorithm [65] is used to calculate fair share of link bandwidth of the connection, and this is entered in the explicit rate ( $R_E$ ) field of the forward feedback packet. When the packet reaches the client, the  $R_E$  field indicates the transmission rate the client can support. Using this information, clients send feedback to the server requesting specific transmission rates. Backward feedback packets are merged at intermediate nodes, concatenating the rate fields, and eliminating some if required according to specified criteria, so that the number of requested rates in a feedback packet does not exceed the number of layers the sender can support.

### 3.2.3. Rate Shaping

Rate shaping is a technique to adapt the rate of compressed video bit-streams to the target rate constraint. A rate shaper is an interface (or filter) between the encoder and the transport layer, with which the encoder's output rate can be matched to the available network bandwidth. Rate shaping does not require interaction with the encoder, rate shaping is applicable to any video coding scheme and is applicable to both live and stored video. Rate shaping can be achieved at either the transport layer [66][67][68] or at the compression layer [69].

A filter performs rate shaping and is required for the server-based rate control. This is because the stored video may be pre-compressed at a certain rate, which may not match the available bandwidth in the network. There are a number of filters that can be used to achieve rate shaping.

- **Codec filter:** decompresses and compresses a video stream. It is commonly used to perform transcoding between different compression schemes. Depending on the compression scheme used, transcoding could be simplified without full decompression and recompression.
- **Frame-dropping filter:** distinguishes between the frame types (e.g., I-, P-, and B-frame in MPEG) and drop frames according to importance. For example, the dropping order would be first B-frames, then P-frames, and finally I-frames. The frame-dropping filter is used to reduce the data rate of a video stream by discarding a number of frames and transmitting the remaining frames at a lower rate. The frame-dropping filter could be used at the server [70] or used in the network.
- **Layer-dropping filter:** distinguishes between the layers and drop layers according to importance. The dropping order is from the highest enhancement layer down to the base layer.
- **Frequency filter:** performs filtering operations on the compression layer.
  - **Low-pass filtering:** discards the DCT coefficients of the higher frequencies.
  - **Color reduction filter:** performs the same operation as a low-pass filter except that it only operates on the chrominance information in the video stream.
  - **Color-to-monochrome filter:** removes all color information from the video stream. In MPEG, this replaces each chrominance block with an empty block.
- **Re-quantization filter:** performs operations on the compression layer (i.e., DCT coefficients) [71]. The filter first extracts the DCT coefficients from the compressed video stream through techniques such as de-quantization then re-quantizes the DCT coefficients with a larger quantization step, resulting in rate reduction.

*Selective Frame Discard*

Selective frame discard [72] preemptively drops frames at the server in an intelligent manner by considering available network bandwidth and client QoS requirements. The selective frame discard has two major advantages. The first is that by taking the network bandwidth and client buffer constraints into account, the server can make the best use of network resources by selectively discarding frames in order to minimize the likelihood of future frames being discarded, thereby increasing the overall quality of the video delivered. The second is that unlike frame dropping in the network or at the client, the server can also take advantage of application-specific information such as regions of interest and group of pictures (GOP) structure, in its decision in discarding frames.

*Dynamic Rate Shaping (DRS)*

DRS [73] is based on the rate-distortion theory in which it selectively discards the Discrete Cosine Transform (DCT) coefficients for high frequencies so that the target rate can be achieved. The HVS is less sensitive to higher frequencies, and so DRS selects the highest frequencies and discards the DCT coefficients of these frequencies until the target rate is met.

*IVS Rate Shaping*

The IVS H.261 video coder [74] achieves rate shaping by changing one or more of the following:

- Refresh rate: Refresh rate is the rate of I-frames, which are encoded by the video encoder. Decreasing the refresh rate can reduce the output rate of the encoder, but will reduce quality.
- Quantizer: This specifies the number of DCT coefficients that are encoded. Increasing the quantizer decreases the number of encoded coefficients and the image is coarser.
- Movement detection threshold: For inter-frame coding (P and B frames), the DCT is applied to signal differences. The movement detection threshold limits the number of blocks, which are detected to be sufficiently different from the previous frames. Increasing this threshold decreases the output rate of the encoder and results in reduced video quality.

Two modes are used for controlling the rate of encoder. In Privilege Quality mode (PQ mode), only the refresh rate is changed. In Privilege Rate mode (PR mode), only the quantizer and movement detection threshold are changed. PQ mode control results in higher frame rates, but with lower SNR (signal-to-noise ratio) than PR mode. The receiver periodically sends packet loss feedback. The encoder is dynamically adapted:

*If (median loss > tolerable loss)*  
     {  $Max\_rate = Max(Max\_rate/GAIN, Min\_rate)$  }  
*else*  
     {  $Max\_rate = Max(Max\_rate + INC, Min\_rate)$  }

Two dimensional scaling changes both the frame rate and the bit rate based on feedback [75]. Other rate shaping mechanisms use similar methods but differ in how the rate shaping is achieved for example using block dropping [76] and frame dropping [77].

### 3.2.4. Rate Adaptive Encoding

The objective of a rate-adaptive encoding algorithm is to adapt the encoding configuration and thus the output bitrate in response to network conditions. Adaptive encoding can be achieved by the alteration of the encoder's configuration parameters such as the quantization parameter (QP) and/or the alteration of the video frame rate. Traditional video encoders (e.g., H.261 [78], MPEG-1 [79], MPEG-2 [80]) typically rely on altering the QP of the encoder to achieve rate adaptation but cannot change the frame rate. MPEG-4 [81] and H.263 [82] coding schemes are suitable for very low bit-rate video applications since they allow for temporal scalability i.e. dynamic frame rate, achieved by frame skipping.

Often a scalable video coding scheme compresses a video sequence into multiple layers. Base layer, can be independently decoded and provide coarse visual quality; enhancement layers, can only be decoded together with the base layer and provide better visual quality. The complete bitstream (i.e., all layers) provides the highest quality. Layers can be transmitted in the same channel (by interleaving the base layer with the enhancement layers) or in separate channels. Scalable coding can be used to achieve rate control during network congestion and can provide a range of picture quality suited to heterogeneous requirements of receivers in a multicast scenario.

Scalability of video consists of:

- SNR scalability: SNR scalable coding quantizes (Q) the DCT coefficients to different levels of accuracy by using different quantization parameters (QP), which are then coded by variable length coding (VLC). The resulting streams have different SNR levels. The smaller the quantization parameter is, the better the quality the video stream (Figure 3.5).
- Spatial scalability: Spatially scalable video is encoded by using spatially up-sampled pictures from a lower layer as a prediction in a higher layer (Figure 3.6).

- Temporal scalability: Temporally scalable video is encoded by making use of temporally up-sampled pictures from a lower layer as a prediction in a higher layer. Temporally scalable codec uses temporal down-sampling and temporal up-sampling. Temporal down-sampling uses frame skipping/dropping. Temporal up-sampling uses frame copying.

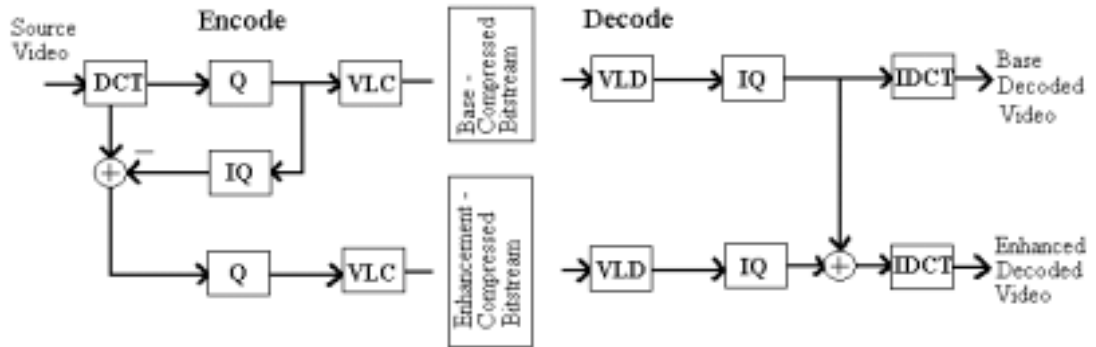


FIGURE 3.5: 2-LEVEL SNR SCALABLE ENCODER

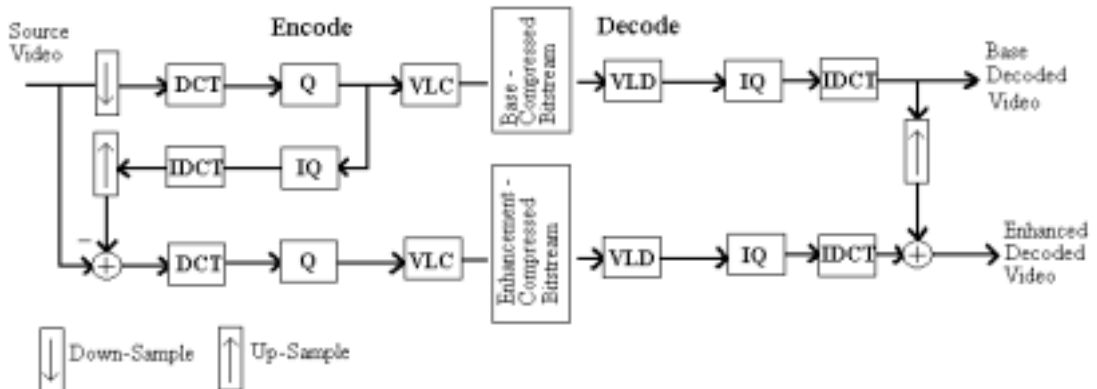


FIGURE 3.6: 2-LEVEL SPATIAL SCALABLE ENCODER

### *Fine granularity scalability (FGS)*

FGS was proposed to MPEG-4 in the Amendment on Streaming Video Profile of the MPEG-4 standard [83] [84], [85], [86]. An FGS encoder compresses a raw video sequence into two substreams, i.e., a base layer bit-stream and one or two enhancement bit streams. Different from an SNR scalable encoder, an FGS encoder uses bit plane coding to represent the enhancement stream making it capable of achieving continuous rate control. This is because the enhancement bit-stream can be truncated anywhere to achieve the target bit-rate. In this way, FGS [87] can accommodate a wide range of data rate variability by distributing enhancement layers over a wide range of bit rates, it provides efficient coding based on bit plane coding which is more efficient than run-level coding, and it separates the FGS layer from the motion compensation to eliminate drift in the enhancement layers.

The MPEG-4 FGS framework [88][89] consists of a base layer and one or two enhancement layers. The base layer is generated by DCT, ME/MC (Motion Estimation and Motion Compensation), and entropy coding. To provide consistent quality, the QP is fixed, however, this leads to data rate variability. There are two types of enhancement layers defined for hybrid temporal-SNR scalability:

- SNR-FGS (FGS) layer enhances video quality by adding DCT coefficients with a reduced quantization step size leading to highly accurate DCT coefficients and high quality video. Frequency weighting and selective enhancement shifting factors (block bit plane shift) is applied before encoding the bit planes so that certain macroblocks in a VOP or the most relevant DCT coefficients can be given a finer QP in the enhancement layer.
- Temporal FGS (FGST) layer improves temporal resolution by providing a higher frame rate and smooth motion.

There are two possible implementations of a hybrid temporal- SNR scalability structure. The FGS-FGST first enhances quality with the FGS layer to improve spatial resolution of the video quality and then the second enhancement layer improves quality with the FGST layer (Figure 3.7). The FGST-FGS enhances the temporal resolution first and then the spatial resolution (Figure 3.8).

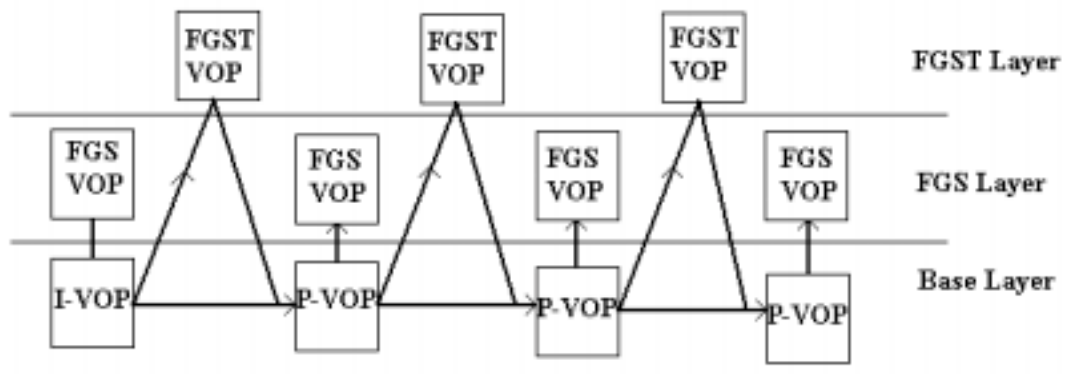


FIGURE 3.7: FGS-FGST IMPLEMENTATION



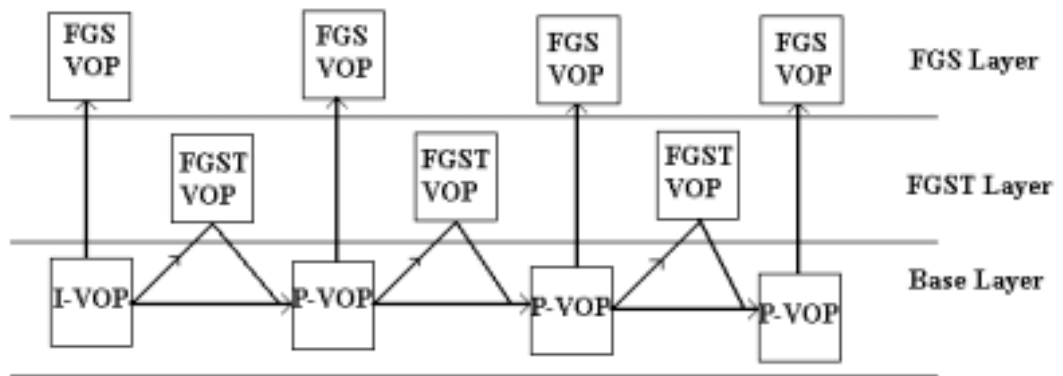


FIGURE 3.8: FGST-FGS IMPLEMENTATION

### *Progressive Fine Granularity Scalability (PFGS)*

A variation of FGS is PFGS [90]. PFGS is similar to FGS achieving fine granular bit-rate scalability and error resilience. But unlike FGS, which only has two layers, PFGS could have more than two layers. The essential difference between FGS and PFGS is that FGS only uses the base layer as a reference for motion prediction while PFGS uses multiple layers as references to reduce the prediction error, resulting in higher coding efficiency.

### *Object Adaptation*

MPEG-4 is the first international standard addressing the coding of video objects (VO's) [91]. In MPEG-4, a frame of a video object is called a video object plane (VOP), which is encoded separately. Access to video objects provides greater flexibility to perform adaptive encoding. In particular, the target bit-rate can be distributed among video objects, in addition to the alteration of QP on each VOP [92].

### *Rate-Distortion*

For all video coding algorithms, the main problem is how to determine an optimal value for QP to achieve the target bit-rate. In the rate-distortion (R-D) theory, there are two approaches for encoding rate control. The model-based approach assumes DCT input distributions and quantizer characteristics [93] to obtain closed-form solutions using continuous optimization theory. Whilst the operational R-D method considers practical coding environments to determine a finite set of possible QP [94], [95], [96], [97]. This set of QP is used by the rate control algorithm to determine the optimal strategy to minimize the distortion under the constraint of a given target bit rate [98].

*Wavelet-based scalable encoding*

In wavelet compression, the image is divided into various sub-bands with increasing resolutions. Image data in each sub-band is transformed using a wavelet function to obtain transformed coefficients. The transformed coefficients are then quantized and run length encoded before transmission. Wavelet compression overcomes the blocking effects of DCT based methods since the entire image is used in encoding instead of blocks. A wavelet encoder can benefit from network feedback such as available capacity to achieve this scalability [99]. Wavelet compression results in progressively encoded video by coupling wavelet transforms with encoding techniques resulting in a continuous rate scalability, where the video can be encoded at any desired rate within the scalable range [100]. Two common approaches for wavelet compression are to use a Motion-Compensated (MC) 2-dimensional (2-D) wavelet function [101] or a 3-D wavelet [102].

**3.2.5. Transcoder-based adaptation**

Adaptation can also occur within the network using video gateways placed at appropriate locations in the network to convert through transcoding a high bandwidth transmission into a transmission with appropriate bandwidth to accommodate groups of poorly connected receivers [103]. Additionally, receivers can adapt to network congestion by dynamically switching to a gateway with a more appropriate service for their requirements. Or the gateway can use feedback directly from the clients to use an adaptive rate-control algorithm to adjust its transmission. The main difficulties with transcoder-based adaptation schemes are the design of the transcoding algorithm, and the placement or selection of the gateway to perform the transcoding. The input format is converted into an intermediate representation by a decoder and is transformed, and delivered to the encoder, which produces a new bit stream in a new format.

Instead of a single transformation, multiple transforms can be performed, allowing for greater flexibility in choosing an encoder/decoder pair, and optimizing performance by enabling a higher level of intermediate representation (such as DCT coefficients) to be used instead of decomposing the input stream into pixel format to achieve a target output rate, including temporal and spatial decimation and/or frame geometry conversion (Figure 3.9).

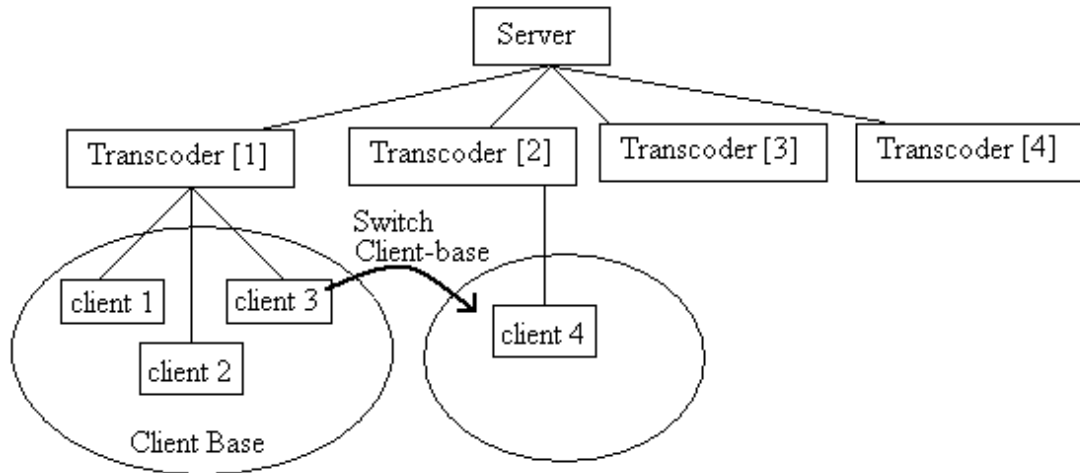


FIGURE 3.9: MULTIPLE TRANSCODER SYSTEM

Configuring transcoders requires an external control interface through which parameters such as codec parameters, target bitrate, etc. can be specified. A more flexible scheme for configuration and control of transcoders has been suggested in [104][105]. However, work has been done to devise a control scheme that automatically configures transcoders within the multicast tree to support branches with bad reception [106] creating a hierarchical client-base structure. A group of clients affected by a bottleneck tries to locate a client higher up in the hierarchy with better reception to provide a customized, transcoded version of the session stream by multicasting request messages. To prevent an implosion of requests from multiple clients in the group, a client delays its request by an interval proportional to its distance from the stream source plus a small random interval. If the client receives an identical request during this delay, it cancels its own request.

### 3.2.6. Error Control

The objective of congestion control is to prevent or minimize packet loss. However, packet loss and/or serious packet delays is sometimes unavoidable. Error control techniques can be used to maximize video quality in presence of packet loss. These techniques can be classified into four types:

- **Forward Error Correction (FEC):** The idea of FEC is to add extra (redundant) information to a compressed video bit-stream so that the original video can be reconstructed in presence of packet loss. FEC incurs only a small transmission delay [107] but can be ineffective when bursty packet loss occurs exceeding the recovery capability of the FEC codes.
- **Retransmission:** Retransmission-based schemes such as automatic repeat request (ARQ) are generally not used for streamed multimedia applications, as the delay

requirements cannot be satisfied. However, if the one-way trip time is short relative to the maximum allowable delay, a retransmission-based approach (or delay-constrained retransmission) is possible [108].

- **Error-resilience:** Error-resilient schemes operate at the compression layer to limit the potential corruption that may be caused when packets are lost. MPEG-4 has several such error-resilience schemes and these are discussed in Volume 2 Appendix B - MPEG-4.
- **Error concealment:** Error concealment is a post-processing technique used by the decoder. When bit errors occur, the decoder uses error concealment to hide/mask the loss from the viewer.

### *FEC*

FEC adds extra (redundant) information to a compressed video bit-stream so that the original video can be reconstructed in presence of packet loss. Based on the type of redundant information added, FEC schemes can be classified into three categories:

1. Channel coding.
2. Source coding-based FEC.
3. Joint source/channel coding.

### *Channel Coding*

Channel coding is usually used in block codes. A video stream is segmented, of which each segment is packetised into  $k$  packets and a block code is applied to the  $k$  packets. The channel encoder places the  $k$  packets into a group, creating additional packets from them. The total number of packets in the group becomes  $n$ , where ( $n > k$ ) If a client receives  $K$  packets, in order to recover a segment, a client must receive  $K$  packets in the  $n$ -packet block, where  $K$  not less than  $k$ , the original number of packets in the segment. The client only needs to receive any  $k$  packets in the  $n$ -packet block so that it can reconstruct all the original  $k$  packets (Figure 3.10).

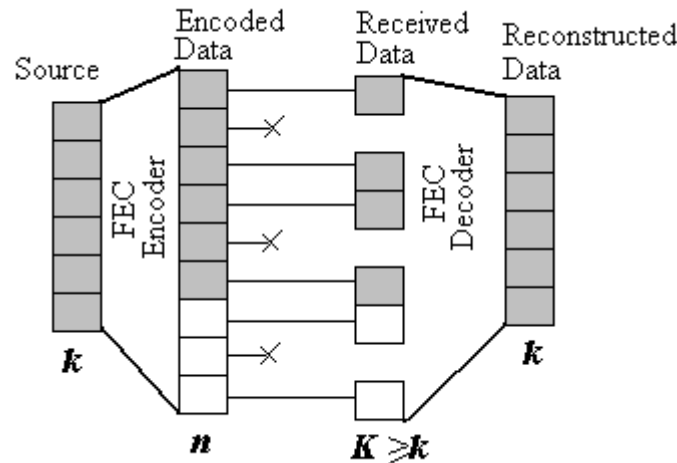


FIGURE 3.10: CHANNEL CODING

The ability to recover from any  $k$  out of  $n$  packets regardless of which packets are lost, allows the network and receivers to discard some of the packets which cannot be handled due to limited bandwidth or processing power. However, there are also some disadvantages associated with channel coding.

1. It increases the transmission rate by a factor of  $n/k$  if  $(n - k)$  redundant packets are added to every  $k$  original packets. The higher the loss rate is, the higher the transmission rate is required to recover from the loss. The higher the transmission rate is, the more congested the network gets, which leads to an even higher loss rate.
2. It increases delay, as a channel encoder must wait for all  $k$  packets in a segment before it can generate the  $(n - k)$  redundant packets. Similarly, the client has to wait for at least  $k$  packets of a block to arrive before it can decode and playback the video segment.
3. It is not adaptive to varying loss characteristics and it works best only when the packet loss rate is stable. To improve the adaptive capability of channel coding, feedback can be used, that is, the client conveys the loss characteristics to the source, the channel encoder can adapt the redundancy accordingly.

**Equal error protection (EEP):** all the bits of the compressed video stream are treated equally, and given an equal amount of redundancy. However, the compressed video stream typically does not consist of bits of equal significance.

**Unequal error protection (UEP):** the more significant information bits are given more protection [109], for example, Priority Encoding Transmission (PET) [110]. This scheme allows a client to set different levels (priorities) of error protection for different segments of the video stream making it more efficient (i.e. with less redundancy) and suitable for transporting video, which has frame hierarchy of priority (i.e., I-, P-, and B-frames).

**Hierarchical FEC (HFEC)** [111]: is a receiver-driven hierarchical layered FEC scheme where additional streams with only FEC redundant information are generated along with the video layers, i.e. in addition to hierarchically encoded video layers, FEC layers are produced. Each of the FEC layers is used to recover a different video layer, and each of the FEC layers is sent to a different multicast group. Subscribing to more FEC groups corresponds to higher level of protection.

### *Joint Source/Channel Coding*

Source coding focuses on developing efficient source coding techniques whilst channel coding focuses developing robust channel coding techniques [112]. Joint channel/source coding is the combination of both techniques [113]. Channel/source encoding tries to find the optimal source-encoding rate  $R_o$  that achieves the minimum distortion  $D_{MIN}$ . There is a trade off effect here as the lower the source-encoding rate  $R'$  is, the larger its distortion  $D$ . However, if the total rate,  $R_T$ , is fixed (i.e., the source-encoding rate  $R$  plus the channel-coding redundancy rate  $R'$ ), the higher  $R$  is, the lower  $R'$  must be, leading to a higher probability  $P_c$  of the packet will be corrupted. So, if  $R$  is increased, redundancy  $R'$  is decreased, and the probability of corruption increases and so does distortion,  $D$ .

When  $R' \searrow$  then  $D \nearrow$

$$R_T = (R + R')$$

If ( $R_T = \text{constant}$ )

$$R \nearrow \text{ then } R' \searrow \text{ then } P_c \nearrow \text{ then } D \nearrow$$

Using feedback information, the joint source/channel optimizer finds the optimal rate allocation between the source coding and the channel coding for a given loss characteristic and conveys the optimal rate to both the source encoder and the channel encoder. Then the source encoder chooses an appropriate QP to achieve its target rate and the channel encoder chooses a suitable channel code to match the channel loss characteristic.

### *Delay-constrained Retransmission - ARQ*

When packets are lost, the client sends feedback to notify the server that the packet was lost. The server simply retransmits the lost packets. This method is unsuitable for real-time video since a retransmitted packet arrives at least  $1.5 RTT$ 's after the transmission of the original packet, which might exceed the delay required by the application. However, if the one-way trip time is short enough relative to the maximum allowable delay, a retransmission is possible [114], [115]. Retransmissions can be classified by who initiates the retransmission request, i.e. sender, receiver or hybrid. In receiver-based control, when the receiver detects a lost packet  $N$ , it checks if the current time plus the estimated  $RTT$  and a slack term,  $D_s$ , is

less than the packets playout time. Then, the client sends the request for retransmission of packet  $N$  to the server. The slack term,  $D_s$ , could include flexibility or error tolerance in the estimation of the RTT. In sender-based control, the sender receives a request for retransmission of packet  $N$ , this time the sender determines whether the packet will arrive on time by adding the current time, the estimated one-way delay and a slack term. The sender can either receive as part of the retransmission request the playout time of the lost packet or else determine it from the encoder. Hybrid control is a combination of the sender-based control and the receiver-based control. The receiver makes decisions on whether to send retransmission requests while the sender makes decisions on whether to disregard requests for retransmission. In the multicast case, retransmission has to be restricted to avoid an implosion of retransmission requests at the sender.

#### *Error-resilience and Concealment*

Error-resilience attempts to prevent errors propagating in the decoding of the bitstream. The standardized error-resilient tools include resynchronization marking, data partitioning, and data recovery (e.g., reversible variable length codes (RVLC)) [116], [117]. These are targeted at error-prone environments like wireless channels. On the other hand, when packet loss is detected, the receiver can employ error concealment to mask the lost data. There are two basic approaches for error concealment [118], spatial and temporal interpolation. In spatial interpolation, missing pixel values are reconstructed using neighboring spatial information, whereas in temporal interpolation, the lost data is reconstructed from data in the previous frames. Both of these are discussed with particular reference to MPEG-4 in Volume 2 Appendix B - MPEG-4.

#### **3.2.7. Critique of Adaptation Techniques**

In all of the adaptation algorithms discussed, there is no definition of quality. The basic question is what is quality in terms of the multimedia content being streamed?

- **Dependency on algorithm control parameters.**

Several algorithms have a strong dependency on the choice of control parameters used within the algorithm. These parameters have an inexplicable origin and have been empirically optimised. Consider, rate control schemes with an AIMD-like behaviour whereby the transmission rate is increased by some additive increase factor,  $\alpha$ , and reduced multiplicatively by a decrease constant,  $\beta$ . The main disadvantage of AIMD algorithms is that they have more abrupt changes and oscillations in sending rate as the operation of these algorithms relies heavily on the values chosen for  $\alpha$  and  $\beta$ . If  $\alpha$  is chosen to be too large,

then the increasing the transmission rate could push the system into causing congestion. This in turn causes the client to experience loss to which the server multiplicatively reduces its transmission rate. Thus, the system operation is heavily dependent on the value for  $\alpha$ . If  $\alpha$  is too small, the server is very slow to make use of the extra capacity. One solution would be to use a mechanism to use dynamic values for  $\alpha$  and  $\beta$ . For example,  $\alpha$  could initially be a small value and increase exponentially maximizing the capacity faster. Similarly,  $\beta$  dramatically reduces the transmission rate upon congestion, which can significantly degrade the perceived quality.  $\beta$  should be reduced to a value, which reflects the estimated bottleneck bandwidth of the client. The use of dynamic values is the approach taken in LDA.

- **TCP Throughput model and estimation of network state parameters such as RTT, loss etc.**

Algorithms using the TCP throughput models have limited value, as they do not define which version of TCP they are modelling. They are not effective for high loss rates over 16% and are constrained by the accuracy of the estimation of *RTT* and other parameters. Further, the TCP throughput equation is based only on a single TCP connection with a steady-loss ratio.

- **Over-reaction based on immediate feedback.**

If there are short bursty losses as seen in wireless networks, the system reacts, however, the question is, should the server reduce the transmission rate in response to a short bursty loss. It would seem more sensible for the server to maintain a history of the clients' feedback and monitor both in terms of the clients' long-term and short-term experience. For example, a client is receiving 64kbps, but periodically there is a short bursty loss (e.g. caused by some object in the environment periodically sending signals which interfere with a clients reception), overall the clients receiving bit rate is 64kbps, if the clients transmission rate is reduced, this will not eliminate the short bursty losses. This was addressed in DAA, which uses an EWMA to avoid reactions to sudden bursty losses and in TFRC, which uses loss events consisting of multiple consecutive losses as an indicator of congestion. However, most algorithms react only on immediate feedback.

- **Responsiveness of the algorithm.**

The converse argument of reacting based on both short-term and long-term feedback is that when there are sudden jumps of congestion sustained over a longer period of time, the algorithms are slower to react and recover from the congestion. For RTCP based feedback, the minimum interval between feedback messages is 5 seconds. The server generally only



reacts upon receipt of a new feedback message. Is this adaptation interval short or long enough to adapt quickly and effectively to changing network conditions? What happens if this RTCP packet is lost, then adaptation at a potentially crucial moment will only occur at a minimum 10-second interval. Lost feedback messages should be eliminated when the rate control algorithm requires it for adaptation, thus hybrid transport layer protocols should be used, for example, RTP/UDP/IP and RTCP/TCP/IP.

- **ACK based schemes.**

Typically multimedia is streamed over UDP with the main argument for this being that ACKs and retransmissions have little benefit. Often, the lost packet cannot be retransmitted, as the playout time is less than the RTT required to retransmit the lost packet. Thus, ACK based schemes, such as TFRC, can cause an unnecessary load on the network.

- **Transcoder Schemes**

Transcoder schemes require some sort of interaction with the network. The main problem is the distribution of the transcoder. Even if this was optimised, transcoders then need some sort of decode-encode policy, which makes them susceptible to the arguments of the adaptive encoding techniques.

- **Network Schemes**

Prioritised transmission of streams using IP header fields such as ToS, assumes a homogenous network, i.e. where the same policy, interpretation and treatment of the packets is applied throughout the network. This is not the case as the policies applied within various subnets of the network often differ.

- **Retransmission**

Retransmission-based schemes are generally not used for real-time video, as the delay requirements cannot be satisfied.

- **FEC**

There are several disadvantages associated with FEC techniques. The transmission rate is increased by a factor of  $n/k$  if  $(n - k)$  redundant packets are added to every  $k$  original packets. The higher the loss rate is, the higher the transmission rate is required to recover from the loss. The higher the transmission rate is, the more congested the network gets, which leads to an even higher loss rate. It increases delay, as a channel encoder must wait for all  $k$  packets in a segment before it can generate the  $(n - k)$  redundant packets. Similarly, the client has to wait for at least  $k$  packets of a block to arrive before it can decode and

playback the video segment. It is not adaptive to varying loss characteristics and it works best only when the packet loss rate is stable. To improve the adaptive capability of channel coding, feedback can be used, that is, the client conveys the loss characteristics to the source, and the channel encoder can adapt the redundancy accordingly.

○ **Translation of rate into real video encoding parameters**

There is a major gap in the rate control algorithms presented. Consider a simple rate control algorithm, a server is delivering video at 50kbps, and based on feedback, the algorithm indicates that the transmission rate can be increased to 55kps. There are two main questions, how should the extra 5kps be achieved, should the frame rate be increased, or the QP increased etc.? Even if this was known, how is it achieved, does it require interaction with the encoder or can the server apply some parsing of the bitstream? So for example, if the server can increase the frame rate, what happens if the increased frame rate exceeds the new transmission rate as suggested by the algorithm i.e. 55kbps, should the server aim as close to the new transmission rate? This is not mentioned in the algorithms discussed.

○ **Definition of a layer.**

In the layered algorithms, there is no definition of what constitutes a base layer and its enhancement layers. Further there is an inter-layer dependency; for example, the base layer needs to be correctly decoded before subsequent enhancement layers can be decoded. However, if the base layer is not correctly decoded, the enhancement layers cannot be decoded. Even if layers are correctly decoded, as they may have different paths and different round-trip times, the receivers need to resynchronise the arriving data, which can further complicate the playout of the stream. The inter-layer dependency problem is addressed in DSG where each layer is self-contained and can be decoded independently.

○ **Codec constraints.**

Adaptive encoding techniques are constrained by the adaptability of the encoder being used. Not all codecs support a dynamic QP, spatial filtering or temporal filtering. Adaptive encoding is only really effective for live transmissions; otherwise a pre-compressed stream must be decompressed and then re-compressed on the fly, so that the adaptive encoding policy can be applied. This may cause a serious processing overload and delay on the server. This method is not really suitable for unicast scenarios where there may be hundreds of users all connected to the one server requesting different content. If applied to multicast scenarios, the flexibility of adaptation is traded off with the heterogeneity of the receivers.

- **Object, Spatial and Temporal Scalability relating to content.**

Even if a codec supports spatial and/or temporal scalability, the actual scalability policy should relate to the content of the video stream. For example, a client is watching a football clip, temporal scalability is undesirable as the user has a preference for motion continuity and can tolerate degradation in the spatial scalability. That is, if there are frequent scene changes and high motion in the clip, due to the high motion content, the user has less time to notice spatial detail and so spatial detail should be scaled more relative to the temporal detail. This issue is not addressed in adaptive encoding algorithms. In MPEG-4 there is a lot of work on object-content scalability. Object extraction and segmentation is very difficult to do efficiently on the fly and requires heavy-duty processing. Object scalability is highly subjective, for example, what one user may consider a prioritised object, another may not. Objects have a dynamic importance throughout the clip, for example, in a football clip, for the most of the clip, the goalkeeper may not be very interesting and important, but if there is a penalty shoot out, the goalkeeper will become a very important object.

- **User Perception**

None of the algorithms consider user-perception as a factor in their decision making process. The user is the primary entity affected by adaptation and therefore should be given priority in the adaptation decision-making process. For example, once again, if a clip is being streamed at a particular encoding configuration and the system needs to degrade the quality being delivered, how this adaptation occurs should be dictated by the users' perception. The way to degrade should be such to have the least negative impact on the users' perception.

In conclusion, the objective of adaptation is to prevent or minimize packet loss. However, there needs to be some sort of understanding of quality in order for adaptation to occur in an achievable and intelligent manner. The primary question, which must be answered, is what is video quality and how should it be video quality adapted in response to network congestion?

### 3.3. Subjective Test Methodologies

There are a number of different test methodologies. The choice of methodology should reflect the goals of the experiment. The main testing methodologies are [119]:

- Absolute Category Rating Method (ACR)
- Degraded Category Rating Method (DCR)
- Pair Comparison Method (PC)

Others include:

- Forced Choice
- Multiple Stimulus Hidden Reference and Anchors (MUSHRA)

#### 3.3.1. Absolute Category Rating Method (ACR)

The Absolute Category Rating method is a category judgment where the test sequences are presented one at a time and are rated independently on a category scale (Figure 3.11). (This method is also called Single Stimulus Method). The method specifies that after each presentation the subjects are asked to evaluate the quality of the sequence shown using a five-point quality grading scale.

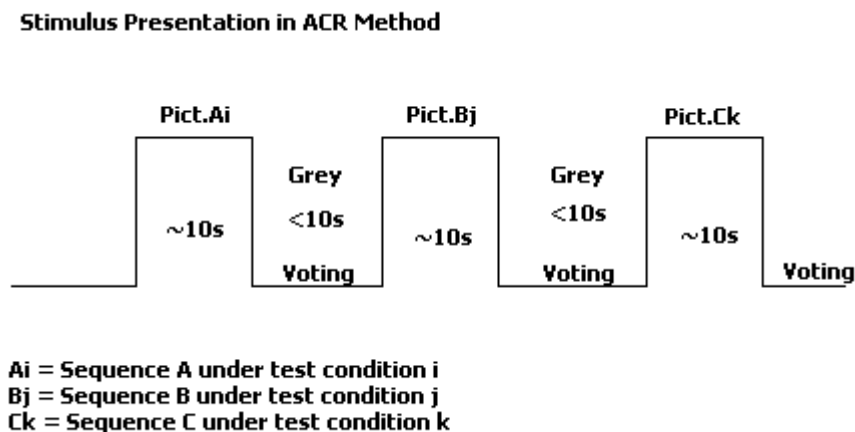


FIGURE 3.11: ACR METHODOLOGY

#### 3.3.2. Degraded Category Rating Method (DCR)

The Degradation Category Rating implies that the test sequences are presented in pairs: the first stimulus presented in each pair is always the source reference, while the second stimulus is the same source presented through one of the systems under test (Figure 3.12).

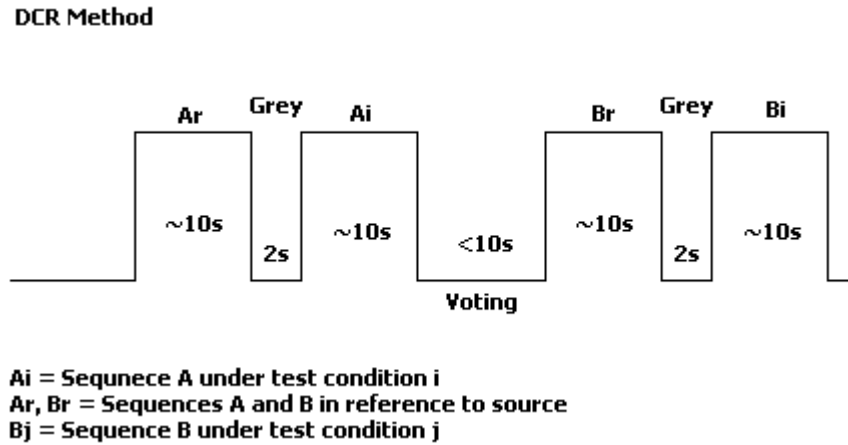


FIGURE 3.12: DCR METHODOLOGY

In this case the subjects are asked to rate the impairment of the second stimulus in relation to the reference using the five-point impairment scale. The necessary number of replications is obtained for the DCR method by repeating the same test conditions at different points of time in the test.

### 3.3.3. Pair Comparison Method (PC)

The method of Pair Comparisons implies that the test sequences are presented in pairs, consisting of the same sequence being presented first through one system under test and then through another system. The systems under tests (A, B, C, etc.) are generally combined in all the possible  $n(n-1)$  combinations AB, BA, CA, etc. Thus, all the pairs of sequences should be displayed in both the possible orders (e.g. AB, BA). After each pair a judgment is made on which element in a pair is preferred in the context of the test scenario. For the PC method, the number of replications need not generally be considered, because the method itself implies repeated presentation of the same conditions, although in different pairs (Figure 3.13).

**PC Method**

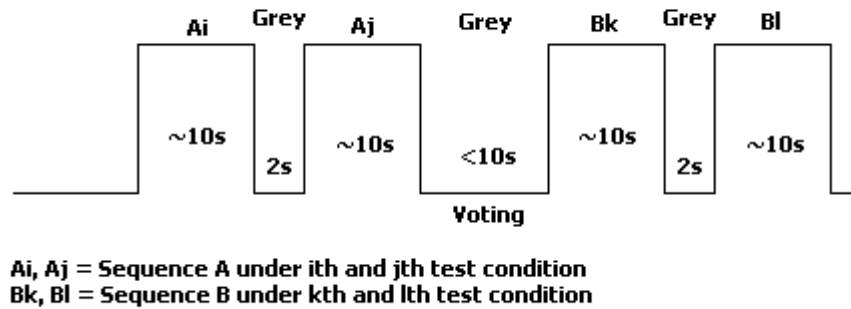


FIGURE 3.13: PAIR COMPARISON METHODOLOGY

**3.3.4. Forced Choice Methodology**

The forced choice methodology is often employed in cognitive science, and PC is one of its applications. In forced choice, the subject is presented with a pair of alternatives separated by a short gap or signal. The subject must choose one of the alternatives according to some test criteria. (Figure 3.14).

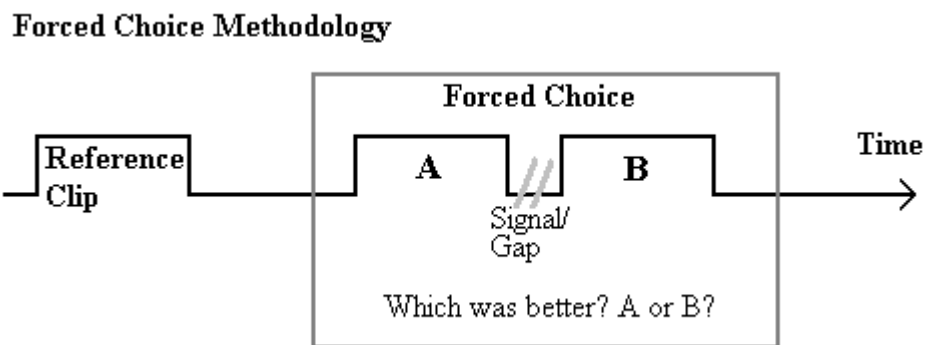


FIGURE 3.14: FORCED CHOICE METHODOLOGY

At the beginning of the test procedure, the reference clip is shown. During a single trial, the subject is shown two degraded versions of the same clip, 'A' and 'B'. A degraded version of the clip is one with a lower encoding configuration. These clips are shown consecutively separated by a short signal or gap. The subjects' task is to choose whether the first or second clip was better. It is not a forced choice when the reference clip is presented along with a degraded clip, as the two clips are not equal. In forced choice, there are equally probable alternative degraded versions of the clip between which the subject must choose. When a subject cannot make a decision, they are forced to make a choice. The bias is binary, which simplifies the grading scale or rating procedure allowing for reliability, verification and validation of the results.

### 3.3.5. Multiple Stimulus Hidden Reference and Anchors (MUSHRA)

The MUSHRA methodology [120, 121] under development for subjective testing of video by the EBU (European Broadcasting Union) and has been used as a subjective test methodology for audio in MPEG-4 Audio Version 2 verification tests [122]. The subject is presented with all different processed versions of the test item at the same time. The subject is not expected to observe all versions simultaneously, but they are available. This allows the subject to observe one version of the test material, quickly followed by another, facilitating the subject to come to a decision about the relative quality of the different versions. In this methodology expert quality assessors are required as they can provide consistent results more quickly. Of the multiple stimuli, one is the known reference, one is a hidden reference and other stimuli must include hidden anchors with clearly defined parameters and targeted at gaining low marks. The purpose of the hidden reference and anchors is to ensure that the full range of the grading scale is used and in this fashion can be used to calibrate the subjects grading scale. The subject assesses the quality of all stimuli, according to the five-interval continuous quality scale (CQS). For example, a subject may consider the known reference to be only ‘good’ and the most degraded version as being ‘fair’. All grades assessed by this subject should fall between ‘fair’ and ‘good’.

### 3.3.6. Grading Scales

When Single Stimulus methods are used, the quality can be rated by the subject using either the quality scale (Table 3.1) or the impairment scale (Table 3.2) [123].

Quality Scale	Rating
Excellent	5
Good	4
Fair	3
Poor	2
Bad	1

TABLE 3.1: FIVE POINT QUALITY GRADING SCALE

<b>Impairment Scale</b>	<b>Rating</b>
Imperceptible	5
Perceptible but not annoying	4
Slightly annoying	3
Annoying	2
Very annoying	1

TABLE 3.2: FIVE POINT IMPAIRMENT SCALE

To reduce the bias due to interpretation of the labels as in Table 1, the Continuous Quality Scale (CQS) is used [124]. CQS consists of identical graphical scales (typically 10 cm long or more). The scale is divided into five equal intervals marked with their descriptor labels.

If the systems which are assessed in a test are judged as being rather equal in overall quality and therefore get very similar scores, it may be advantageous to rate additional quality components on separate scales for each condition. In this way it is possible to receive information on specific characteristics where the test objects are perceived as significantly different, even if the overall quality is practically the same. Results from such additional tests can give valuable information of the systems under test. The Double Stimulus Continuous Quality Scale (DSCQS), uses both the five-point quality grading scale and the impairment scale.



### 3.4. Objective Metrics

From the literature there are many objective metrics that can be used. The most common of which is the Peak Signal to Noise Ratio (PSNR), and has been widely used in many applications and adaptation algorithms to assess video quality. The advantage of PSNR is that it is very easy to compute. However, PSNR does not match well to the characteristics of human visual system [125]. The main goal of objective metrics is to measure the perceived quality of a given image or video. The properties of the HVS determine the visibility of distortions and thus the perceived quality. The fundamental challenge of objective metrics is to understand what quality means to the viewer. There are many factors that affect how viewers perceive quality, such as:

- Content
- Viewing distance
- Display size
- Resolution
- Brightness
- Contrast
- Sharpness/fidelity
- Colour
- Natural content vs. synthetic.

There are different ways these metrics can be applied to measure quality, namely, in the presence of the reference video clip and without the reference clip. A single-ended method is used when there is no reference video clip available whereas a double-ended method uses the reference clip and makes measurements comparing the reference and the degraded clip. The double-ended method can be applied in two ways, using the full reference clip in quality measurement or using only features extracted from the reference clip in the quality measurement (Figure 3.15).

- **Full Reference:** This is a double-ended method using both the reference and degraded clips. The comparison between reference and degraded clips uses a spatial and temporal alignment process to compensate for any vertical or horizontal picture shift or cropping, and correction for any offsets or gain differences in the luminance and chrominance channels. The objective quality metric is then calculated by applying a human vision perceptual model. It is generally accepted that the double-ended method using full reference video information provides the best accuracy for objective quality rating.
- **Reduced Reference:** This is a variation on the double-ended method using

measurement systems at points A and B. Specific visual features are extracted from both the reference using system A and the degraded clips using system B. Extracted features can include blockiness, spatial and temporal signal information and noise. The features from system A and B are then compared and applied to a human vision perceptual model to give an indication of video quality.

- **No Reference:** This is a single-ended method using on the degraded clip for quality measurement. The lack of a reference clip means that the quality measurement may be subject to errors caused by picture content resembling the specific impairment parameters that are being detected.

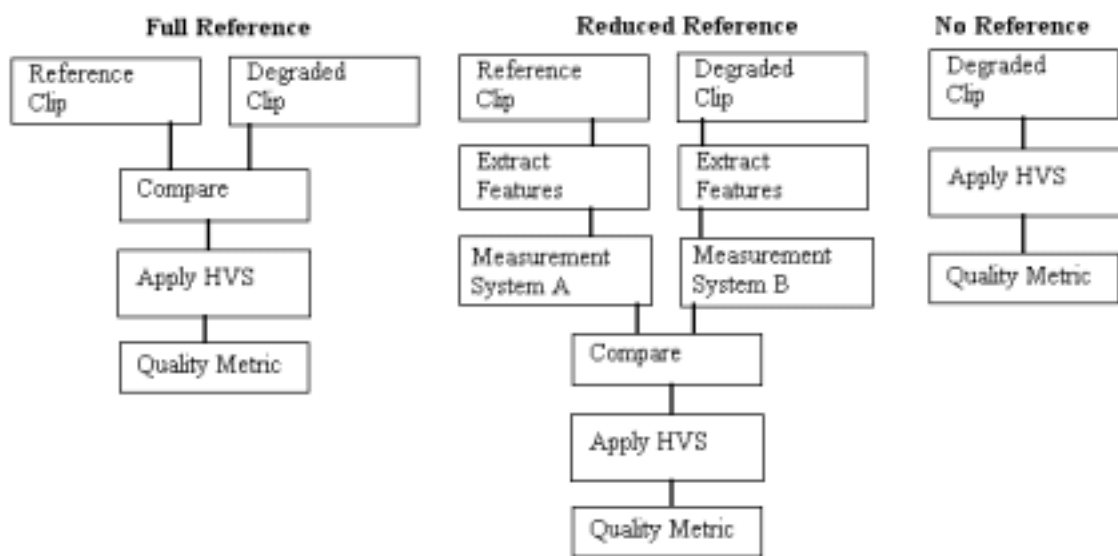


FIGURE 3.15: OBJECTIVE METRIC MEASUREMENT METHODOLOGIES

### 3.4.1. VQEG

The Video Quality Experts Group (VQEG) is a group formed in 1997 with the task of collecting reliable subjective ratings for a defined set of test sequences and to evaluate the performance of various video quality metrics [126]. In 2000, the VQEG performed a major study of various objective metrics on behalf of the ITU [127], [128] to compare the performances of various objective metrics against subjective testing. The results of the VQEG study found that:

- No objective metric is able to fully replace subjective testing.
- No objective metric statistically outperforms the others in all reference conditions.
- No objective metric statistically outperforms PSNR in all reference conditions.
- No single method can be recommended by VQEG to the ITU.

Many of the quality metrics described are extremely computationally intense and cannot be implemented for real-time analysis.

### 3.4.2. Review of Objective Metrics

#### *PSNR*

The PSNR metric is computed only using the luminance signal. An 8-bit image will contain pixel luminance values that vary from 0 (black) to 255 (white). There are two definitions of PSNR, both formulae work as this metric calculates the relative values and not the absolute values. Typically PSNR values vary between 20 and 40 dB. The PSNR is defined according to the following formula:

$m$  = row.

$n$  = column.

$d(m,n)$  = degraded pixel value at  $p$ , and  $(m, n)$ .

$o(m,n)$  = original pixel value at  $p$ , and  $(m, n)$ .

MSE = Mean Square Error is the mean difference overall pixels.

RMSE = Square Root MSE.

$$PSNR = 20 \log_{10} \left( \frac{255}{RMSE} \right)$$

$$MSE = \frac{1}{NM} \sum_{m,n}^{N,M} [o(m,n) - d(m,n)]^2$$

This formula demonstrates that two images are different but does not indicate the visibility of this difference. For example, the MSE can be produced in a number of different ways. That is, consider an image where the pixel values have been altered slightly over the entire image and an image where there is a concentrated alteration in a small part of the image, both will result in the same MSE value. One will be more perceptible to the user than the other. This metric is concerned with noise and degradation of the signal between the original and reconstructed image. As previously mentioned, this metric does not incorporate any aspects of the HVS. The PSNR metric does not take the visual masking phenomenon into consideration i.e. every single errored pixel contributes to the decrease of the PSNR, even if this error is not perceived.

#### *Image Evaluation Based on Segmentation - CPqD*

This methodology for video quality assessment uses objective parameters based on image segmentation [129]. Scenes are segmented into plane, edge and texture regions. For example, blocking distortion can be measured using an edge detector applied to the plane regions of the video scene in which the visual perception of this distortion is more noticeable. A set of objective parameters is assigned to each of these regions and is computed by a direct comparison between original and impaired scenes.

A perceptual-based model is defined that predicts subjective scores by calculating the relationship between objective measures and results of subjective assessment tests. The relationship between each objective parameter and the subjective impairment level is approximated by a logistic curve, resulting in an estimated impairment level for each parameter. All estimators are applied to fields rather than frames of video to ensure the statistical reliability of the measures in scenes with a high level of motion. The final result is achieved by linearly combining the estimated impairment levels, where the weight of each impairment level is proportional to its statistical reliability.

#### *Picture Quality Rating (PQR) – Tektronix/Sarnoff*

The PQR metric is based on a visual discrimination model that simulates the responses of human spatio-temporal visual system and degree of perceptibility of differences between the reference and degraded clip measured in units of just-noticeable difference (JND). Analysis of the differences between the degraded clip and the reference clip with the Sarnoff/Tektronix Human Vision Model provides a measure of the PQR metric.

The Sarnoff Visual Discrimination Model (VDM) has been employed in measuring still image fidelity [130]. The VDM was later modified to the Sarnoff JND metric for color video [131]. It convolves, samples and decomposes the clips using a Laplacian transform for spatial frequency separation, local contrast computation, as well as directional filtering, from which a phase-independent contrast energy measure is calculated. The contrast energy measure is subjected to a masking stage, which comprises a normalization process from which a JND map is computed. The JND map consists of three maps, the luminance JND map, the chrominance JND maps and the combined luma-chroma JND map. Each of the three JND maps is reduced to produce a PQR value that predicts the perceptual ratings that viewers assign to a degraded colour image sequence relative to its reference.

#### *3D Spatio-Temporal Filters - NHK/Mitsubishi Electric Corp.*

The model emulates human-visual characteristics using 3D spatio-temporal filters, which are applied to differences between the reference and degraded clips. The filter operates on the luminance field. The filters applied to the clips are a pixel based spatial filter, a block based filter with a noise masking effect, a frame based filter and finally a sequence-based filter that measures motion vectors and performs object segmentation etc. Collectively these filters are known as the Human Visual Filter. The MSE is calculated by subtracting the degraded clip from the reference clip. Then, the MSE is weighted by the Human Visual Filter producing a measure of perceived quality.

*Motion Picture Quality Metric (MPQM)*

The Moving Picture Quality Metric (MPQM) [125] is based on a multi-channel vision model comprising of a local contrast definition and Gabor-related filters for the spatial decomposition, two temporal mechanisms, as well as a spatio-temporal contrast sensitivity function and a simple intra-channel model of contrast masking. A color version of the MPQM based on an opponent-colors space was presented as well as a variety of applications and extensions of the MPQM [132], e.g. for assessing the quality of certain image features such as contours, textures and blocking artifacts, or the study of motion rendition [133]. The main drawback of MPQM 's is that it is frequency-domain implementation of the spatio-temporal filtering process, which, is not practical for measuring the quality of sequences with a duration of more than a few seconds due to the huge memory requirements.

*Perceptual Distortion Metric (PDM) - EPFL*

The perceptual distortion metric (PDM) developed by EPFL is based on a spatio-temporal model of the human visual system [134]. It is based on the Normalization Video Fidelity Metric (NVFM) [135] which is a more practical quality assessment system similar to MPQM except it uses a pyramid transform for spatial filtering and discrete time-domain filter approximations for temporal analysis. It consists of four stages processing both the reference and the degraded clips pass. The first converts the input to an opponent colour space. The second stage implements a spatio-temporal perceptual decomposition into separate visual perception channels of different temporal frequency, spatial frequency and orientation. The third stage models effects of pattern masking by weighting according to spatio-temporal contrast sensitivity data to a model of contrast gain control. The final stage of the metric is a detection stage and computes a distortion measure from the difference between the measured parameters of the reference and degraded clips [136].

*Digital Video Quality (DVQ) - NASA*

The digital video quality (DVQ) metric incorporates many aspects of human visual sensitivity [137, 138]. The DVQ metric consists of several processing stages performed on the reference and degraded clips for both the temporal contrast sensitivity function and the spatial contrast sensitivity function. The first step consists of sampling, cropping, and colour transformations that confine processing to a region of interest and express the clips in a perceptual colour space. Local contrast is calculated by blocking and a Discrete Cosine Transform (DCT). Local contrast is the ratio of DCT amplitude to DC amplitude for the corresponding block. A temporal filtering operation determines the temporal contrast

sensitivity function. The spatial contrast sensitivity function is computed by converting the result to units of just-noticeable differences by dividing each DCT coefficient by its respective visual threshold. At the next stage the two sequences are subtracted. Finally a contrast masking operation is performed on the clips and the masked differences mapped to the perceptual error over various dimensions, and the pooled error is converted to visual quality (VQ).

*Perceptual Video Quality Metric (PVQM) - KPN/Swisscom CT*

The Perceptual Video Quality Metric (PVQM) [139] uses the same technique for measuring video quality as the Perceptual Speech Quality Measure (PSQM) in measuring speech quality [140]. The metric measures spatial and temporal distortions and localized spatio-temporal errors. The PVQM uses a special brightness/contrast adaptation of the distorted video clip. The spatio-temporal alignment is carried out by block a matching procedure.

The spatial luminance analysis is based on edge detection of the Y signal, whilst the temporal analysis is based on difference frames analysis of the Y signal. Elements of the HVS are reflected in the PVQM algorithm that calculates perceptual parameters in three stages. The first stage estimates the contrast sensitivity functions of the luminance and chrominance signals. The second computes the edginess of the luminance signal and the third stage computes the chrominance error. Finally, the results of these three stages are mapped to a single quality value, which correlates to subjective perception of the video quality.

*Video Quality Metric (VQM) - NTIA*

The Video Quality Metric (VQM) [141] uses feature extraction and analysis to calculate perceived quality of video. These processes include sampling of the reference and degraded video clips, calibration of the original and processed video streams, extraction of perception-based features from spatio-temporal blocks from the clips. These features were selected empirically from a number of candidates so as to yield the best correlation with subjective data. First, horizontal and vertical edge enhancement filters are applied to allow for spatial gradient computation in the feature extraction stage. The resulting sequences are divided into spatio-temporal blocks. A number of features measuring the amount and orientation of activity in each of these blocks are then computed from the spatial luminance gradient. To measure the distortion, the features from the reference and the degraded clip are compared using a process similar to masking.

# Chapter 4

## Optimum Adaptation Trajectories

### *Chapter Abstract*

As discussed in the previous chapter, most adaptive delivery mechanisms for streaming multimedia content do not explicitly consider user-perceived quality when making adaptations. This research proposes that an optimal adaptation trajectory through the set of possible encodings configurations exists, and that it indicates how to adapt encoding quality in response to changes in network conditions to maximize user-perceived quality. Such an optimum adaptation trajectory can be used to complement any transmission adaptation policy, which aims to maximise user-perceived quality of the delivered multimedia. This chapter describes the subjective tests that were carried out to find such trajectories for a number of different MPEG-4 video clips.

Section 1 proposes the concept of an OAT, and sections 2 to 5 describes the subjective testing processes that were carried out to discover this OAT. In section 6, the results from subjective testing are statistically analysed. Section 7 evaluates the validity of using interpolation on the results obtained. Section 8 analyses and discusses the results. Finally section 9 summarises this chapter and makes some conclusions.

### 4.1. Introduction

Best-effort IP networks are unreliable and unpredictable, particularly in wireless networks. There can be many factors that affect the quality of a transmission, such as delay, jitter and loss. Congested network conditions results in lost video packets, which, as a consequence, produces poor quality video. Further, there are strict delay constraints imposed by streamed multimedia traffic. If a video packet does not arrive before its playout time, the packet is effectively lost. As mentioned in Chapter 2, Section 2.6, packet losses have a particularly devastating effect on the smooth continuous playout of a video sequence due to inter-frame dependencies. A slightly degraded quality but uncorrupted video stream is less irritating to the user than an uncorrupted stream. However, rapidly fluctuating quality should also be avoided as the human vision system adapts to a specific quality after a few seconds and it becomes annoying if the viewer has to adjust to a varying quality over short time scales.

Controlled video quality adaptation is needed to reduce the negative effects of congestion on the stream whilst providing the highest possible level of service and quality.

One possible approach to the problem of network congestion and resulting packet loss and delay is to use feedback mechanisms to adapt the output bit rate of the encoders, which, in turn adapts the video quality, based on implicit or explicit information received about the state of the network. Several bit rate control mechanisms based on feedback have been presented in the last few years. As Real Time Control Protocol (RTCP) provides network-level QoS monitoring and congestion control information such as packet loss, round trip delay, and jitter. Many applications use RTCP to provide control mechanisms for transmission of video over IP networks. However, the network-level QoS parameters provided by RTCP are not video content-based and it is difficult to gauge the quality of the received video stream from this feedback. For example, assume that two video packets are lost during the transmission. One contains important information related to the video synchronization code, whereas the other only contains bits for rebuilding certain video blocks. Obviously, the impact of the first packet on the reconstructed video quality is much worse than that of the second packet. But RTCP cannot distinguish the difference since it does not indicate the content of the packets. That is to say, the network level QoS parameters provided by RTCP do not explicitly reflect the user-perceived quality. Over the past few years, video quality evaluation and measurement schemes have been developed. Some objective quality assessment techniques have been devised to provide perceptual video quality measures.

### **4.1.1. Adaptive Streaming**

The concept of adaptive video streaming is based on the widely accepted maxim that users prefer a reduced bit rate to packet losses [142]. A survey of the well-known adaptation policies is discussed in Chapter 3 Section 2. These adaptation policies regardless of whether they are sender-based, receiver-based, hybrid, or encoder-based, address the problem of how to adapt only in terms of adjusting the transmission rate, window size or encoding parameters and do not consider the impact of perception nor do they indicate how this bit rate adaptation can be achieved in terms of real-video encoding parameters. For example,

- The sender-based scheme Loss-Delay Algorithm (LDA) and its variants indicate how to adjust the transmission rate of the sender in response to various network conditions of loss and delay. But for a given bit rate, there are several ways to encode the content.
- The receiver-based RLM scheme (Receiver-driven Layered Multicast) uses a policy



of join/leave experiments for additional layers of video to improve the perceived quality and make the most of the available bandwidth. But the base and enhancement layers can be composed in a large number of different ways.

#### 4.1.2. Optimum Adaptation Trajectories

This research proposes that there is an optimal way in which multimedia transmissions should be adapted in response to network conditions to maximize the user-perceived quality. This is based on the hypothesis that within the set of different ways to achieve a target bit rate, there exists an encoding configuration that maximizes the user-perceived quality. If a particular multimedia file has  $n$  independent encoding configurations then, there exists an adaptation space with  $n$  dimensions. When adapting the transmission from some point within that space to meet a new target bit rate, the adaptive server should select the encoding configuration that maximizes the user-perceived quality for that given bit rate. When the transmission is adjusted across its full range, the locus of these selected encoding configurations should yield an OAT within that adaptation space.

This approach is applicable to any type of multimedia content. The work presented here focuses for concreteness on the adaptation of MPEG-4 video streams within a two dimensional adaptation space defined by **frame rate** and **spatial resolution**. These encoding variables were chosen as they most closely map to the spatial and temporal complexities of the video content. The example shown in Figure 4.1 indicates that, when degrading the quality from position [25,100%], there are a number of possibilities such as [15,100%], [20,85%] or [25, 70%] which all lie within a zone of Equal Average Bit Rate (EABR). The clips falling within a particular zone of EABR have different, but similar bit rates. For example, consider, the bit rates corresponding to the encoding points [17, 100%], [25, 79%] and [25, 63%] were 85, 88, and 82 kbps respectively. To compare clips of exactly the same bit rate would require a target bit rate to be specified, and then the encoder would use proprietary means to achieve this bit rate by compromising the quality of the encoding in an unknown manner. Using zones of EABR effectively quantises the bit rate of different video sequences with different encoding configurations. The boundaries of these zones of EABR are represented as linear contours for simplicity, since their actual shape is irrelevant for this scheme.

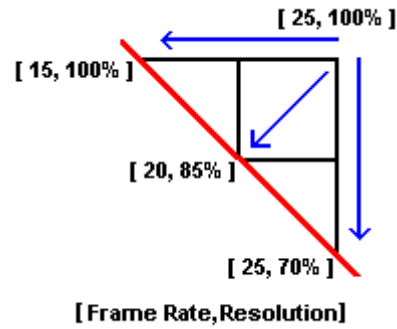


FIGURE 4.1: PROBLEM STATEMENT

The OAT indicates how the quality should be adapted (upgraded or downgraded) so as to maximize the user-perceived quality. The OAT may be dependent on the characteristics of the content. There is a content space in which all types of video content exist in terms of spatial and temporal complexity (or detail and action). Every type of video content within this space can be expanded to an adaptation space as shown in Figure 4.2.

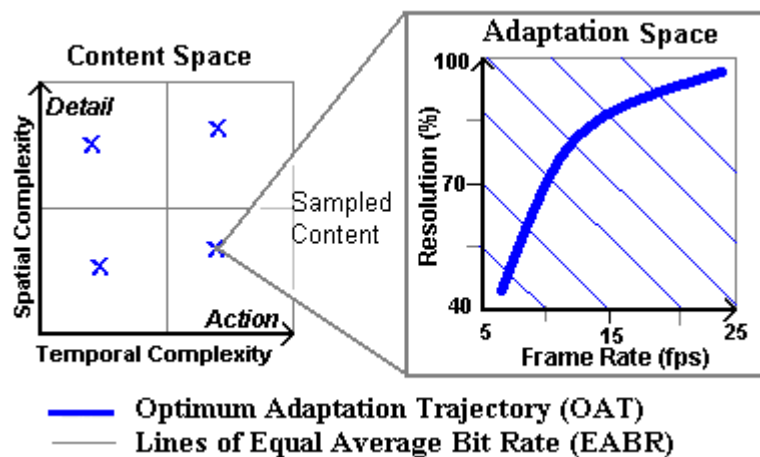


FIGURE 4.2: MAPPING CONTENT SPACE TO ADAPTATION SPACE

### 4.1.3. Human Perception

A key issue addressed in this research is how to adapt in order to maximize the resulting user-perceived quality. This raises the question of how user-perceived quality can be assessed in practice. In order to understand the quality and perceived quality of a video sequence, the human visual system (HVS) must be considered. It is well known that the commonly used metric, the signal-to-noise ratio (SNR) is not correlated with human vision, [143]. The HVS is extremely complicated but it has been shown that visual perception is mainly governed by two key concepts:

- Contrast sensitivity: accounts for the perception of a single stimulus. The contrast

sensitivity of the HVS is when a signal is detected only if its contrast is greater than some threshold defined as the detection threshold.

- Masking: quantizes the interactions between several stimuli. A stimulus will be perceived differently as a function of the background onto which it lies. Both signals (the foreground and the background) interfere and their relative perception is different to their singular perception i.e. one stimulus only. Masking is the detection threshold of the foreground as it is modified as a function of the contrast of the background [144].

### **4.1.4. Limitations of Objective Metrics**

Clearly it would be difficult to produce an adaptive system that accounts for the many facets of the HVS, so some alternative strategy is required if user-perceived quality is to be accommodated. Several objective metrics of video quality have been reviewed in Chapter 3 Section 2, but they are limited and not satisfactory in quantifying human perception. Further, it can be argued that to date, objective metrics were not designed to assess the quality of an adapting video stream. This is shown in Chapter 5, Section 1, where the OATs are discovered using objective metrics to determine whether they correlate to those discovered by subjective testing. Despite significant research into the HVS, the ITU-T has yet to find or recommend an objective metric that correlates adequately to human perception, and they recommend that subjective testing is required to complement objective metrics [145], [146], [147] but also, more importantly, that objective metrics are not a direct replacement for subjective testing. Subjective quality assessment is more appropriate for research related purposes whereas objective metrics are more suited to equipment specifications and day-to-day system performance measurement. Given the lack of suitable objective metrics and the recommendations of the ITU-T, subjective testing was employed to discover the existence of an OAT.

## 4. 2. Subjective Testing Considerations

Subjective testing is a very broad science, which includes elements of psychology, psychophysics and cognitive science. There are many factors that need to be carefully considered in the design of an appropriate and experimentally sound subjective test plan. These include, the choice of suitable test material, the test methodology should reflect the goals of the experiments, the scoring methods and scale, results acquisition and analysis and finally subject considerations.

The primary goal of this subjective test plan is to discover the existence of an OAT in adaptation space, which maximizes user-perceived quality and ascertain the dependency of this OAT on the content characteristics.

### 4.2.1. Choice of Test Material

In order to control the characteristics of the source signal, the test sequences are defined according to the goal of the experiment. The clips are chosen to cover a broad spectrum of subject matter [148] in which the following factors are specifically considered:

- Spatial parameters
- Temporal parameters
- Audio parameters.

Audio information affects visual perception and therefore overall perception depending on the context of the audio content. This is a very complicated relationship to capture and analyse. Further, users are more sensitive to variations in audio quality. In general, the audio stream is not adapted, as it constitutes a small fraction of the overall bitstream and is considered the most important part of the stream. Therefore, in the subjective testing, the test sequences chosen have no audio part and consequently have no audio context since the purpose of the OAT is to adapt only the video stream.

### 4.2.2. Spatial and Temporal Considerations

Video clips can be used to capture a variety of events, which can vary greatly. Clips that contain high action detail may be highly dependent on temporal encoding parameters, such as frame rate. Inadequate frame rates for such clips result in incomprehensible output because the content of the clip is changing rapidly. These fast scene changes and high action content can prevent the succession of frames from clearly conveying actual events in the clip, thus inhibiting user perception and cognition. Consider, for example, the case of a

video clip of a football game. If the user sees a player running towards the goal in one frame and in the next frame the player is seen celebrating after scoring, this makes no sense to the viewer because they have not seen the important intermediate frames showing the player score. The converse is when there is a great amount of detail in the scene, which is necessary to the clips comprehension, appreciation and context. For example, consider a clip which contains subtitles, it is essential that the user can read the text regardless of what is happening in the clip. In this case, the encoding parameters relating to the spatial detail, clarity and quality must be given highest priority. There is a clear trade-off between spatial and temporal encoding parameters when making an adaptation decision for the clip.

Thus the spatial and temporal information of the scenes are critical parameters. These parameters, not only play a crucial role in determining the amount of video compression that is possible but also can affect how they are perceived. The spatial and temporal characteristics of the stream can greatly affect the way in which the video should be adapted and thus affect the discovery of the OAT. The selection of appropriate test sequences is a key point in the planning of subjective testing. The values of spatial and temporal information can be determined using spatial complexity and temporal complexity metrics. In order to select relevant test sequences, it is necessary to consider the relative spatial and temporal information found in the various sequences available [149]. If a small number of test sequences are to be used in a given test, it may be important to choose sequences that span the extremes of the spatial-temporal information plane.

The Spatial Information (SI) metric is based on a Sobel filter. The luminance plane of each frame at time  $n$  is filtered using a Sobel filter. The standard deviation over all the pixels in each Sobel filtered frame is calculated. This is repeated for each frame in the sequence and results in a time series of spatial information of the scene. The maximum value in the time series is chosen to represent the spatial information content for the scene.

The Temporal Information (TI) is based on a motion difference feature, which is the difference between the pixel values (of the luminance plane) at the same location in space but in successive frames. The TI value is taken as the maximum over the time series. Scene cuts can result in miscalculation of TI and so to avoid this, two values could be taken, one where the scene changes are included and another where they are excluded. Volume 2 Appendix C contains more detail on the spatial and temporal complexity metrics.

### 4.2.3. Subjective Test Methodologies

There are a number of different test methodologies. The choice of methodology and its corresponding grading scale should reflect the goals of the experiment. The main testing methodologies are [149]:

- Absolute Category Rating Method (ACR)
- Degraded Category Rating Method (DCR)
- Pair Comparison Method (PC)

Others include:

- Forced Choice
- MUltiple Stimulus Hidden Reference and Anchors (MUSHRA)

These discussed in more detail in Chapter 3 Section 3. To discover the OAT, the Forced Choice Methodology was used. This decision was motivated by several factors:

- The facility of testing for reliability and replication.
- The simplicity of the grading scale and subsequent statistical analysis.

The forced choice methodology is often employed in cognitive science, and PC is one of its applications. In forced choice, the subject is presented with a pair of alternatives separated by a short gap or signal. The subject must choose one of the alternatives according to some test criteria. (Figure 4.3).

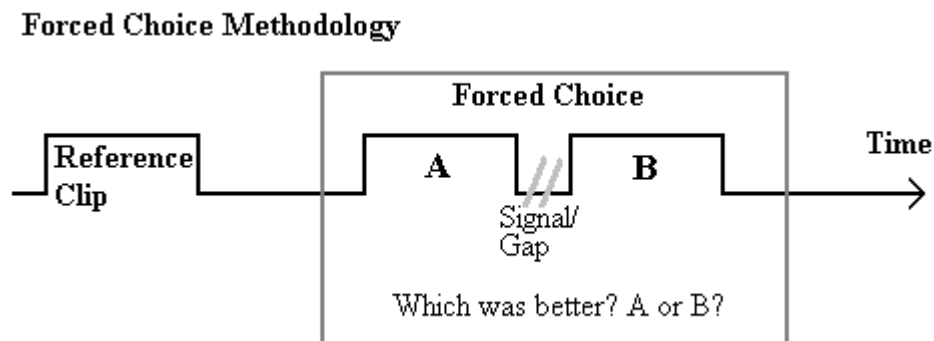


FIGURE 4.3: FORCED CHOICE METHODOLOGY

At the beginning of the test procedure, the reference clip is shown. During a single trial, the subject is shown two degraded versions of the same clip, 'A' and 'B'. A degraded version of the clip is one with a lower encoding configuration. These clips are shown consecutively separated by a short signal or gap. The subjects' task is to choose whether the first or second clip was better. It is not a forced choice when the reference clip is presented along with a degraded clip, as the two clips are not equal. In forced choice, there are equally probable alternative degraded versions of the clip between which the subject must choose. When a subject cannot make a decision, they are forced to make a choice. The bias is binary, which

simplifies the grading scale or rating procedure allowing for reliability, verification and validation of the results.

### 4.2.4. Known Criticisms of Conventional Subjective Testing Methods

A criticism of the ACR, DCR and PC methods, is the vocabulary of the impairment quality scale [150]. How subjects interpret the impairment scale is highly subjective. There is a subtle difference between impairment and quality which subjects may not be able to distinguish. There is also human behaviour to consider, when presenting degraded versions and making relative comparison, subjects tend to place their grades towards the low end of the scale [151]. However, if using the forced choice method, the bias is binary, better or worse, simplifying the rating procedure.

The speed of the score rating and presentation of the clips is another factor to be addressed. When showing clips, they must be shown in a very short interval, to prevent the user forgetting the quality of the previous one, these effects are known as forgiveness and recency. For the DCR method, the voting time is flexible and should only take approximately 10 seconds. With the Double Stimulus Continuous Quality Scale (DSCQS) rating scale, this may not be long enough. However, if using the Forced Choice method, the user must make a very quick decision, choosing either the first or second clip as being better or worse.

The duration of the test sequences should be quite short and is recommended at 10 seconds in duration. A short duration is necessary so as to ensure that the pair of clips is fresh in the subjects mind [152]. Giving the shortness of the clips, the subject may only perceive noticeable degradation at a particular point in the play out. Thus, simultaneous presentation of the clips could lead to mis-perception. In reality, users can only watch one thing at a time, and studies have shown that users can only concentrate on a particular portion of the screen at a time, for example, a clip where there are subtitles and video playing, the user can only concentrate or truly perceive one at a time.

### 4.2.5. Subject Considerations

- **Subject visual impairments:** Prior to a session, the subjects are screened for normal visual acuity or corrected-to-normal acuity and for normal colour vision.
- **Fatigue and boredom:** These could have a significant impact on a subjects' perception, the subject should be informed that they can leave at any time and should inform the tester if they are suffering from either fatigue or boredom to prevent a void test result. If

all subjects are being presented with the entire range of clips within a content category, then depending on the length of the test process, there may need to be a scheduled break.

- **Randomisation within testing:** In order to prevent users becoming trained video quality assessors towards the end of the test process, tests should incorporate randomized “within subject tests” and also include some dummy tests by swapping the order of the reference and degraded clips.

#### **4.2.6. Reliability and Replication**

To ensure reliable results from the subject, there should be repetition of the same test conditions (with the same test sequences) for the same subject.

Reliability of a subjective test:

- Intra-individual ("within subject") reliability refers to the agreement between a certain subject's repeated ratings of the same test condition;
- Inter-individual ("between subjects") reliability refers to the agreement between different subjects' ratings of the same test condition.

At least two, if possible more, replications should be included (i.e. repetitions of identical conditions) in the experiment. The main reason being that "within subject variation" can be measured using the replicated data. Replications make it possible to calculate individual reliability per subject and, if necessary, to discard unreliable results from some subjects. An estimate of both within- and between- subject standard deviation is a prerequisite for making a correct analysis of variance and to generalize results to a wider population. By randomizing the presentation of test sequences between subjects, learning effects within a test are to some extent balanced out.

#### **4.2.7. Population Sample**

The possible number of subjects in a viewing test is typically from 4 to 40. Four is the absolute minimum for statistical reasons, while there is rarely any point in going beyond 40. The actual number in a specific test should really depend on the required validity and the need to generalize from a sample to a larger population. In general, at least 15 observers should participate in the experiment. They should not be directly involved in picture quality evaluation as part of their work and should not be experienced video quality assessors.

The first phase of testing should be over-tested and over-sampled. The experience of this will help to decide the acceptable sample population size to get meaningful results during



subsequent testing. Using a sample that is pre-determined and turns out to be too small may lead to many difficulties in replicating test conditions. In addition, two independent samples should be taken. That is, the test should be conducted in parallel and independently. The results from the two samples should return the same results, thus verifying the procedure and results obtained.

### 4.3. Test Sequence Selection

Standard test sequences (Figure 4.4) are created and made available by the Video Quality Experts Group (VQEG) [153] [154].








PHASE 1			
			
src5_ref__625	src19_ref__525	src21_ref__525	src14_ref__525
Canoe C1	Foot ball C2	Susie C3	Washington DC C4
PHASE 2			
			
src6_ref__625	src9_ref__625	src18_ref__525	
Formula-1 C7	Rugby C9	Yosemite C12	

FIGURE 4.4: SCREEN SHOT OF TEST SEQUENCES

Of the many test sequences available, seven were chosen, four of which were used in the first phase of testing and the remaining three in phase two (Chapter 4, Section 4). The spatial and temporal complexity of these raw YUV reference test sequences was calculated in [155] (Figure 4.5).

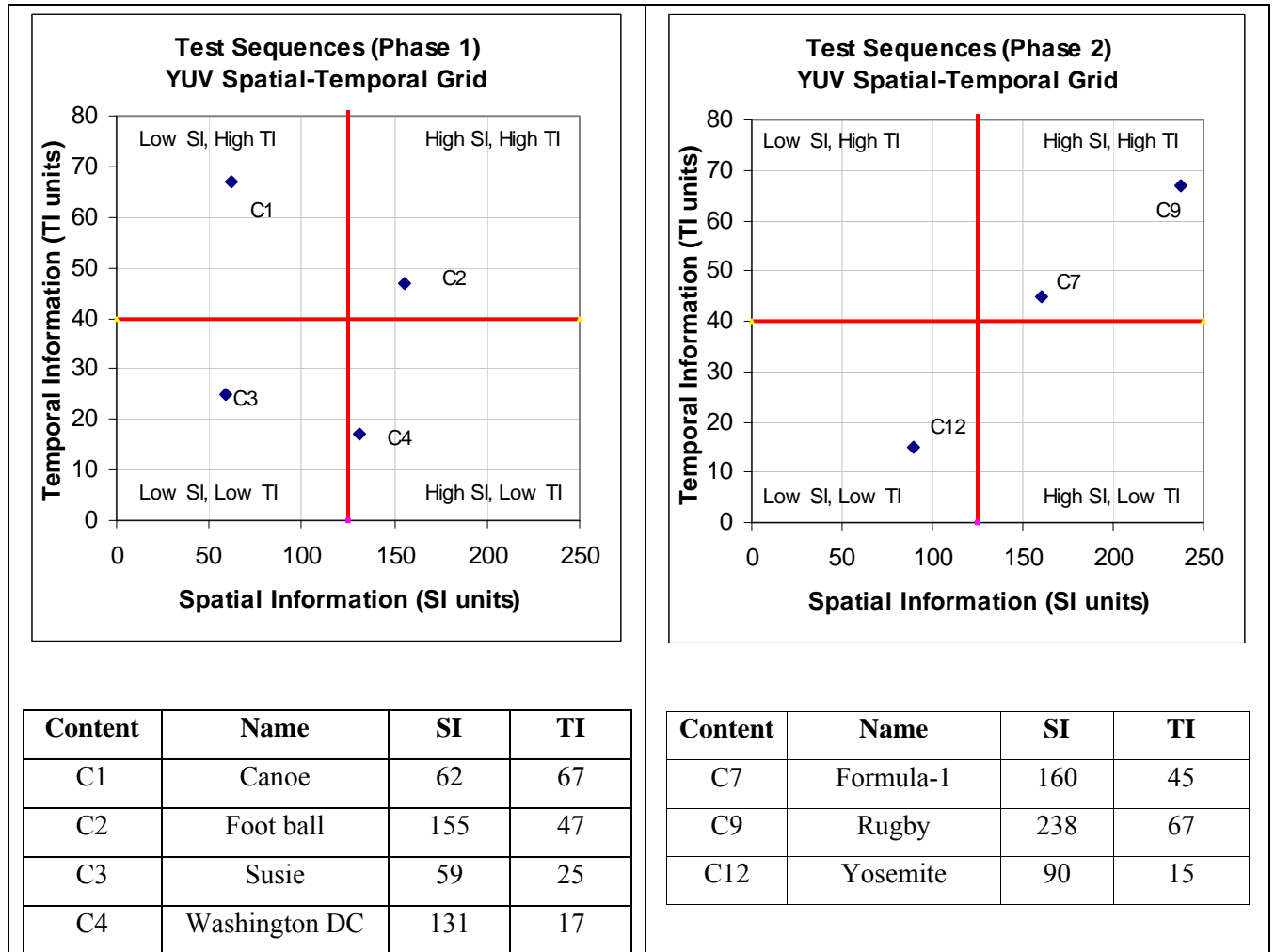


FIGURE 4.5: TEST SEQUENCE SI-TI VALUES

The test sequences used were selected so as to cover as much of the content space as possible thereby testing the most extreme cases of spatial and temporal complexity. In general, the test sequences could be classified as shown in Table 4.1.

Sequence	Spatial Complexity	Temporal Complexity
C1	Low	High
C2	High	High
C3	Low	Low
C4	High	Low
C7	High	High
C9	High	High
C12	Low	Low

TABLE 4.1: GENERAL TEST SEQUENCE CLASSIFICATION

The spatial and temporal complexity information values are calculated by VQEG for the raw YUV test sequences. When the test sequences are encoded, these values will be attenuated, as there will be a loss of both spatial and temporal information in the sequence. However, this does not affect the choice or use of the test sequences, as these values are only needed for an estimation of the clips content. The spatial and temporal complexity of the sequences has been further analysed in Volume 2 Appendix C.

### 4.3.1. Test Sequence Preparation

The test sequences were only available in raw YUV format. YUV format is raw uncompressed video data, which results in a 10 second clip being over 300,000KB. Even a high performance PC has difficulties in a smooth continuous playout of these files due to the large amounts of memory required to play them. These raw YUV files had to be transcoded to a more manageable format, uncompressed AVI, using the publicly available program, yuv2avi.exe, from StreamCrest [156].

From uncompressed AVI, the test sequences were then encoded to a reference test sequence with MPEG-4 encoding, using the most accurate, best quality compression at 25fps with a key frame every 10 frames and QCIF (176x144) resolution. During the preparation of the test sequences for the subjective testing, the encoding method used was the “most accurate”, that is, no target bit rate was specified, and the encoder followed the supplied encoding parameters as closely as possible regardless of the resulting bit rate. The encoder would not use proprietary means to achieve the target bit rate. Not only is the method of achieving the target bit rate proprietary, thus making the results of the subjective testing strongly encoder-dependent but also details of its operation are unknown.

- Application: QuickTime Player Pro v6.4.
- Encoder: MPEG-4 Video, Simple Profile
- Compression method: Most accurate compression
- Frames Per Second = 25fps
- Key frame every 10 frames
- Display Size = QCIF (176 x 144)

#### 4.4. Sampling the Adaptation Space

The subjective tests were conducted in two phases.

**Phase 1:** considered 4 test sequences, one taken from each quadrant of the SI-TI grid. Adaptation space was sampled using a logarithmic scale, to reflect Weber's Law of Just Noticeable Difference. The frame rates tested were {5, 7, 11, 17, 25} fps and the spatial resolutions were {100%, 79%, 63%, 50%, 40%}. A total of 120 subjects were tested during Phase 1.

**Phase 2:** considered different test sequences with similar SI-TI values to those used for Phase 1. However, this time, the adaptation space was sampled using a linear scale. The frame rates tested were {5, 10, 15, 20, 25} fps and the spatial resolutions were {100%, 85%, 70%, 55%, 40%}. A total of 40 subjects were tested during Phase 2.

The purpose of having two different test phases was to verify and validate the results from Phase 1. In addition, by using different encoding scales, different encoding configurations could be tested and it could be ascertained that the OAT was similar in shape regardless of whether a linear or logarithmic scale was used, and regardless of the encoding points tested. Both test phases should return similar results regardless of the encoding configurations tested.

##### 4.4.1. Weber's Law of Just Noticeable Difference (JND)

The "Just Noticeable Difference" is the minimum amount by which stimulus intensity must be changed in order to produce a noticeable variation in perception. Weber's Law states that, perception is proportional to the logarithm of stimulus [157] [158]. So, no matter what the intensity of the initial stimulus is, the proportional change will always be the same for a given stimulus. In context for any of the sensory perception, the amount of change of a present stimulus when the magnitude increases or decreases will always be perceived proportionally the same to the initial magnitude, no matter how intense the stimulus is. The Weber Fraction shows that the JND is at a constant proportion to the magnitude of the initial stimulus. Webers law is used as a basis for determining the logarithmic encoding scale of adaptation space for phase one of testing.

The spatial resolution and frame rate parameters for phase one of testing were chosen to reflect Weber's Law of JND. To determine increments of JND, the following procedure was used.

Let the number of sample points,  $i=5$ .

Let  $k = 0, 1, 2, \dots (i-1)$ .

The general formula is:

$$Parameter\_Value = \text{Log}^{-1} \left[ \text{Log}(Min) + k \left( \frac{\text{Log}(Max) - \text{Log}(Min)}{i-1} \right) \right]$$

Where *Max* and *Min* represent the maximum and minimum value for a particular encoding parameter.

#### 4.4.2. Frame Rate Parameters

In the reference sequence at 25fps, each frame has a sample duration of 40ms. To achieve a frame rate of 20fps, the reference test sequence was re-encoded with exactly the same encoding parameters such as resolution, frame sequence and frame rate parameters (i.e. a key frame every 10 frames). The resulting degraded test sequence contained 20fps, but each frame now has a sample duration of 50ms, and so on for each of the tested frame rates. This means that even though, frames have been dropped, there is no jerkiness or stuttering of the playout of the degraded test sequence. These effects of jerkiness would have a significant impact on the perception of the test sequence.

The HVS cannot appreciate more than 24fps [159]. The upper frame rate is bounded at 25fps and the lower frame rate is bounded at 5fps, as below this value, the video sequence fails to be a motion sequence but a rather becomes a series of images.

##### *Phase 1: Logarithmic Scale*

During phase one of subjective testing, a logarithmic scale was used.

The maximum number of steps is,  $i=5$ .

Maximum frame rate = 25fps.

Minimum frame rate = 5fps.

Frame rate values = {5, 8, 11, 17, 25} fps.

##### *Phase 2: Linear Scale*

During phase two of subjective testing, a linear scale was used.

The maximum number of steps is,  $i=5$ .

Maximum frame rate = 25fps.

Minimum frame rate = 5fps.

Frame rate values = {5, 10, 15, 20, 25} fps.

### 4.4.3. Spatial Resolution Parameters

To achieve a spatial resolution of 90%, for example, the reference clip was spatially sub-sampled at 90%(QCIF) maintaining the aspect ratio of the clip i.e.  $\sim 158 \times 130$ , and then the up-sampled to maintain the original display size without any modification to the bit rate or encoding parameters of the test sequence. This emulates the behaviour of an adaptive transmitter that uses spatial resolution to reduce its output bit rate, with a client that rescales so that the actual display size is not varied during play-out. The upper spatial resolution is bounded at 100% and the lower spatial resolution is bounded at 40%. It was observed that spatial resolution below 40% was highly distorted.

#### *Phase 1: Logarithmic Scale*

During phase one of subjective testing, a logarithmic scale was used.

The maximum number of steps is,  $i=5$ .

Maximum spatial resolution = 100%.

Minimum spatial resolution = 40%.

Resolution values = {40, 50, 63, 79, 100}%.

The determination of the spatial resolution is obtained as a percentage of the original resolution (Table 4.2). The original sequence is in QCIF size, i.e.  $176 \times 144$ . For example, to achieve a spatial resolution of 79%, the original was sub-sampled at 79%(QCIF) i.e.  $\sim 140 \times 115$ , and then the sequence was up-sampled to maintain the original display size.

<b>Resolution</b>	<b>Sub-sampled</b>	<b>Up-sampled</b>
100%	176 x 144	QCIF
79.53%	140 x 115	QCIF
63.25%	111 x 91	QCIF
50.3%	89 x 72	QCIF
40%	70 x 58	QCIF

TABLE 4.2: LOGARITHMIC RESOLUTION SCALE

#### *Phase 2: Linear Scale*

During phase two of subjective testing, a linear scale was used.

The maximum number of steps is,  $i=5$ .

The maximum resolution = 100%.

The minimum resolution = 40%.

Resolution values (%) = {40, 55, 70, 85, 100}%.

Resolution	Sub-sampled	Up-sampled
100%	176 x 144	QCIF
85%	150 x 122	QCIF
70%	123 x 101	QCIF
55%	97 x 79	QCIF
40%	70 x 58	QCIF

TABLE 4.3: LINEAR RESOLUTION SCALE

#### 4.4.4. Adaptation Space

Each reference clip was encoded using the encoding configuration parameters obtained for frame rate and spatial resolution (Figure 4.6).

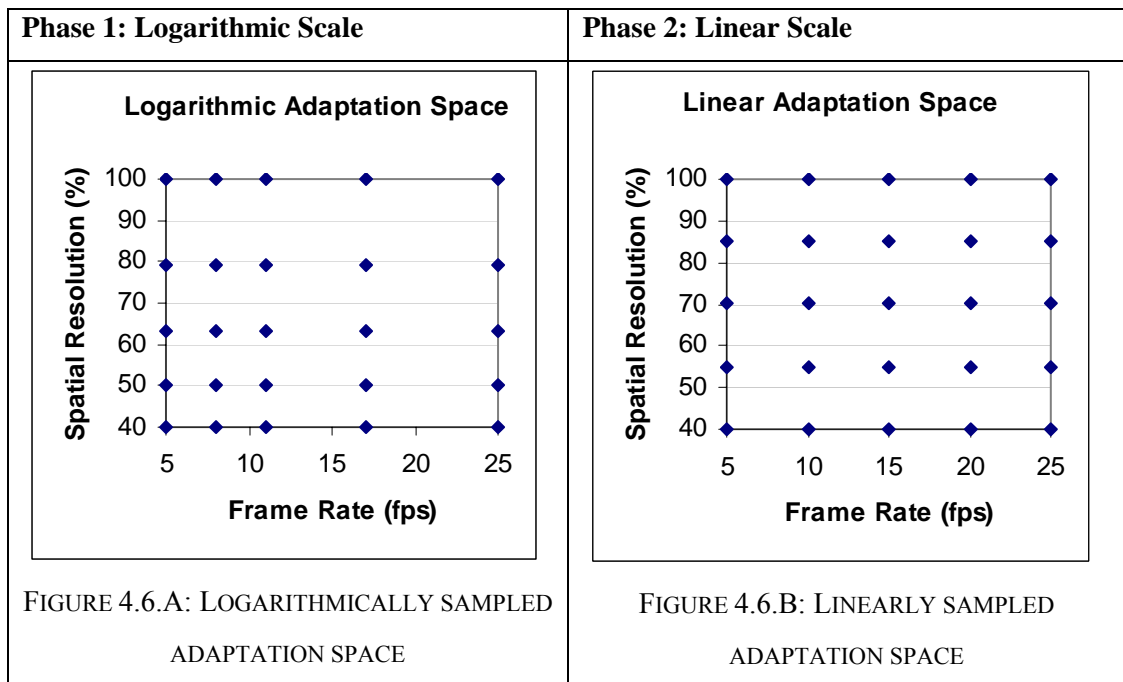


FIGURE 4.6: SAMPLING ADAPTATION SPACE

Once all the test sequences were prepared, their resulting bit rates were measured (Volume 2 Appendix E). It can be seen that clips with higher spatial and temporal complexity have a greater bit rate than those with lower complexities. It can also be seen that by reducing the encoding parameters, this naturally manifests itself in a reduced bit rate for the sequence.

#### 4.4.5. Determining Test Cases

Recall from Chapter 4, Section 1, that clips falling within a particular zone of Equal Average Bit Rate (EABR) have different (but similar) bit rates. For example, consider, the

bit rates corresponding to the encoding points [17, 100%], [25, 79%] and [25, 63%] were 85, 88, and 82 kbps respectively. To compare clips of exactly the same bit rate would require a target bit rate to be specified, and then the codec would use proprietary means to achieve this bit rate by compromising the quality of the encoding in an unknown manner. By using zones of EABR, the bit rate of different test sequences with different encoding configurations is effectively quantised which in turn dramatically reduces the number of test cases.

The size of the zones of EABR was chosen such that each zone contained only 3 or 4 clips. It would have been more accurate to test all pairs of clips within a zone of EABR but there was a need to reduce the number of test cases for each clip and limit the test duration for each subject to 30 minutes. There is no knowledge of where the OAT might lie in the adaptation space. Thus, only the encodings with differing frame rate and resolutions within a zone of EABR were compared. It was considered counter-intuitive that a user should prefer a clip with a lower resolution when all other encoding factors are exactly the same. For example, it was considered redundant to compare clips of [25, 79%] and [25, 63%] as it is expected that the clip with [25, 79%] would be preferred. A full listing of the subjective tests for each content type and the final data results are in Volume 2 Appendix E: Data Results.



## 4.5. Test Procedure

The test procedure had to be carefully controlled so that all variables were kept consistent throughout the testing phase, which could lead to ambiguity in the results obtained.

These test considerations are:

- Parallel testing
- Test environment
- Viewing device
- Viewing distance
- Subject screening
- Test methodology

### 4.5.1. Parallel Testing

Two people conducted the tests in parallel and independently not only for the purpose of speed and efficiency, but also to ensure that the results obtained from the testing were not influenced by the tester conducting the test. Thus, if there was a good correlation between the results obtained from both testers working independently conducting the same tests using the same devices with exactly the same setup and configuration on different subjects and in a different testing room, then the test procedure and methodology could be validated as being independent of these factors.

### 4.5.2. Test Environment

To maintain consistency in the test environment between tests, the tests were carried out under artificial lighting conditions with no interfering external light sources. The lighting environment was comparable to that of a typical office environment. No direct sunlight was allowed into the room but there was some natural light from windows. Further, it was ensured that there were no distractions or noise interferences during a test.

### 4.5.3. Viewing Device

As the intended application of these tests is to stream video to mobile devices, it was important to use a device that simulated the visual properties such a device. The most common screen types on a Personal Digital Assistant (PDA) or video enabled mobile phone are Liquid Crystal Display (LCD) and Thin Film Transistor (TFT) displays.

The use of PDAs as testing devices was considered, but due to the problems of,

- Physical device positioning to ensure a consistent test environment.

- Sequencing clips and conducting tests on such a device would require significant setup interference from the tester.
- Memory capacity of the device is limited and may not be able to store all the clips required to be tested.
- Some clips have a relatively high playout bit rate, which the device processor may not be able to support.

TFT screen laptops with video clips appropriately sized to simulate a PDA screen were decided upon as the best method of performing the test. To display the clips in the correct order and inter clip timings, a web page containing links to each clip was set up on each laptop. The video clips were displayed in a Quicktime player window in the centre of the screen with the background screen set to a neutral colour. To ensure reliability between devices all display and laptop system parameters were checked for consistency before each test session.

The laptop settings:

- Fujitsu Siemens LifeBook
- Mobile Pentium 4, 1.6GHz processor
- 256MB RAM
- Operating system: Microsoft Windows 2000
- Display: ATI Mobility Radeon 7500, True Colour 32Bit
- Total Screen Area: 1024x768

#### **4.5.4. Viewing Distance**

The viewing distance should be consistent and controlled for the duration of the test. A distance of three times the screen height (95cm) is used as recommended in the ITU-T Recommendation P.910. The viewing distance was maintained by keeping the test subjects chair aligned with markings on the floor and requesting that the subject not lean forward during the test. Screen tilt was also kept constant as TFT screens characteristics tend to vary with viewing angle.

#### **4.5.5. Subject Screening**

Before tests were conducted each subject was required to fill in a questionnaire intended to identify subjects who could prejudice the results. Candidates who were ever involved with video encoding, video quality testing or research, were excused from the test, as were subjects who had visual impairments such as colour blindness. Although, subjects who wore glasses for viewing T.V. or working on a computer were allowed to participate in the testing only if they wore their glasses.

#### **4.5.6. Test Methodology**

The forced choice methodology was chosen for conducting the tests due to the simplicity and lack of ambiguity in the rating/grading process, i.e., which was better, the first or the second? At the beginning of the test, the subject is presented with the reference clip, so that they are familiar and accustomed to the content of the clip and how it will be displayed by the player on the screen. Then, the subject is shown a pair of clips with differing configurations of frame rate and spatial resolution. Immediately after the second clip has played out, the subject is forced to choose the clip, which they consider to have the better quality. When a subject cannot make a decision, they are forced to make a choice, so that the bias is binary, which simplifies the rating procedure and allows for reliability, verification and validation of the results. When a subject cannot make a decision, they are forced to make a choice, which shall be referred to as a Random Choice (RC). The occurrences of such RCs are measured to give an indication of pairs of clips that had little difference.

The tester does not hint, explain or define anything about quality, video, perception or even the goals of the experiment to the subject. In this way, the subject must make their choice based on their own perception and understanding of video quality. The actual disclaimer and information record used during subjective testing is shown in Volume 2 Appendix D.

## 4.6. Results and Statistical Analysis

### 4.6.1. Inter-Tester Reliability

In the testing phase, the pairs of clips were randomly labelled A and B, so the subject had no clues about the content or quality of the clip. In each of the tests, the subject was forced to choose either clip A or B. The percentage preference for each A or B for each tester was calculated to determine if there was a high correlation between the results from both testers and verify inter-tester reliability.

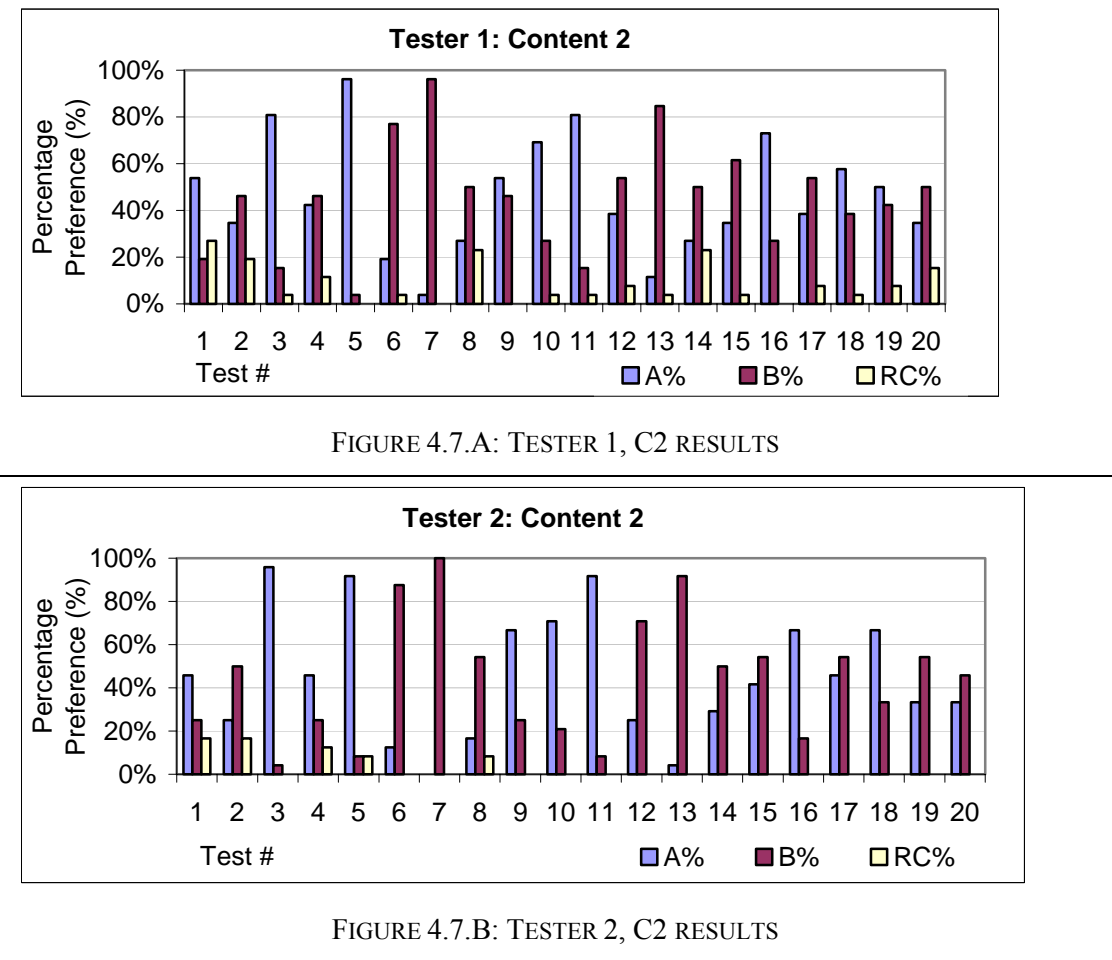
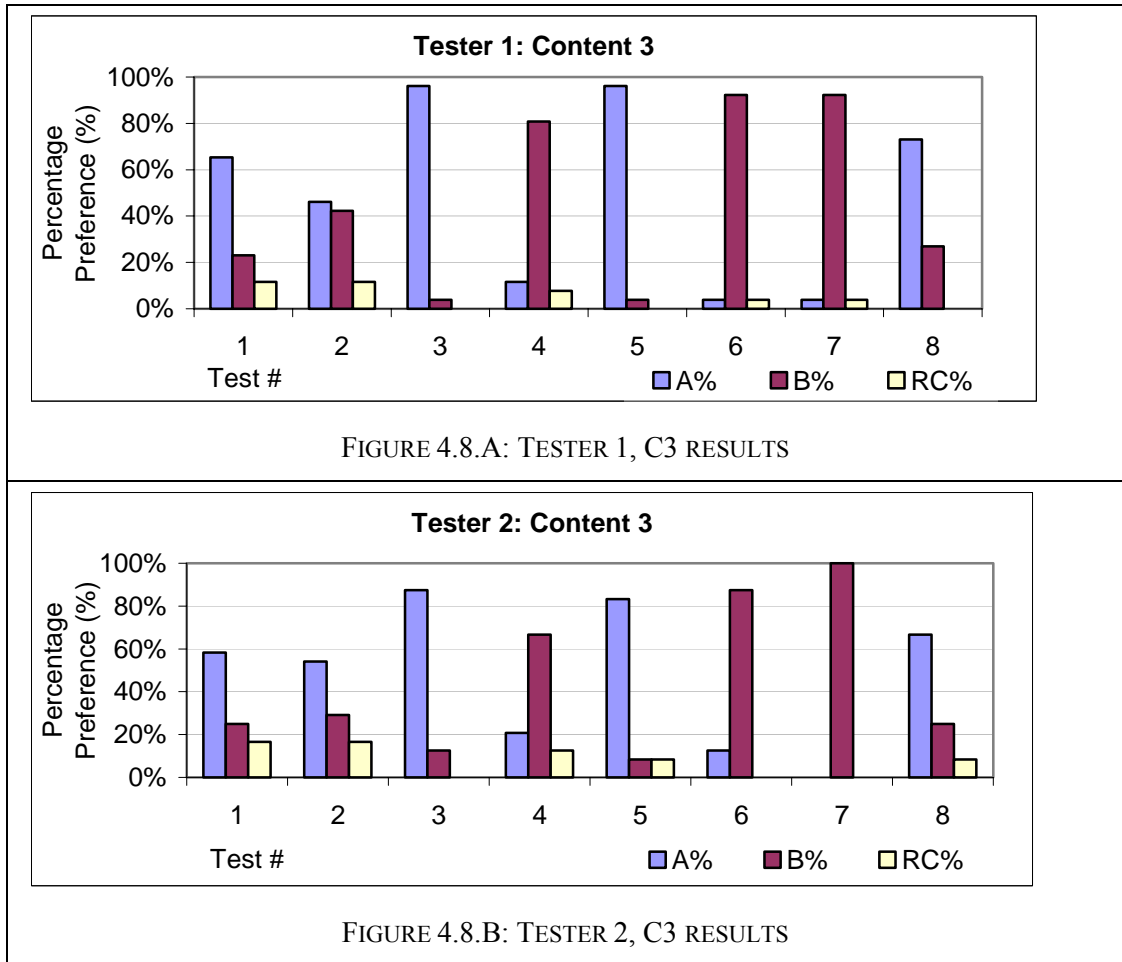


FIGURE 4.7: INTER-TESTER RELIABILITY FOR C2



The correlation between the two testers for the test sequences is:

For content 2 (Figure 4.7): 94%

For content 3 (Figure 4.8): 97%

Initially, 50 test participants from the general college population ranging in ages from 18-50 were tested. Once the results from each tester were analysed for inter-tester reliability and showed a good correlation between the results, the results for both testers could be combined. It was noted which choices were RCs. However, as the experiments progressed, subjects became more familiar with the test procedure and no longer mentioned which were RCs.

#### 4.6.2. Statistical Analysis

Having tested the results for inter-tester reliability, the results from both testers were combined. The true mean of the results must be calculated and the minimum sample size estimated. The sample size must be large enough to yield meaningful results. Finally, the data is tested for statistical significance [160].

*Estimating the True Mean of Population*

Due to the nature of the tests conducted, in that each pair of clips is an independent test from all the others and require a different population and sample size. A thousand samples were taken from the population tested and their respective means calculated. The sample means should converge around the true mean of the population,  $\mu$ , with a normal distribution.

Using the **Central Limit Theorem**, the sampling distribution of the sample mean is approximately normal for large samples. Consider a random sample of  $n$  observations selected from a population with a mean  $\mu$  and a standard deviation  $\sigma$ . When  $n$  is sufficiently large, the sampling distribution of  $x$  will be approximately a normal distribution with the mean  $\mu_{\bar{x}} = \mu$  and a standard deviation  $\sigma_{\bar{x}} = \frac{\sigma}{\sqrt{n}}$ . The larger sample size, the better will

be the normal approximation to the sampling distribution of  $\bar{x}$ . (Figure 4.9)

The sample mean  $\bar{x}$  is a point estimator of the population mean  $\mu$ . The sample variance  $s^2$  is a point estimator of the population variance  $\sigma^2$ .

Using the formula, the estimated population mean,  $\mu$ , is,

$$\bar{x} \pm 2\sigma_{\bar{x}} = \bar{x} \pm 2\frac{\sigma}{\sqrt{n}} \approx \bar{x} \pm 2\frac{s}{\sqrt{n}}$$

The confidence in this true mean calculation is derived from the fact that if random samples were repeatedly taken from the population, they would form the interval  $\bar{x} \pm 2\sigma_{\bar{x}}$  each time, approximately 95% of the intervals would contain  $\mu$ .

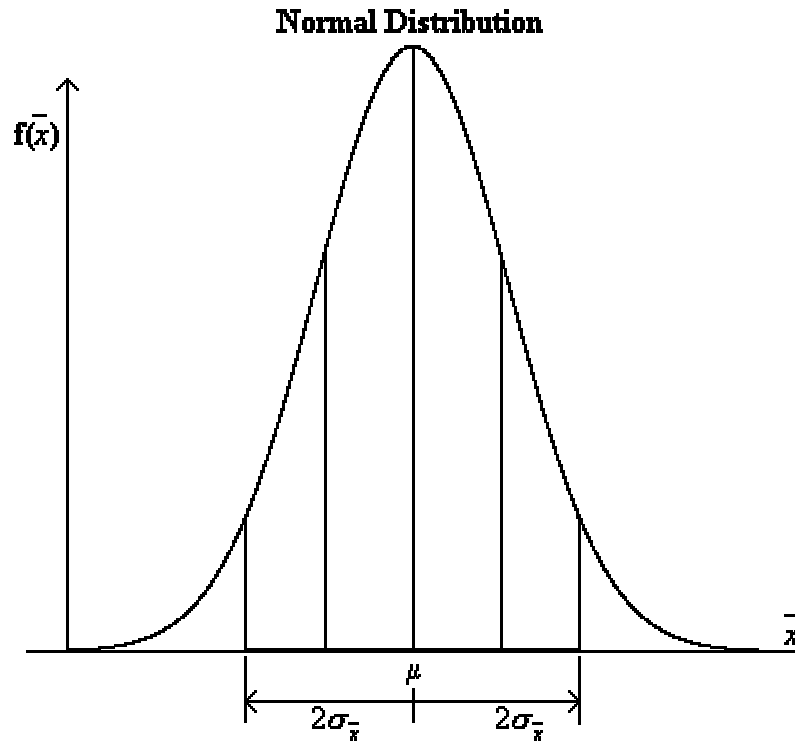


FIGURE 4.9: NORMAL DISTRIBUTION ABOUT THE MEAN

The confidence interval (Table 4.4) is a formula that indicates how to use sample data to calculate the interval that estimates a population parameter. The confidence coefficient is the probability that an interval estimator encloses the population parameter. The confidence level is the confidence coefficient expressed as a percentage.

<b>Confidence Levels</b>			
$100(1-\alpha)$	$\alpha$	$\alpha/2$	$z_{\alpha/2}$
90%	0.1	0.05	1.645
95%	0.05	0.025	1.96
99%	0.01	0.005	2.575

TABLE 4.4: CONFIDENCE INTERVALS FOR POPULATION MEAN ESTIMATION

Large sample 100(1- $\alpha$ )% Confidence Interval for  $\mu$ .

$$\bar{x} \pm z_{\alpha/2} \sigma_x = \bar{x} \pm z_{\alpha/2} \frac{\sigma}{\sqrt{n}}$$

$z_{\alpha/2}$  = the z - value with an area  $\alpha/2$  to its right (or left) of the mean and  $\sigma_x = \frac{\sigma}{\sqrt{n}}$

$\sigma$  = the standard deviation of the whole population.

$n$  = the sample size. When  $n > 30$ , the confidence can interval can be approximated :

$$\bar{x} \pm z_{\alpha/2} \frac{s}{\sqrt{n}}$$

$s$  = the sample standard deviation.

For example in Figure 4.10, for a confidence level of 90%,  $\alpha$  is 0.10 and  $\alpha/2$  is 0.05;  $z_{.05}$  is the z-value that locates an area 0.05 in the upper tail of the sampling distribution.

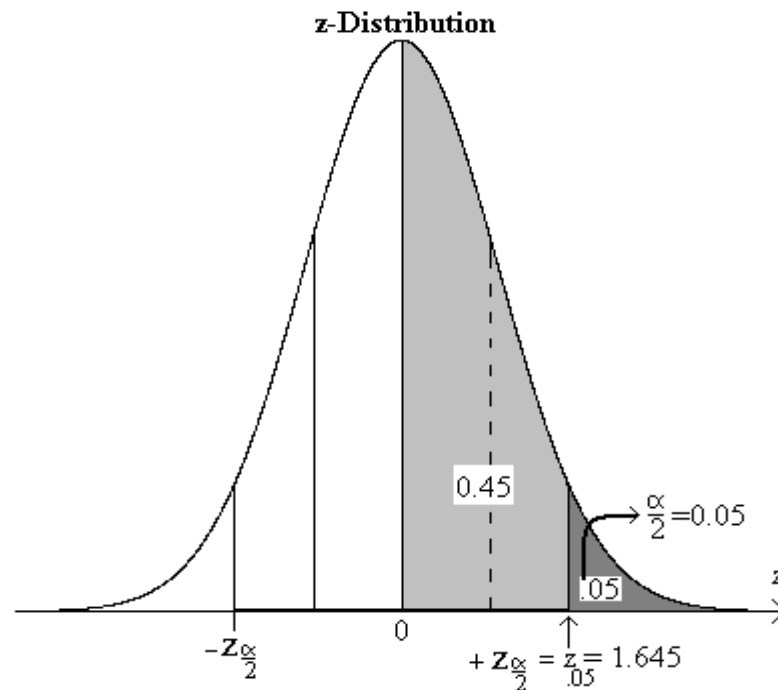


FIGURE 4.10: CONFIDENCE LEVELS AND Z-DISTRIBUTION

From a population of 50, the number of people who chose clip A was 35 i.e. 70% and the number who chose clip B was 15 i.e. 30%. By taking a thousand random samples of size  $n$  from the population, as  $n$  increases, the frequency of the mean converges to the true population mean with a normal distribution about the mean (Figure 4.11).



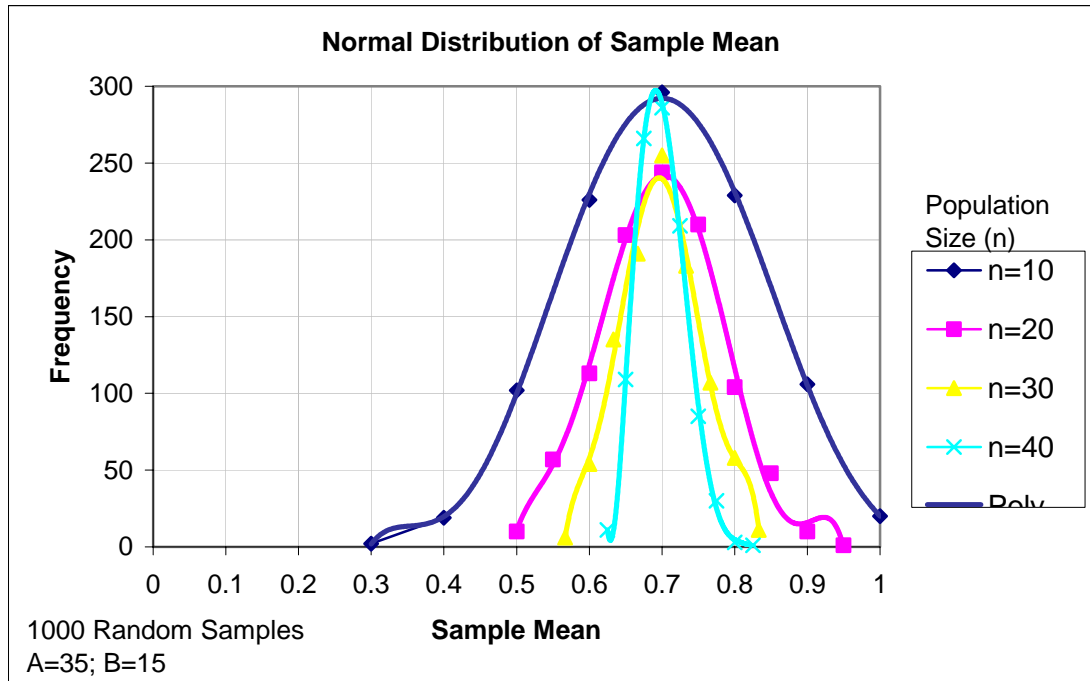


FIGURE 4.11: NORMAL DISTRIBUTION AROUND SAMPLE MEAN

*Estimation of Sample Size*

In order to estimate the true mean of the population to within a bound  $\beta$  with 95% confidence, the required sample size is:

$$z_{\alpha/2} \left( \frac{\sigma}{\sqrt{n}} \right) = \beta$$

$$n = \frac{\left( z_{\alpha/2} \right)^2 \sigma^2}{\beta^2}$$

$\beta=2\%$  was chosen, so the estimate is between +/- 2%.

The sample size for  $\beta=2\%$  and  $\beta=1\%$  is shown for content 2 in Table 4.5 and for content 3 in Table 4.6.

Content 2																				
Test #	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20
$\beta=2\%$	28	33	14	35	7	17	3	32	32	25	14	32	10	31	33	25	34	30	33	33
$\beta=1\%$	112	132	55	140	29	70	10	129	129	101	55	129	39	124	134	101	135	121	133	134

TABLE 4.5: ESTIMATED POPULATION SIZE FOR C2

Content 3								
Test #	T1	T2	T3	T4	T5	T6	T7	T8
$\beta=2\%$	28	34	9	17	9	10	3	25
$\beta=1\%$	112	137	38	70	38	39	10	101

TABLE 4.6: ESTIMATED POPULATION SIZE FOR C3

As each of the tests is independent, the sample size will vary from each pair of clips to be tested. So, in this way, the sample size for all the tests was calculated and then the maximum sample size was selected. The required sample size varied in the range from 4-34. So a sample size of 35 for  $\beta=2\%$  was selected. However, if  $\beta=1\%$ , the resulting sample sizes increases by a factor of 4 with the range of 10-140 population sample size.

### *Chi-Square Statistical Significance*

The Forced Choice Methodology is a typical binomial random variable experiment. The null hypothesis is that there is no preference for one degraded version of the clip over another, thus all clips within a zone of EABR should have equal probability.

### *Characteristics of Binomial and Multinomial Experiments*

The characteristics of binomial experiments are:

1. The experiment consists of  $n$  independent identical trials.
2. There are only 2 possible outcomes on each trial, a preference for clip A or B.
3. The probability of A remains the same from trial to trial and the probability of B remains the same from trial to trial.  $P(A)+P(B)=1$ .
4. The binomial random variable  $x$  is the number of A's in  $n$  trials.

There are several cases when there are three clips contained in a zone of EABR. When three clips fall into a zone of EABR, the results are combined. Again, the null hypothesis is that each clip should have equal preference of  $\frac{1}{3}$ .

1. The experiment consists of  $n$  identical trials.
2. There are  $k$  possible outcomes to each trial.
3. The probabilities of outcome  $k_i$  remains the same from trial to trial.
4. The sum of probabilities is  $\sum P(k)=1$ .
5. The trials are independent.

A binomial experiment is a special case of a multinomial experiment where  $k=2$ .

To test for statistical significance of preference for one clip over another within a zone of EABR, the chi-square test is applied. The chi-square test measures the degree of

disagreement of the data with the null hypothesis. The null hypothesis is that there is no preference for one degraded version of the clip over another, thus all clips within a zone of EABR should have equal probability. Should the null hypothesis be proven false, then there is a statistically significant preference for one clip over another within a zone of EABR.

For a multinomial experiment where there are three possible outcomes, A, B or C.

$p_A$  = Percentage of subjects with a preference for clip A.

$p_B$  = Percentage of subjects with a preference for clip B.

$p_C$  = Percentage of subjects with a preference for clip C.

The null hypothesis,  $H_0$ , is that there is no preference for one clip over another.

$H_0: p_A = p_B = p_C = 1/3$

$H$ : At least one proportion exceeds  $1/3$ .

If the null hypothesis is true, the expected mean value,  $E(n_i)$ , is denoted,

$$E(n_i) = p_i n = \frac{1}{3} (n)$$

Thus,  $E(n_A) \cong E(n_B) \cong E(n_C)$

In the chi-square test,  $\chi^2$ , measures the degree of disagreement between the data and the null hypothesis.

$$\chi^2 = \sum_{i=1}^n \frac{[n_i - E(n_i)]^2}{E(n_i)}$$

$n_A$  = Number of subjects with a preference for clip A.

$n_B$  = Number of subjects with a preference for clip B.

$n_C$  = Number of subjects with a preference for clip C.

$$\chi^2 = \frac{[n_A - E(n_A)]^2}{E(n_A)} + \frac{[n_B - E(n_B)]^2}{E(n_B)} + \frac{[n_C - E(n_C)]^2}{E(n_C)}$$

If there are  $k$  possible outcomes, then  $\chi^2$  has  $(k-1)$  degrees of freedom (df) with the critical values of  $\chi^2_\alpha$  dependent on the confidence level (Table 4.7).

<b>Confidence Level →</b>	<b>90%</b>	<b>95%</b>	<b>97.5%</b>	<b>99%</b>	<b>99.5%</b>
<b>df = (k-1)</b>	$\chi_{0.1}^2$	$\chi_{0.05}^2$	$\chi_{0.025}^2$	$\chi_{0.010}^2$	$\chi_{0.005}^2$
1	2.71	3.84	5.02	6.63	7.88
2	4.61	5.99	7.37	9.21	10.59
3	6.25	7.81	9.35	11.34	12.83

TABLE 4.7: CRITICAL VALUES FOR  $\chi_{\alpha}^2$

If  $\chi^2 > \chi_{\alpha}^2$ , it can be concluded (with  $100(1-\alpha)$  confidence level) that a statistically significant preference exists. The chi-square test was performed on all data and is shown for content 2 in Table 4.8 and for content 3 in Table 4.9.

<b>Content 2</b>										
<b>df</b>	<b>A</b>		<b>B</b>		<b>C</b>		<b>%A</b>	<b>%B</b>	<b>%C</b>	<b>ChiSq (<math>\chi^2</math>)</b>
	<b>Res (%)</b>	<b>Fr (fps)</b>	<b>Res (%)</b>	<b>Fr (fps)</b>	<b>Res (%)</b>	<b>Fr (fps)</b>				
1	100	17	79	25			0.70	0.30	NA	Significance (97.5%→99%)
2	100	11	79	27	40	25	0.44	0.50	0.06	Significance (>99.5%)
2	100	7	79	11	40	17	0.31	0.53	0.16	Significance (97.5%→99.5%)
2	100	5	79	7	50	11	0.20	0.51	0.29	Significance (<90%)
1	79	5	63	7			0.32	0.68	NA	Significance (95%→97.5%)

TABLE 4.8: CHI-SQUARE FOR C2

Content 3										
df	A		B		C		%A	%B	%C	ChiSq ( $\chi^2$ )
	Res (%)	Fr (fps)	Res (%)	Fr (fps)	Res (%)	Fr (fps)				
1	100	17	79	25			0.70	0.30	NA	Significance (97.5%→99%)
2	100	11	79	17	50	25	0.54	0.42	0.04	Significance (>99.5%)
2	100	7	63	11			0.74	0.26	NA	Significance (95%→97.5%)

TABLE 4.9: CHI-SQUARE FOR C3

#### 4.6.3 Data Results

The red line between the encoding points is the path of maximum user preference through the zones of EABR (Figure 4.10). This is the path, which had the highest preference regardless of how dominant this preference may be. Weighted points were then used to obtain the optimal adaptation perception (OAP) points. The weighted points were calculated as the sum of the product of preference with encoding configuration. For example, if 70% of subjects preferred encoding [17, 100%] and 30% preferred encoding point [25, 79%]. The interpolated weighted vector of these two points is  $[70\%(17)+30\%(25), 70\%(100\%)+30\%(79\%)]$  which yields the OAP point [19.4, 93.7%]. The blue line denotes the path of weighted preference by joining the OAP points. Finally a best curve fit is applied to the weighted path of preference.

Volume 2 Appendix E contains a full listing of the data results, including percentage preference for each zone of EABR; the maximum preferred encoding configuration and the OAP for each content type.

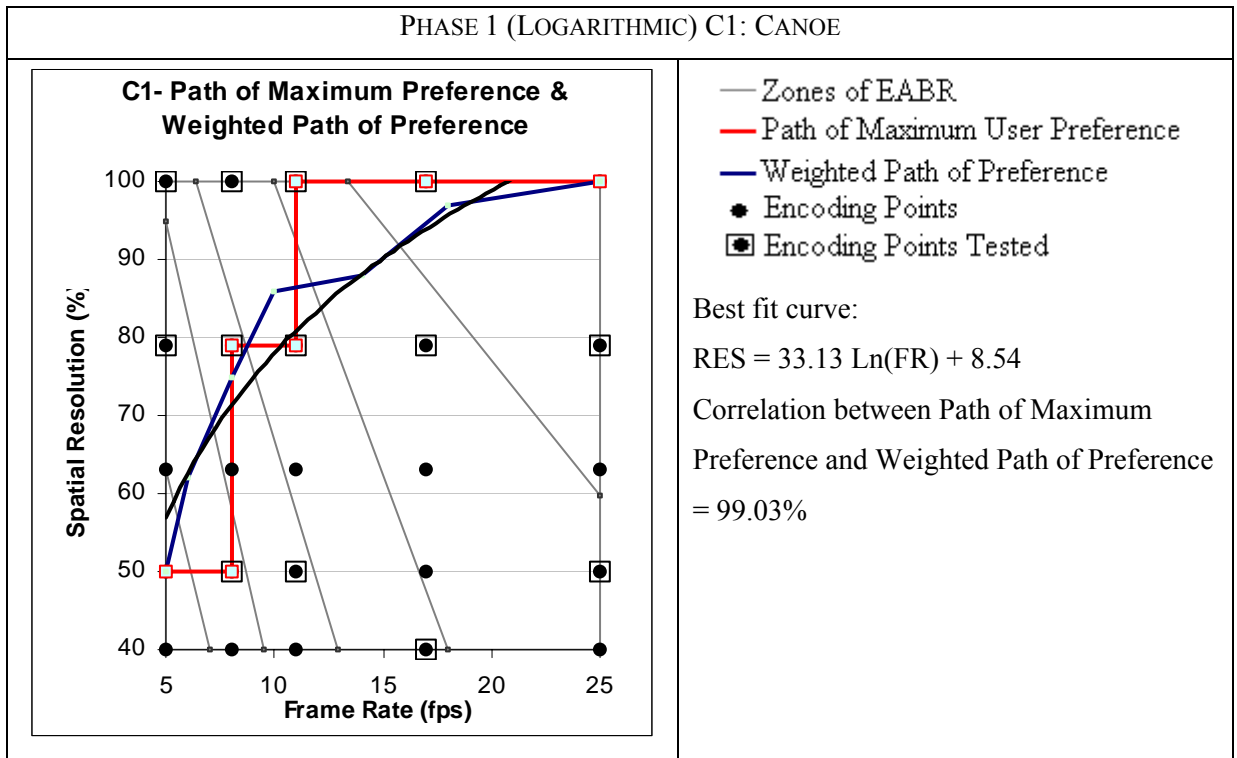


FIGURE 4.11: C1 RESULTS

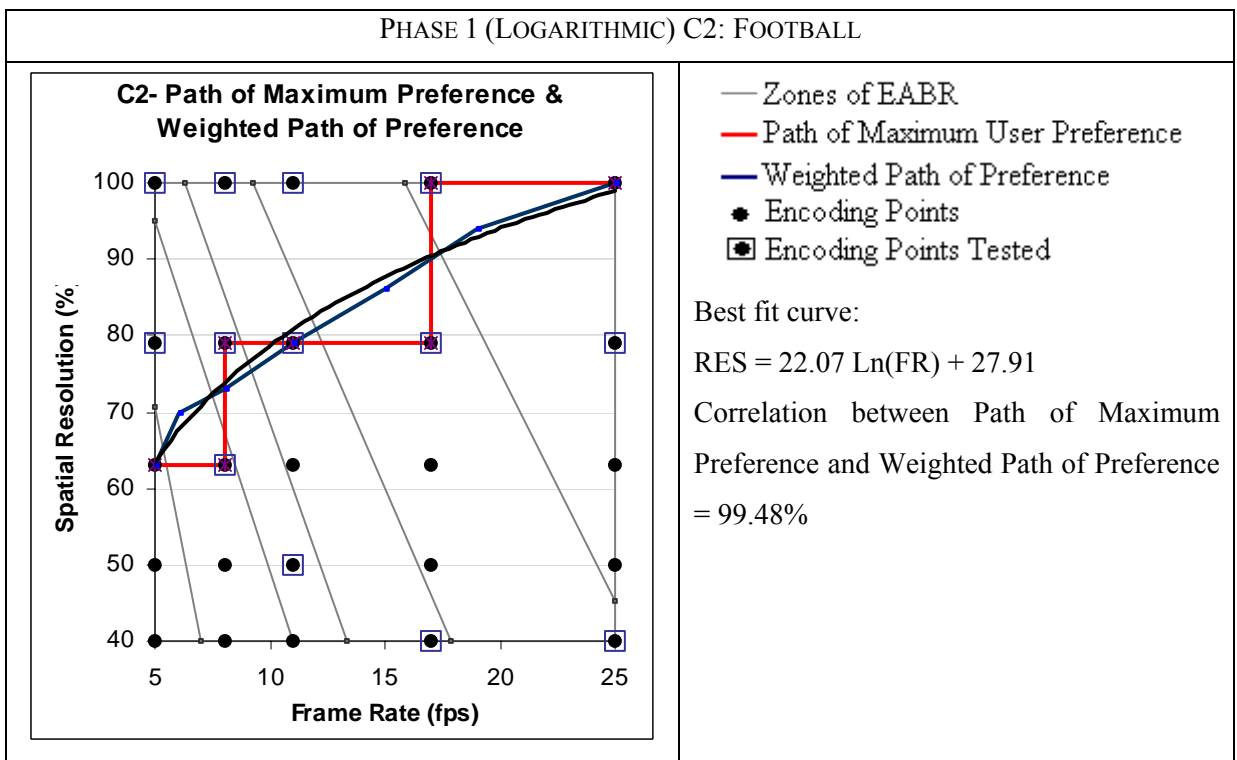


FIGURE 4.12: C2 RESULTS

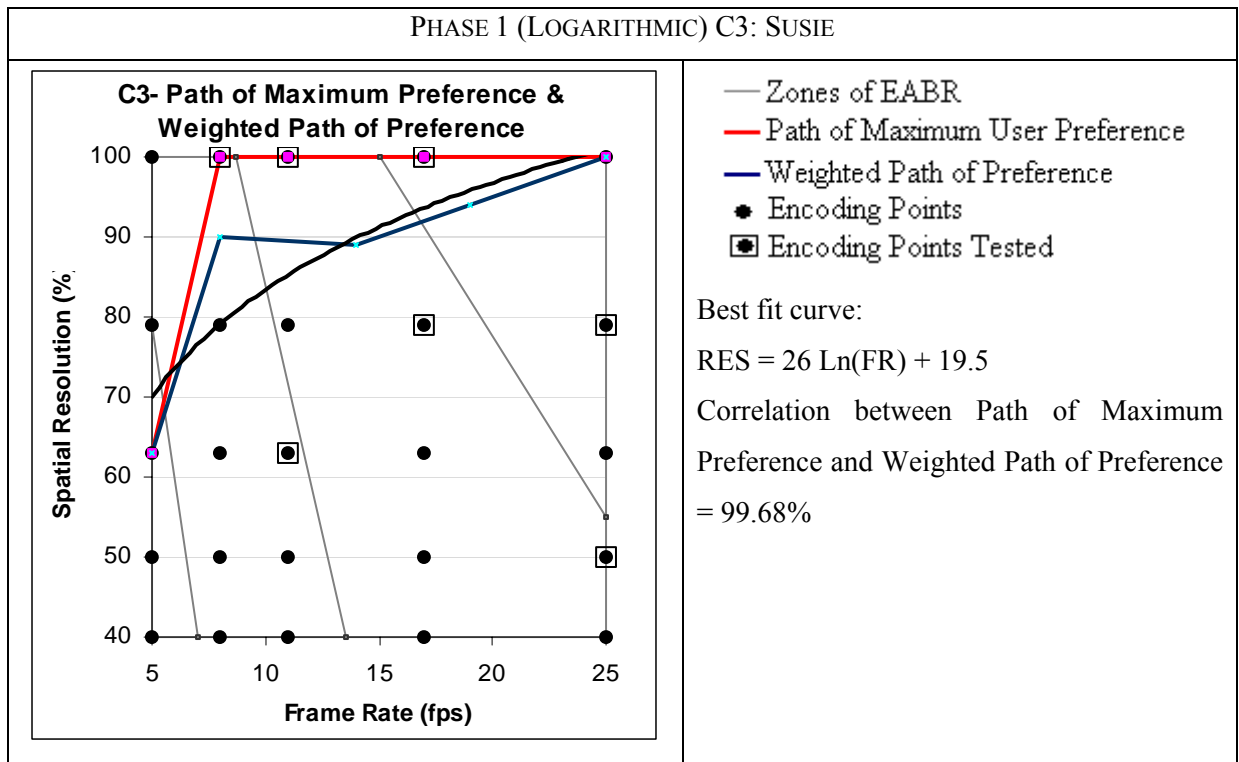


FIGURE 4.13: C3 RESULTS

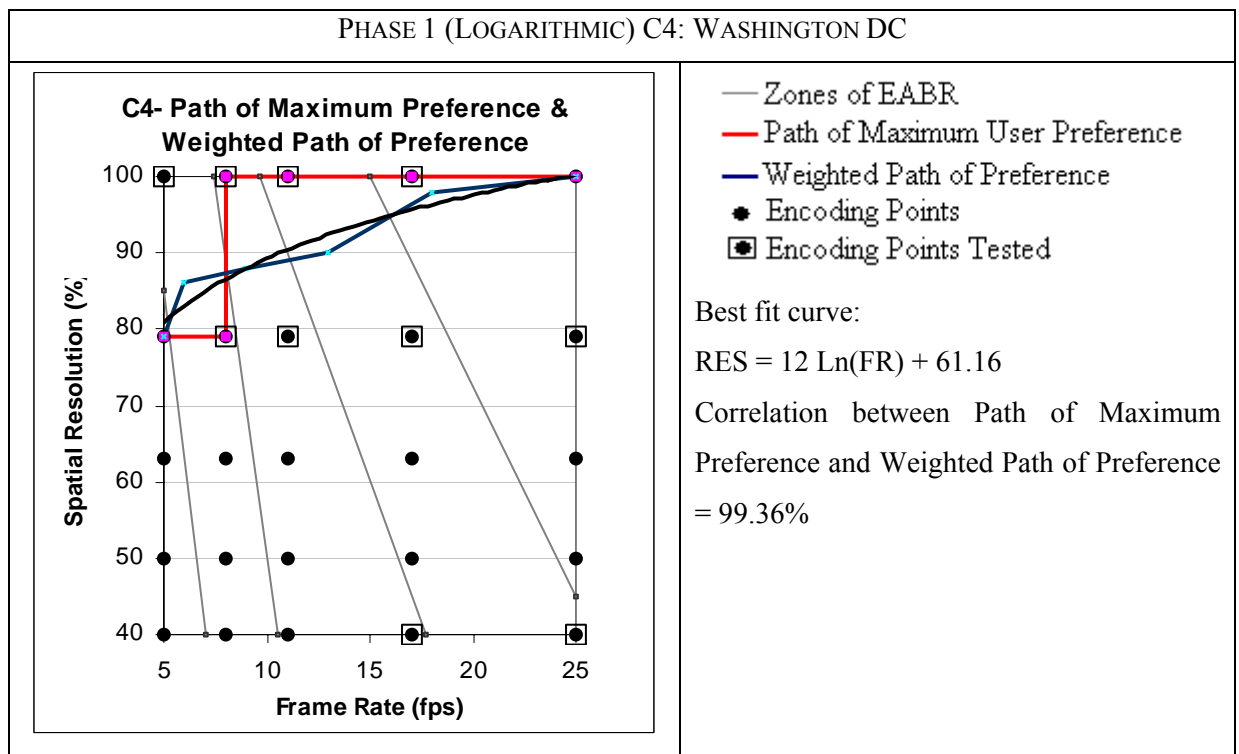


FIGURE 4.14: C4 RESULTS

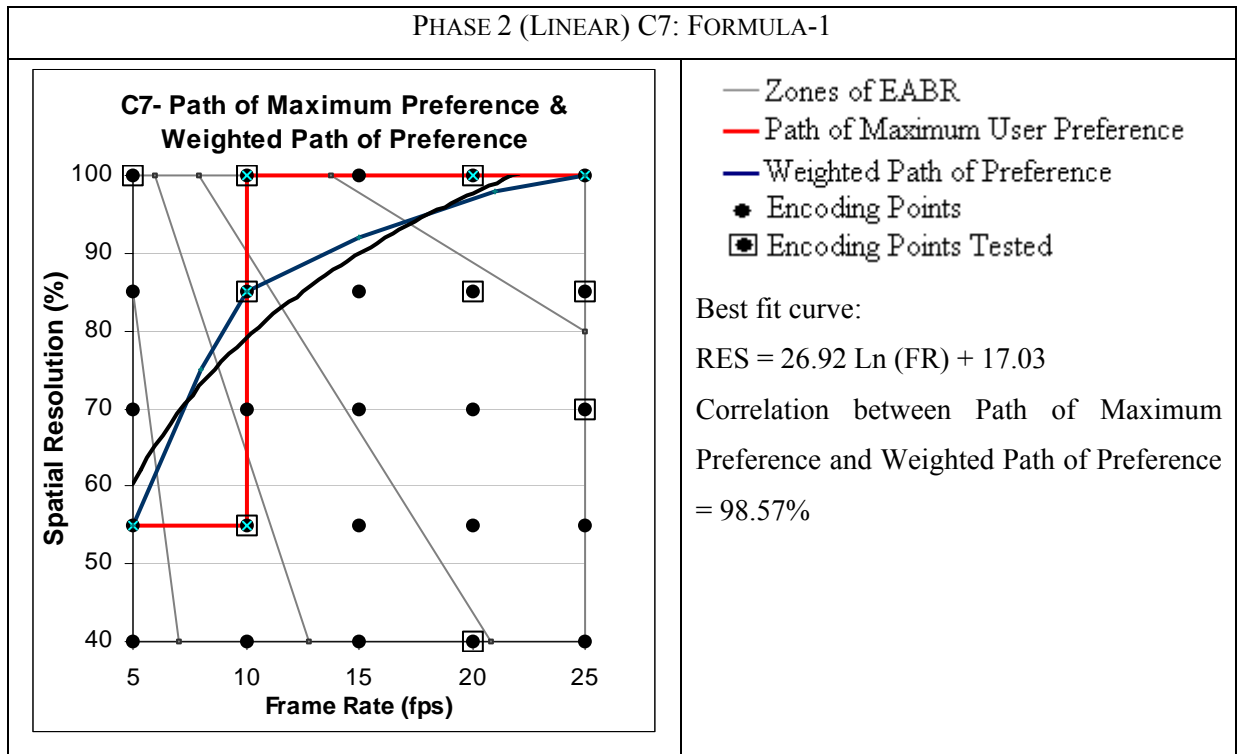


FIGURE 4.15: C7 RESULTS

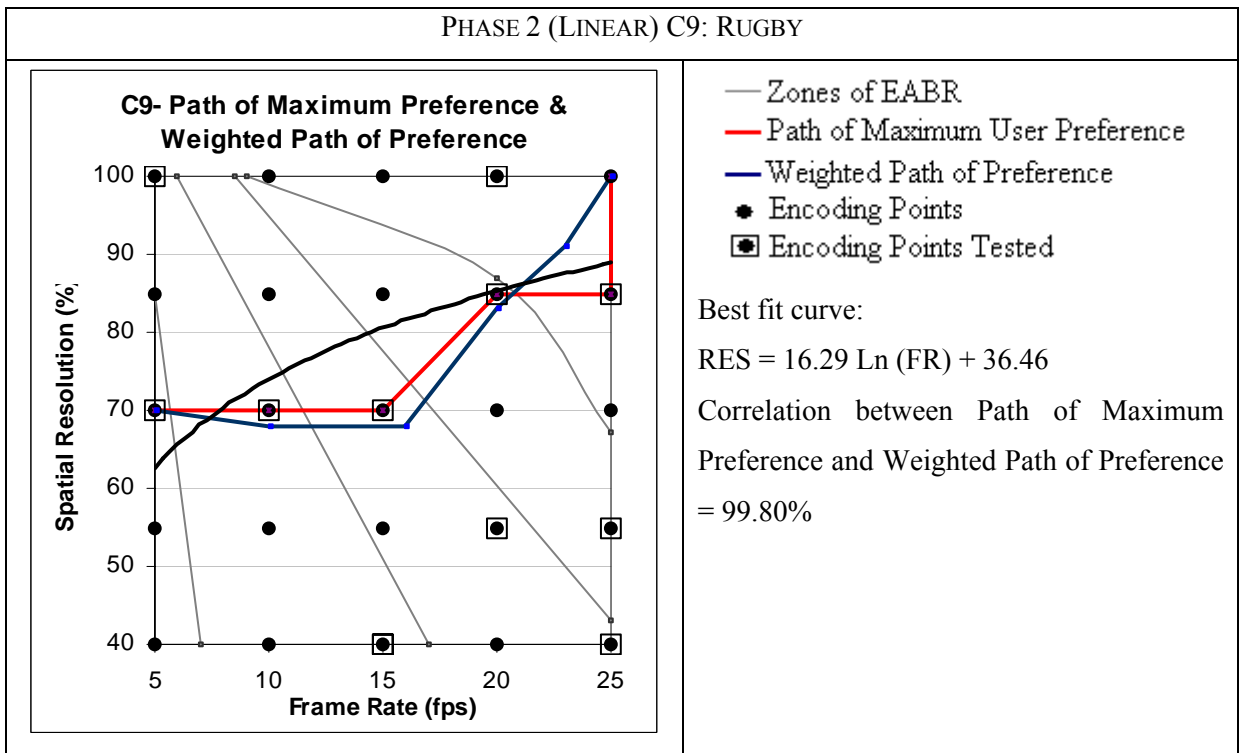


FIGURE 4.16: C9 RESULTS



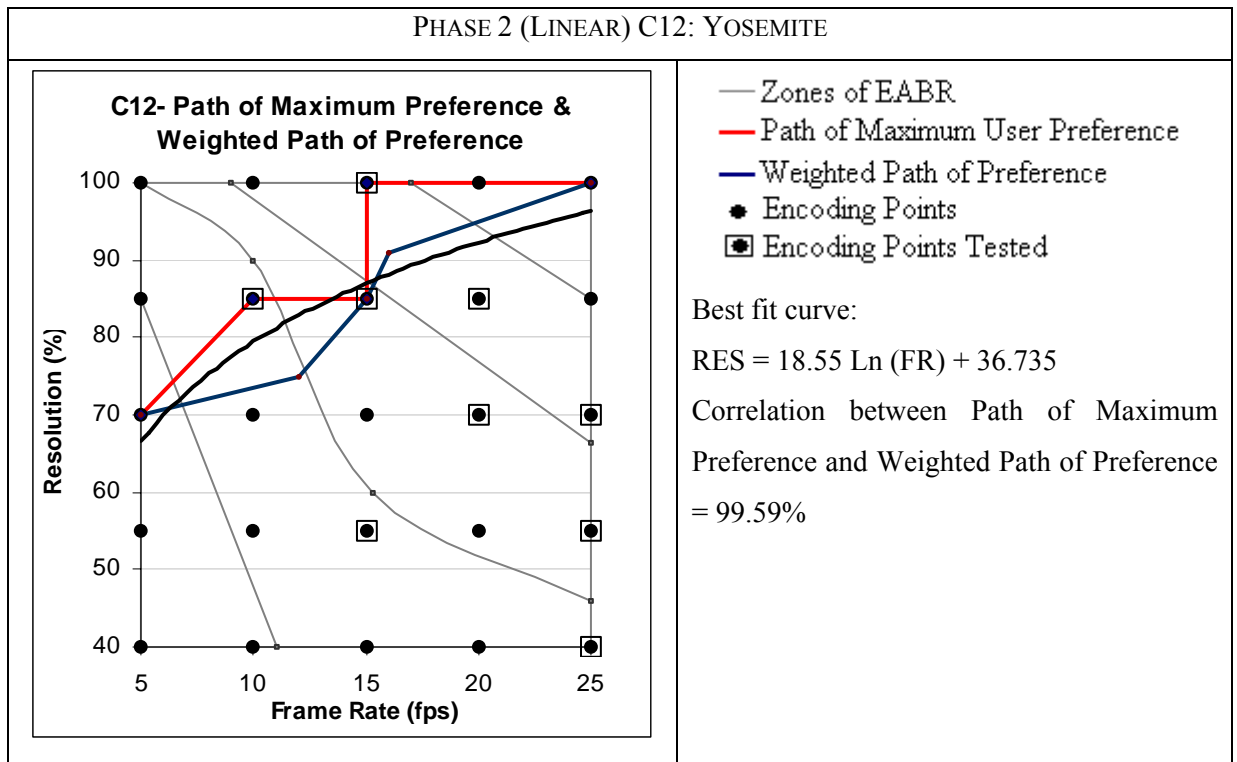


FIGURE 4.17: C12 RESULTS

#### 4.6.4. Discussion

During the statistical analysis, the forced choice methodology can be treated as a series of binomial experiments. The null hypothesis for binomial/multinomial experiments is that there is no preference for one encoding configuration over another, i.e. each encoding should have equal probability. The chi-square test with 95% confidence was performed to measure the degree of disagreement between the data and the null hypothesis. In most cases the chi-squared test indicated a high degree of disagreement with the null hypothesis, indicating a significant preference for one encoding configuration over another within a zone of EABR. Equally, if there is no preference for one encoding configuration over another, then the adaptation may be any encoding configuration within the tested configurations.

There are two valid paths of adaptation, using the path of maximum user preference and using the weighted path of preference. However, the validity of the interpolation to determine the weighted path of preference must be verified. In the next section, the interpolated encoding configurations are tested against both the encoding configurations of maximum user preference and alternative encoding configurations for two content types. It is expected that this will yield either no difference between the interpolated encoding configuration and the encoding configuration of maximum preference or else will show that the interpolated encoding configuration is indeed better and satisfies more users.

## 4.7. Interpolation Tests

The weighted path of preference is obtained by using the OAPs and their percentage preference. It was thought that the interpolated encoding configuration would satisfy more users and that the interpolated encoding configuration would have a greater percentage preference over the maximum preferred encoding configuration. To verify that this assumption is accurate and correct, the interpolation used had to be verified and validated. In the initial stages of subjective testing, the degree of preference of one clip over another was determined. Using this degree of preference, an optimal adaptation perception point (OAP) was calculated using linear interpolation. Ideally it would be desired that there would be a greater preference for the interpolated encoding configuration. There are two scenarios and acceptable outcomes:

- If there is a significant degree of preference between clips A and B in the initial tests, the interpolated point should have either no significant preference or greater preference from the maximum preferred encoding, A, and a significant degree of preference with the alternative encoding point, clip B.
- If on the other hand, there is little difference between the maximum preferred encoding, A, and the alternative encoding, B, then the preference for the interpolated point should have either greater preference or else not be significant from either clip A or clip B.

The forced choice methodology was employed to conduct this phase of subjective testing. The subject was shown a pair of clips, where one clip was encoded using the interpolated encoding configuration and the second was either the maximum preferred encoding or the next nearest encoding configuration within that zone of EABR. It was noted which choices were RCs. However, as the experiments progressed, subjects became more familiar with the test procedure and no longer mentioned which were forced choices. Therefore, to analyse the forced choices would be inaccurate.

### 4.7.1. Interpolation Test Results

The result of a forced choice experiment is a typical binomial random variable. The null hypothesis is that there is no preference for one clip over another, thus both clips should have a probability of 0.5. To measure the degree of preference and test for statistical significance, a Chi-square test was performed. The population sample size for this set of subjective testing was 27.

Two content types were selected to test the validity of interpolation, content 1 and content 3. The tests are conducted in pairs. For example, in the first zone of EABR for content 1 there are two test pairs, T1 and T2 (Figure 4.18). The first test, T1, uses the forced choice between the maximum preferred encoding and the interpolated encoding configuration (marked as the pink square). The second test, T2, uses the forced choice between the interpolated encoding and the nearest encoding configurations within that zone of EABR. The statistical significance of preference is calculated for each test case to determine whether the use of the interpolated encoding is better than, equal to or worse than any other encoding within that zone of EABR.

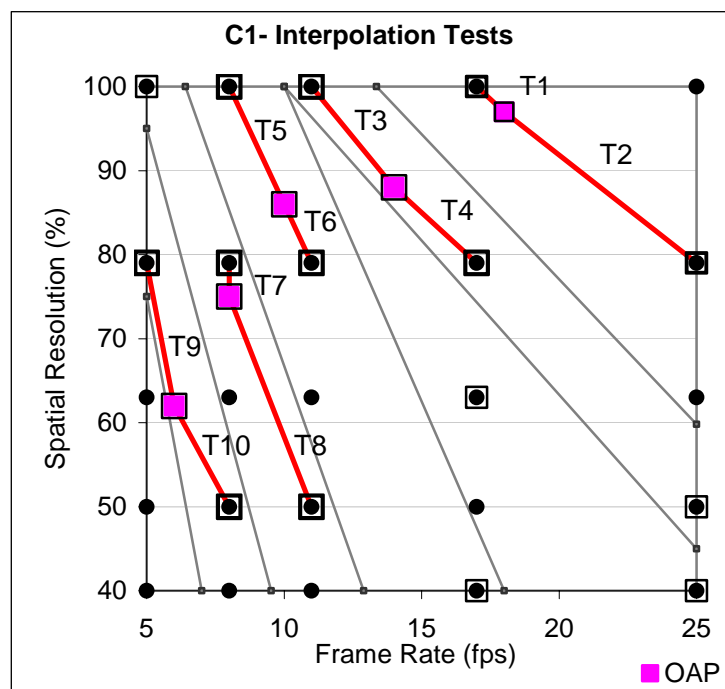


FIGURE 4.18: INTERPOLATION TESTS FOR C1

A full listing of the results from the interpolated tests appears in Volume 2 Appendix F. It was found that there are two classes of interpolation test. The first occurs when there is a statistical significance between the maximum preferred encoding and the next nearest encoding configuration within that zone of EABR (Table 4.10). The second occurs when there is no statistical significance between the maximum preferred encoding and the next nearest encoding configuration within that zone of EABR (Table 4.11).

Content 1 – Test 1 and Test 2										
Test	A		B		I		%A	%B	%I	ChiSq ( $\chi^2$ )
	Res (%)	Fr (fps)	Res (%)	Fr (fps)	Res (%)	Fr (fps)				
Original	100	17	79	25			85	15		13.37 Significance (>99.5%)
T1	100	17			97	18	52		48	0.037 Significance (<90%)
T2			79	25	97	18		19	81	10.7 Significance (>99.5%)

TABLE 4.10: INTERPOLATION RESULTS FOR CONTENT 1 – TEST 1 AND TEST 2

**Analysis:**

- The initial degree of preference is significant between clips A and B.
- The degree of preference between the maximum preferred encoding (A) and the interpolated point (I) is not significant.
- The degree of preference between the interpolated point (I) and clip B is significant.
- Therefore, it is acceptable to replace the maximum preferred encoding with the interpolated encoding configuration.

Content 1 – Test 3 and Test 4										
Test	A		B		I		%A	%B	%I	ChiSq ( $\chi^2$ )
	Res (%)	Fr (fps)	Res (%)	Fr (fps)	Res (%)	Fr (fps)				
Original	100	11	79	17			56	44		0.333 Significance (<90%)
T3	100	11			88	14	48		52	0.037 Significance (<90%)
T4			79	17	88	14		56	44	0.333 Significance (<90%)

TABLE 4.11: INTERPOLATION RESULTS FOR CONTENT 1 – TEST 3 AND TEST 4

**Analysis:**

- The initial degree of preference is not significant between clips A and B.
- The degree of preference between the maximum preferred encoding (A) and the interpolated point (I) is not significant.
- The degree of preference between the interpolated point (I) and clip B is not significant.
- Therefore, it is acceptable to replace the maximum preferred encoding with the interpolated encoding configuration.

**4.7.2. Discussion**

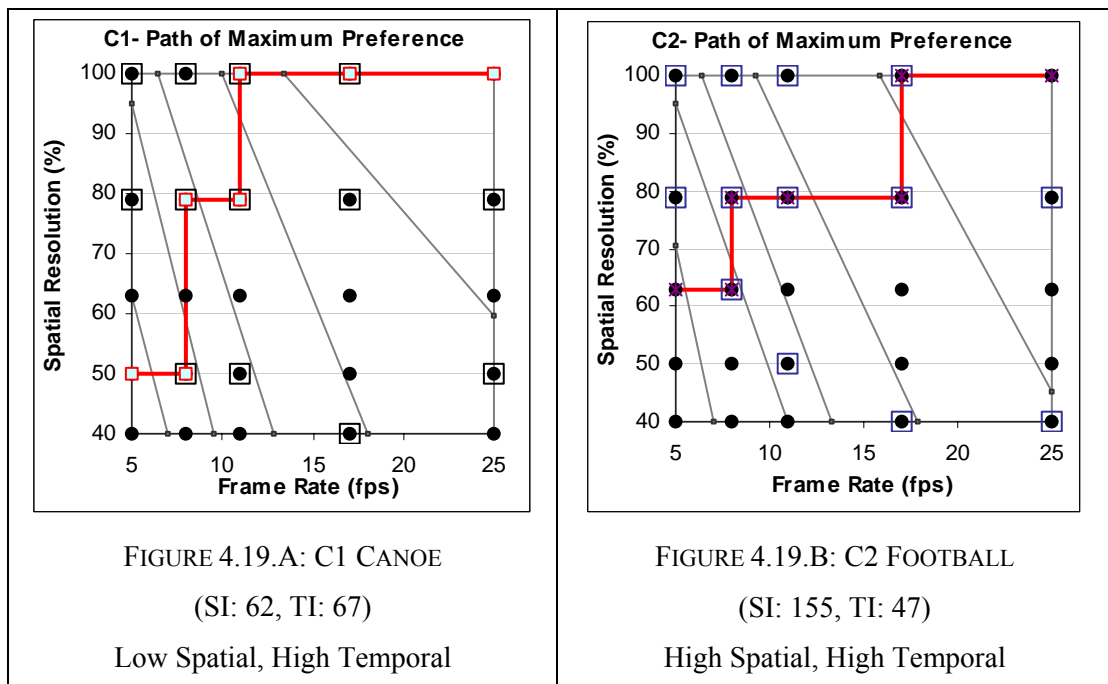
In all cases, the interpolated encoding did not have a statistically significant preference from the maximum preferred encoding indicating that this simple weighted vector approach is acceptable. This implies that either the path of maximum user preference and interpolated weighted path of preference can be used for adaptively streaming video. It was observed that there was a higher incidence of RCs when there was little difference between the maximum preferred encoding and the interpolated encoding.

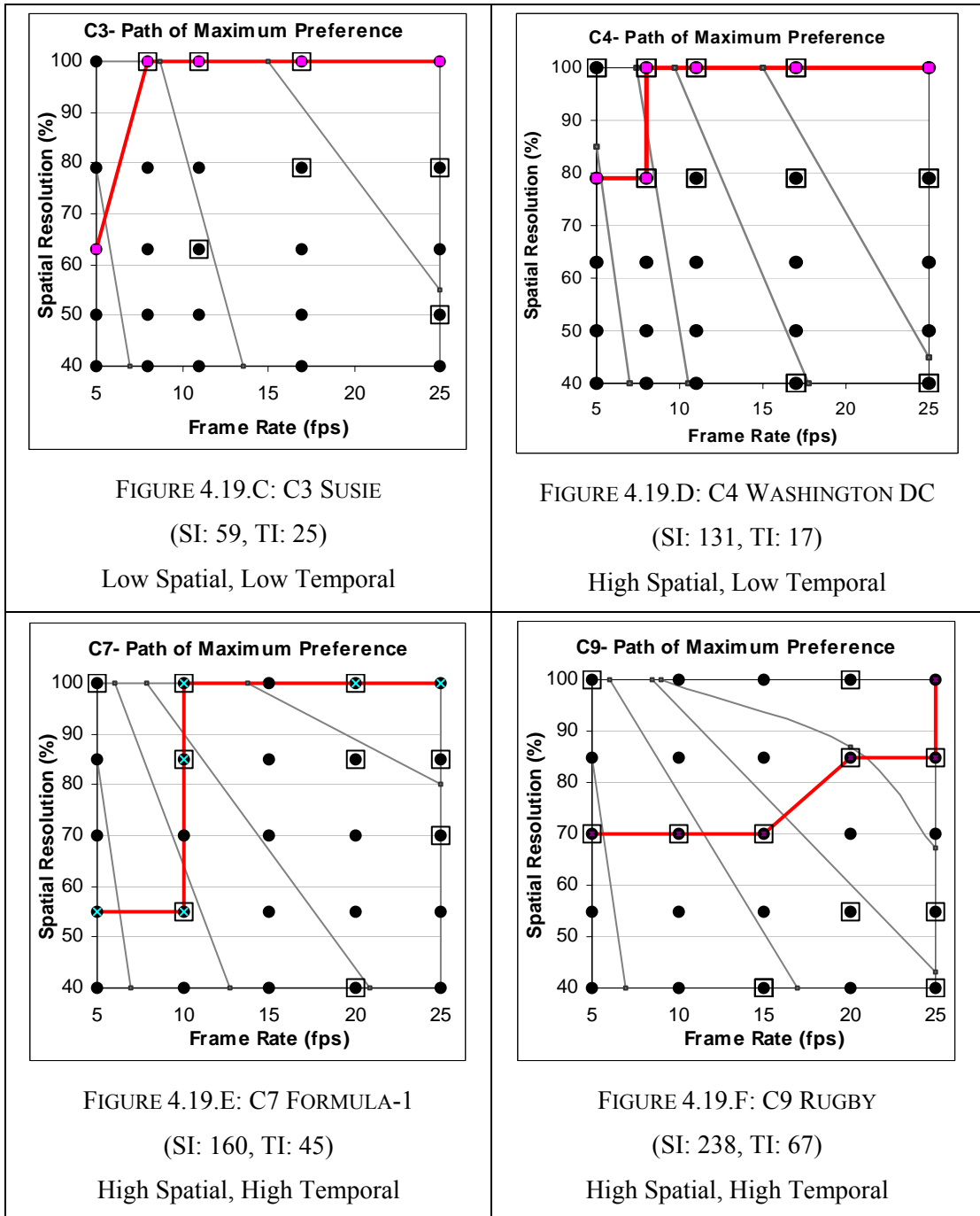
## 4.8. Analysis and Discussion of Results

From the interpolation results presented in the previous section, the interpolated encoding configuration is not statistically significant from using the maximum preferred encoding. This leaves two valid paths of adaptation, the path of maximum user preference and the interpolated, weighted path of preference.

### 4.8.1. Comparison of Paths of Maximum User Preference

From Figure 4.19.A/B/E/F, it can be clearly seen from the lines of maximum user preference that when there is high action (C1, C2, C7 and C9), the resolution is less dominant regardless of whether the clip has high spatial characteristics or not. This implies that the user is more sensitive to continuous motion when there is high temporal information in the video content. Intuitively this makes sense as, when there is high action in a scene, often the scene changes are too fast for the user to be able to assimilate the scene detail. Conversely, in Figure 4.19.C/D/G, when the scene has low temporal requirements (C3, C4, and C12), the resolution becomes more strongly dominant regardless of the spatial characteristics. There is markedly high diversity between the various paths of maximum user preference covering a large amount of adaptation space. By comparing the paths of maximum preference (Figure 4.19.H), it can be concluded that the users prefer a two-dimensional adaptation policy with a trade-off of spatial resolution against frame rate proportional to the contents characteristics.







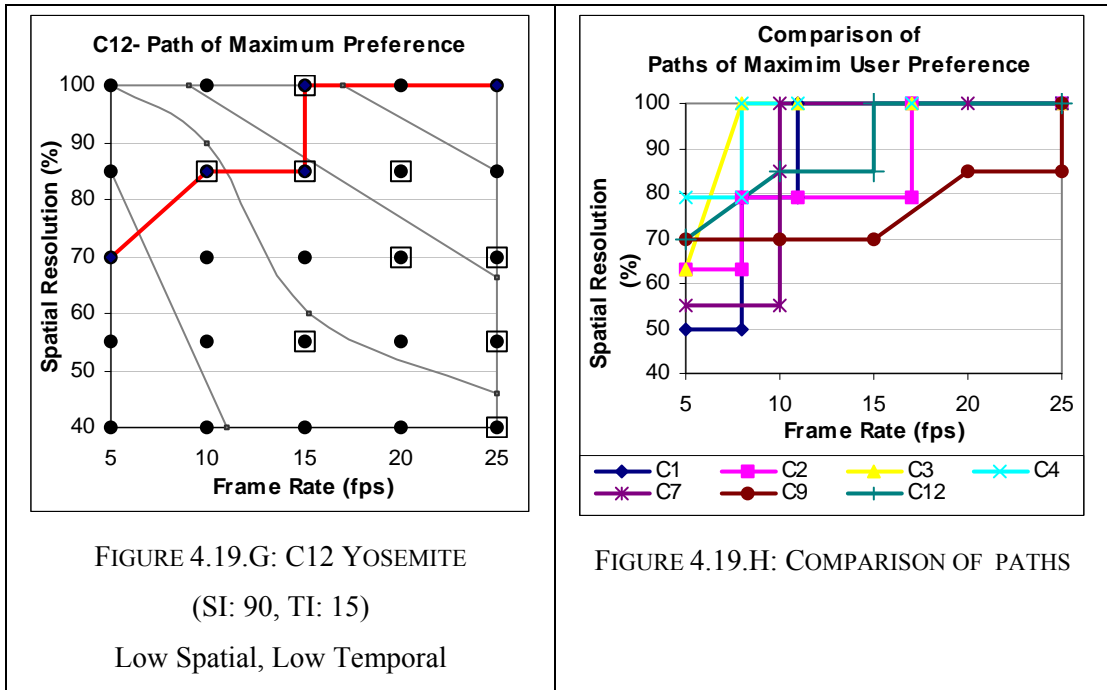


FIGURE 4.19.G: C12 YOSEMITE  
(SI: 90, TI: 15)  
Low Spatial, Low Temporal

FIGURE 4.19.H: COMPARISON OF PATHS

FIGURE 4.19: PATHS OF MAXIMUM PREFERENCE FOR ALL CONTENT TYPES

### 4.8.2. Comparison of Weighted Paths of Preference

By interpolating, the OAP encoding configurations were calculated. It was found that the use of the interpolated points were equally valid to using the maximum preferred encodings. The outlier is C4 (Figure 4.20.D), which is a slow moving highly detailed clip, leans closest towards the spatial resolution axis. Its OAT deviates strongly towards resolution, indicating that when degrading the quality of the video being delivered, the frame rate should be sacrificed for resolution for this particular content type. Conversely for high action clips, frame rate should be given higher precedence to ensure smooth continuous motion as can be seen in the other outlier is C9 (Figure 4.20.F) leans closest towards the frame rate axis. This clip has a high degree of detail and action with its SI-TI values on the extreme boundaries of the SI-TI grid. Due to the closeness of the paths of weighted preference, it indicates that the OAT is not particularly sensitive to the SI-TI values.

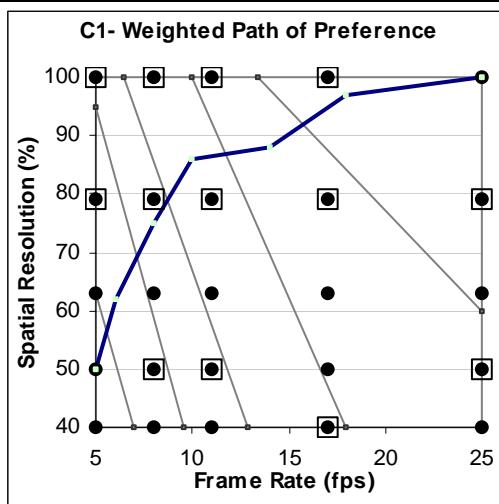


FIGURE 4.20.A: C1 CANOE  
(SI: 62, TI: 67)  
Low Spatial, High Temporal

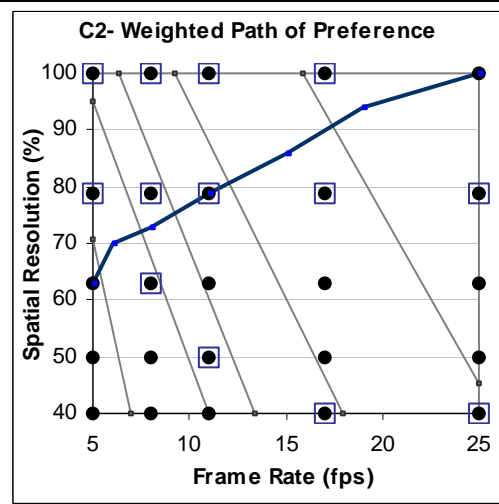


FIGURE 4.20.B: C2 FOOTBALL  
(SI: 155, TI: 47)  
High Spatial, High Temporal

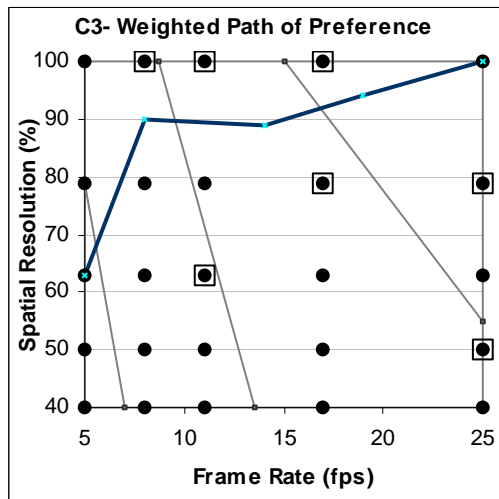


FIGURE 4.20.C: C3 SUSIE  
(SI: 59, TI: 25)  
Low Spatial, Low Temporal

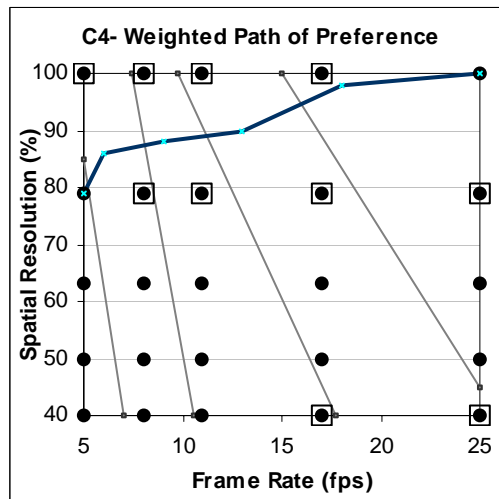


FIGURE 4.20.D: C4 WASHINGTON DC  
(SI: 131, TI: 17)  
High Spatial, Low Temporal

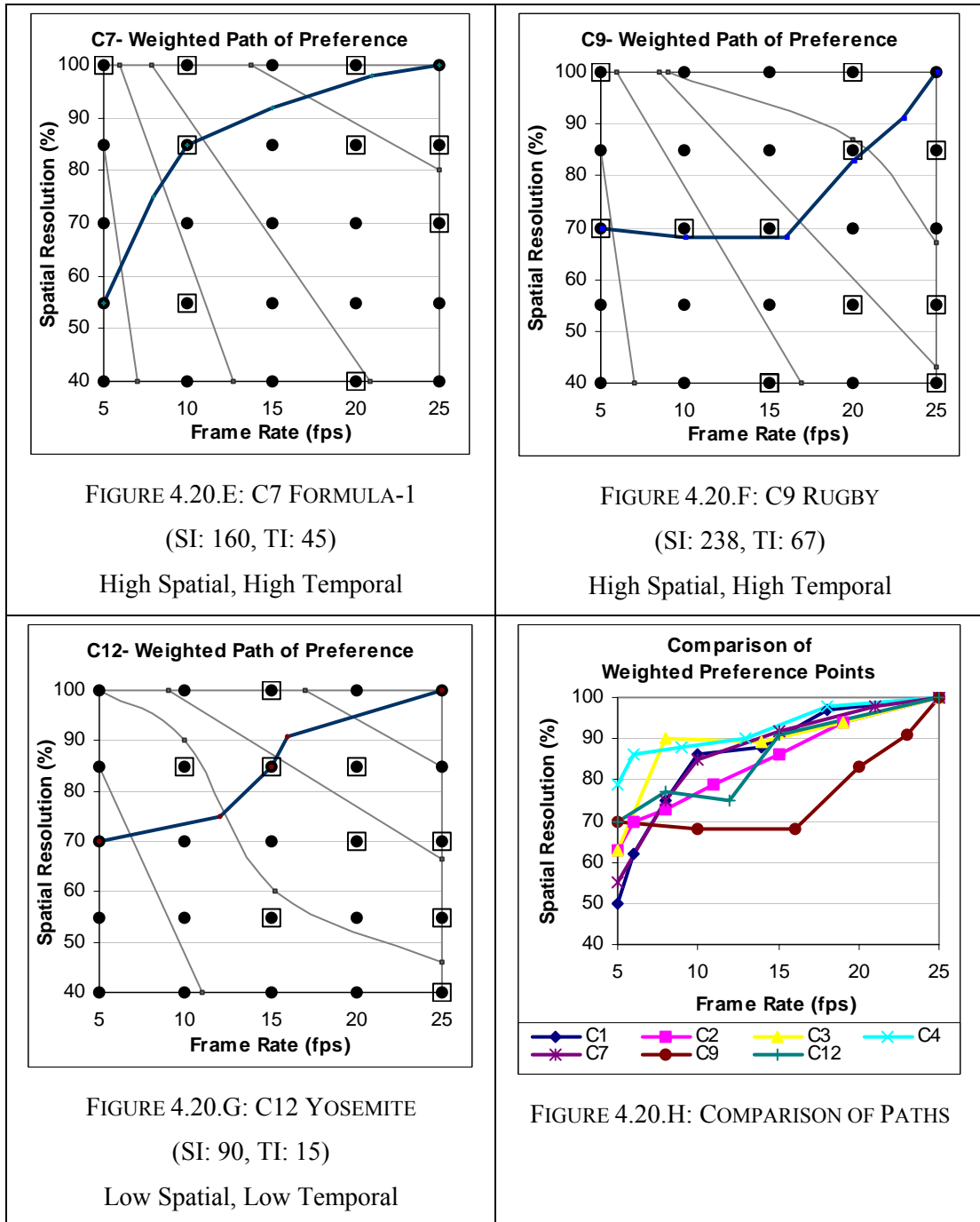


FIGURE 4.20.E: C7 FORMULA-1  
(SI: 160, TI: 45)  
High Spatial, High Temporal

FIGURE 4.20.F: C9 RUGBY  
(SI: 238, TI: 67)  
High Spatial, High Temporal

FIGURE 4.20.G: C12 YOSEMITE  
(SI: 90, TI: 15)  
Low Spatial, Low Temporal

FIGURE 4.20.H: COMPARISON OF PATHS

FIGURE 4.20: WEIGHTED PATH OF PREFERENCE FOR ALL CONTENT TYPES

### 4.8.3. Globally Averaged OAT

The usefulness of the OAT is dependent on the contents’ spatial and temporal characteristics being known a priori. The SI and TI parameters are highly computationally intensive taking around 5 minutes to calculate SI parameter alone for a 10 second clip and 1 minute to calculate the TI parameter. For a real-time encoded clip, this calculation in real-time is not realistic or feasible. To adapt in a real-time streaming scenario using the OAT where the contents characteristics are not known beforehand, there needs to be some way to either

approximate the contents characteristics or apply a globally averaged OAT.

To determine the globally averaged OAT, both the maximum preferred encoding configurations and interpolated encoding configurations for all content types are analysed together and plotted on a scattergram. A least squares fit curve is plotted for both the maximum preferred encodings and the interpolated encodings. A least squares line has one of the following properties.

1. The sum of errors (SE) is equal to zero.
2. The sum of squared errors (SSE) is smaller than that for any other line.

$y_p$  = The predicted value of  $y$ , using the least squares fit line.

$$SSE = \sum (y_i - y_p)^2$$

By measuring how much the data deviates from this line (i.e., the magnitude of deviations between the observed and predicted values of resolution from the least squares line, in other words, the vertical distances between the observed and predicted values), it can be determined how well the data fits the least squares line. There are two best-fit curve lines, one for the maximum preferred encoding configurations for all content types and one for the interpolated encoding configurations for all content types (Figure 4.21). It was interesting that when a least squares fit line was calculated for the interpolated encoding configurations across all content types and the maximum encoding configurations across all content types, these yielded very similar curves with a high degree of correlation (Table 4.12). The Pearson product moment coefficient of correlation  $r$ , measures the relationship between data sets. If  $r$  has a value of  $-1$ , it implies that there is a perfect negative relationship. If  $r$  has a value of  $0$ , there is little or no relationship. If  $r$  has a value of  $+1$ , there is a perfect positive relationship between the observed data and the predicted data by the least squares fit line.

SS = Least squares prediction line equation.

$$SS_{XY} = \sum (x_i - x_p)(y_i - y_p) = \sum x_i y_i - \frac{(\sum x_i)(\sum y_i)}{n}$$

$$SS_{XX} = \sum (x_i - x_p)^2 = \sum x_i^2 - \frac{(\sum x_i)^2}{n}$$

$$SS_{YY} = \sum (y_i - y_p)^2 = \sum y_i^2 - \frac{(\sum y_i)^2}{n}$$

$$r = \frac{SS_{XY}}{\sqrt{SS_{XX} SS_{YY}}}$$

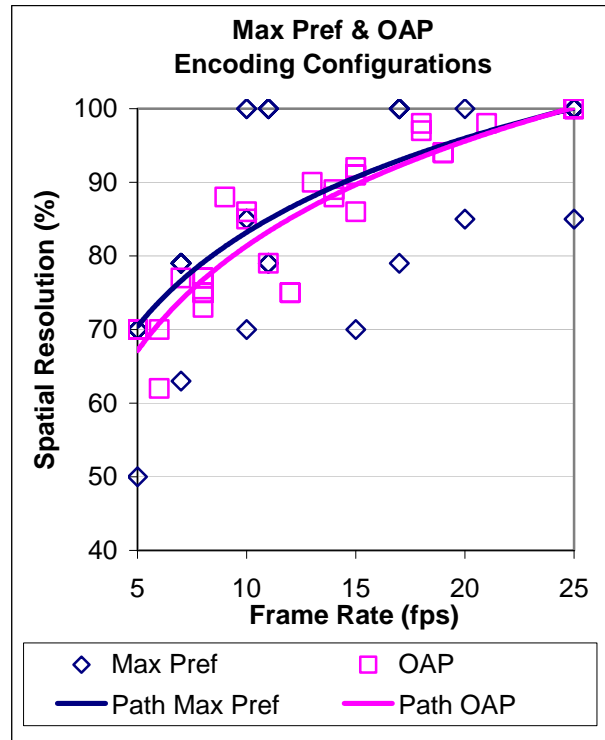


FIGURE 4.21: GLOBALLY AVERAGED PATHS OF MAXIMUM USER PREFERENCE AND WEIGHTED PATH OF PREFERENCE

	<b>Max Preference</b>	<b>OAP</b>
Average absolute deviation of observed from mean	7.36	2.87
Standard deviation of observed from mean	6.87	2.89
SSE of observed from mean	9367.72	7283.76
Correlation	98%	99.7%
Pearson coefficient	98%	100%
Path of OAP: $\text{Res} = 21 \text{ Ln}(\text{Fr}) + 34$ Path of Maximum Preference: $\text{Res} = 18 \text{ Ln}(\text{Fr}) + 41$		

TABLE 4.12: GLOBALLY AVERAGED OAT RESULTS

The correlation between the predicted max preference encoding least squares equation and OAP encoding least squares equation = 0.99979

The Pearson correlation between the predicted max preference encoding least squares

equation and OAP encoding least squares equation = 0.99978

There is a high correlation between the least squares fit line of the path of maximum preference and the least squares fit line of the weighted path of preference. The globally averaged OAT is the OAT that includes all results for the maximum preferred encoding configurations and interpolated encoding configurations for all content types covering the Spatial-Temporal grid.

## 4.9. Summary and Conclusions

This chapter proposed that there is an optimal way in which multimedia transmissions should be adapted in response to network conditions to maximize the user-perceived quality. This is based on the hypothesis that within the set of different ways to achieve a target bit rate, there exists an encoding that maximizes the user-perceived quality. Extensive subjective testing suggests that an optimum adaptation trajectory (OAT) in the space of possible encodings does exist and that it is related to the contents characteristics.

The subjective testing consisted of two independent testers performing identical test procedures and test sequences on subjects taken from a student population ranging in ages from 18-30. The correlation between the results from independent testers was greater than 94% across all content types tested indicating that the results of the testers can be combined.

The subjective tests were conducted in two phases. Phase 1 considered 4 test sequences, one taken from each quadrant of the SI-TI grid. Phase 2 considered 4 different test sequences with similar SI-TI values to those used for Phase 1. However, this time, the adaptation space was sampled using a linear scale. The main objective of having two different test phases was to verify and validate the results from Phase 1. In addition, using different encoding scales, it could be ascertained that the OAT was similar in shape regardless of whether a linear or logarithmic scale was used, and regardless of the encoding points tested. The clips falling within a particular zone of EABR have different (but similar) bit rates. By using zones of EABR, the bit rate of different test sequences with different encoding configurations is effectively quantised which in turn dramatically reduces the number of test cases.

During statistical analysis, the forced choice methodology can be considered as a series of binomial experiments. Depending on the number of clips falling into each zone of EABR, the analysis was either binomial or multinomial. Each possible combination of sequences within a zone of EABR was tested using the forced choice methodology. The null hypothesis for binomial/multinomial experiments is that there is no preference for one encoding configuration over another, thus each encoding should have equal probability. The chi-square test with 95% confidence was performed to measure the degree of disagreement between the data and the null hypothesis. In most cases the chi-squared test indicated significant preference for one encoding configuration over another within a zone of EABR. Equally, if there is no preference for one encoding configuration over another, then the adaptation may be any encoding configuration within the tested configurations.

There are two valid paths of adaptation, using the path of maximum user preference and using the weighted path of preference. Through further subjective testing using the forced choice methodology it was found that in all cases, the interpolated encoding did not have a statistically significant preference from the maximum preferred encoding indicating that this simple weighted vector approach is acceptable. This implies that the path of maximum user preference and interpolated weighted path of preference can both be used.

From the both the path of maximum preference and the weighted path of preference, it can be clearly seen that when there is high action, the resolution is less dominant regardless of whether the clip has high spatial characteristics or not. This implies that the user is more sensitive to continuous motion when there is high temporal information in the video content. Intuitively this makes sense as, when there is high action in a scene, often the scene changes are too fast for the user to be able to assimilate the scene detail. Conversely, when the scene has low temporal requirements, the resolution becomes more strongly dominant regardless of the spatial characteristics. Thus the resulting paths are dependent on the contents characteristics. However, they are not sensitive to the actual spatial and temporal complexity values, thus a crude classification of the contents characteristics may be sufficient to determine the appropriate path of adaptation. It can be concluded that the users prefer a two-dimensional adaptation policy with a trade-off of spatial resolution against frame rate proportional to the contents characteristics.

In the next chapter, the OAT discovered by subjective methods will be compared against those using objective metrics. The same methods used to discover these paths through subjective testing will be applied to several objective metrics in an attempt to discover an automated objective means to discover the OATs. The overall usefulness of adaptation in two-dimensions using the OAT is compared against adaptation using a single-dimension of adaptation, which is typical of adaptation policies currently used.



# Chapter 5

## Validation of the OAT

### *Chapter Abstract*

In the previous chapter the concept of an Optimum Adaptation Trajectory was presented. Extensive subjective testing demonstrated its existence. Objective metrics were initially disregarded due to the large number of objective metrics available. Objective metrics of video quality are not satisfactory in quantifying human perception and as such there are no recommendations or guidelines indicating which should be used. Further, it can be argued that to date, objective metrics were not designed to assess the quality of an adapting video stream. In this chapter, the possibility of discovering the OAT using objective metrics is addressed. Finally, the usefulness of the OAT in an adaptive streaming system is investigated.

Section 1 compares the results obtained using subjective means with results obtained using several sophisticated objective metrics. In section 2, the overall usefulness of adaptation in two-dimensions using the OAT is compared against adaptation using a single-dimension of adaptation. Finally section 3 summarises this chapter and makes some conclusions.

### 5.1. Using Objective Metrics for OAT discovery

In general, the objective metrics discussed in Chapter 3 Section 4, are very complicated and computationally difficult. To investigate the possibility of discovering the OAT using different objective metrics, two metrics were selected: the PSNR and VQM. The metrics were chosen for practical reasons.

- The PSNR is easy to calculate and compute. It is also widely used in adaptive video streaming systems as a measure of video quality.
- The VQM metric from National Telecommunications and Information Administration (NTIA) has a freely available software implementation, which performs the sophisticated VQM analysis and calculations.

#### **5.1.1. OAT Discovery using PSNR**

The most commonly used objective metric of video quality assessment is the Peak Signal to

Noise Ratio (PSNR), and has been widely used in many applications and adaptation algorithms to assess video quality. The advantage of PSNR is that it is very easy to compute. However, PSNR does not match well to the characteristics of HVS. The PSNR metric is computed only using the luminance signal. An 8-bit image will contain pixel luminance values that vary from 0 (black) to 255 (white). Typically PSNR values vary between 20 and 40 dB. The PSNR is defined according to the following formula:

$m = \text{row.}$

$n = \text{column.}$

$d(m,n) = \text{degraded pixel value at p, and (m, n).}$

$o(m,n) = \text{original pixel value at p, and (m, n).}$

MSE = Mean Square Error is the mean difference overall pixels.

RMSE = Square Root MSE.

$$PSNR = 20 \log_{10} \left( \frac{255}{RMSE} \right)$$

$$MSE = \frac{1}{NM} \sum_{m,n} [o(m,n) - d(m,n)]^2$$

The main problem with using PSNR values as a quality assessment method is that even though two images are different, the visibility of this difference is not considered. The PSNR metric does not take the visual masking phenomenon or any aspects of the HVS into consideration, that is, every single errored pixel contributes to the decrease of the PSNR, even if this error is not perceived. For example, the MSE can be produced in a number of different ways. That is, consider an image where the pixel values have been altered slightly over the entire image and an image where there is a concentrated alteration in a small part of the image, both will result in the same MSE value however, one will be more perceptible to the user than the other.

The PSNR is calculated comparing each degraded clip with the reference clip at a frame rate of 25fps and 100% spatial resolution. This will result in a 3D plane with the axes of PSNR, spatial resolution and frame rate (Figure 5.1). The clips were then sorted into zones of Equal Average Bit Rate (EABR). The path of maximum VQM was obtained by selecting the encoding configuration within each zone of EABR that yielded the highest PSNR value (Figure 5.2).

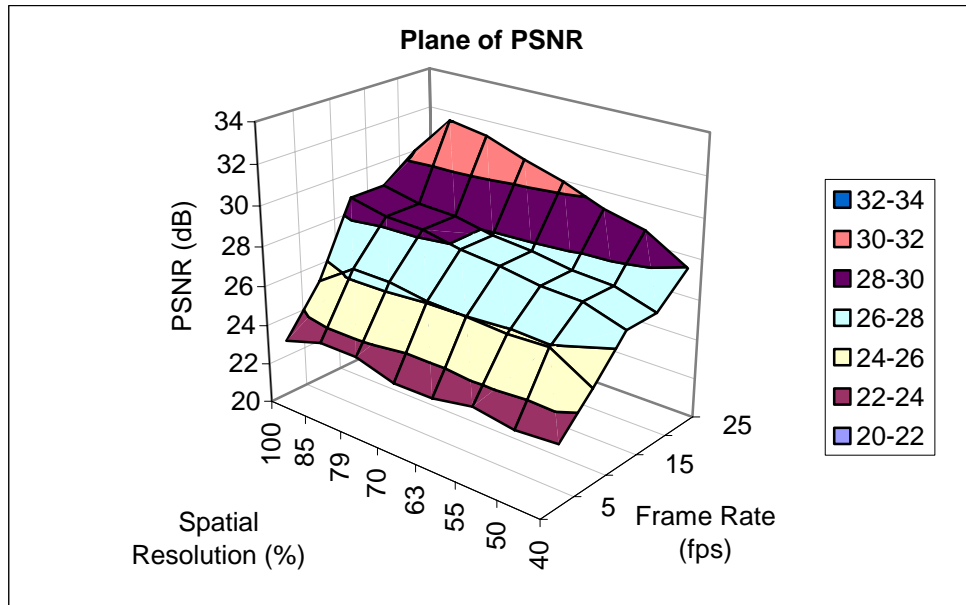


FIGURE 5.1: PLANE OF PSNR

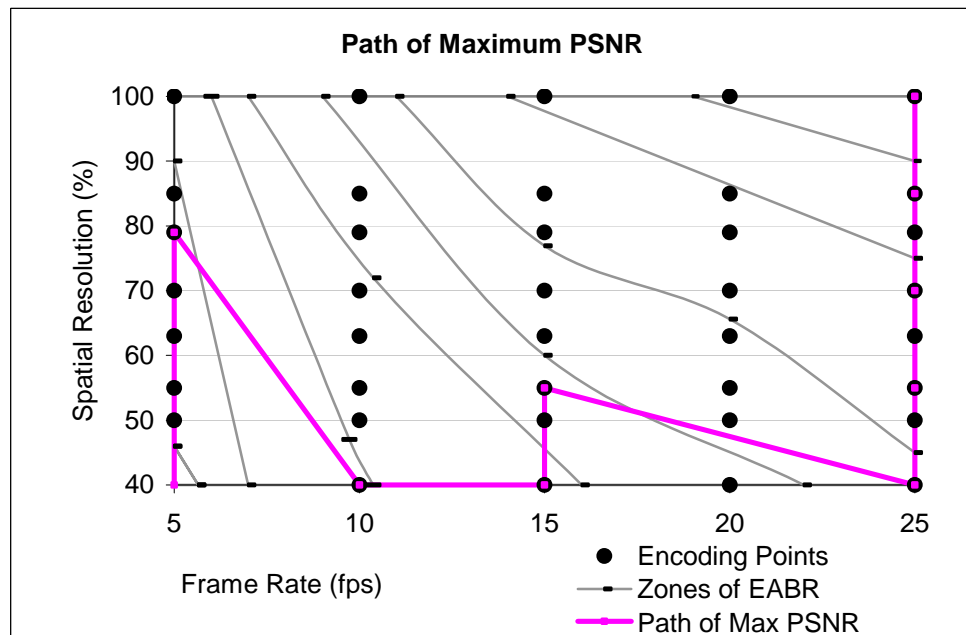


FIGURE 5.2: PATH OF MAXIMUM PSNR

### 5.1.2. OAT Discovery using VQM Metrics

The VQM Metric provides a means of objectively evaluating video quality. The software compares an original video clip and a processed video clip and reports a Video Quality Metric (VQM) that correlates to the perception of a typical end user. The VQM objective metrics are claimed to provide close approximations to the overall quality impressions, or mean opinion scores, of digital video impairments [161] [162] [163]. The quality measurement process includes sampling of the original and processed video streams,

calibration of the original and processed video streams, extraction of perception-based features, computation of video quality parameters, and calculation of VQM models (Figure 5.3).

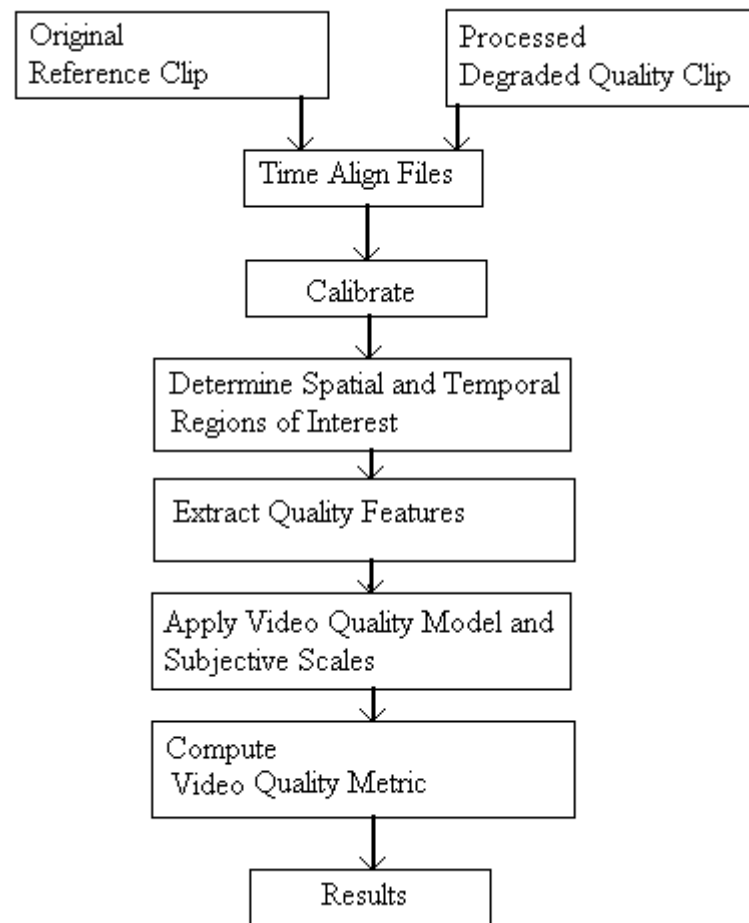


FIGURE 5.3: VQM PROCESSING MODEL

### *Video Quality Models (VQM)*

There are a number of quality models that can be applied to the video clips. Each model calculates the video quality in a different way, taking different aspects of the HVS into consideration. These are,

- **General model (VQM<sub>GEN</sub>):** provides a set of parameters and parameter weightings that allow the user to accurately evaluate video quality over a wide range of quality and bit rates.
- **Developer model (VQM<sub>DEV</sub>):** is a faster version of the General Model, which trades off accuracy for higher processing speed.
- **PSNR model (VQM<sub>PSNR</sub>):** uses the traditional measurement of PSNR. PSNR can also be used to determine if any lossy compression has been performed on the original video.
- **Video-conferencing model (VQM<sub>V-CONF</sub>):** is specifically optimized for the lower video

quality space found in most videoconferencing applications (targeting bit rates from 10 kbps to 1.5 Mbps).

- **Television model ( $VQM_T$ ):** is specifically optimized for the high video quality space found in television video. The television model has objective parameters for measuring the perceptual effects of the usual types of television impairments including blurring, block distortion, noise (in both the luminance and chrominance channels), and error blocks.

### 5.1.3. Methodology for OAT Discovery using VQM Metrics

To discover the OAT using the VQM metrics, each of the degraded clips was compared against the reference clip at a frame rate of 25fps and 100% spatial resolution.

Reference clip: frame rate of 25fps and a spatial resolution of 100%.

Degraded clips: logarithmic and linear frame rate range {5, 8, 10, 11, 15, 17, 20, 25}fps and spatial resolution range {40, 50, 55, 63, 70, 79, 85, 100}%

The VQM metric was calculated for each processed clip against the reference clip. The clips were then sorted into zones of Equal Average Bit Rate (EABR). The path of maximum VQM was obtained by selecting the encoding configuration within each zone of EABR that yielded the highest quality value using the VQM metric. This methodology was applied to each of the VQM Metrics,  $VQM_{GEN}$ ,  $VQM_{DEV}$ ,  $VQM_{V-CONF}$ , and  $VQM_{PSNR}$ . To demonstrate this methodology, the  $VQM_{PSNR}$  is shown.

### 5.1.4. PSNR model ( $VQM_{PSNR}$ )

The  $VQM_{PSNR}$  model uses the peak-signal-to-noise-ratio (PSNR) measurement. Weights are applied to map the PSNR value and to correlate it with results of subjective testing. Generally the PSNR is calculated using a value of 255, but in the calculation of  $VQM_{PSNR}$  a value of 235 is used to avoid dividing by zero when comparing identical images. Figure 5.4 shows a 3-dimensional plane of  $VQM_{PSNR}$  and Figure 5.5 shows the path of maximum  $VQM_{PSNR}$  through the zones of EABR followed by the equation used to calculate this metric.

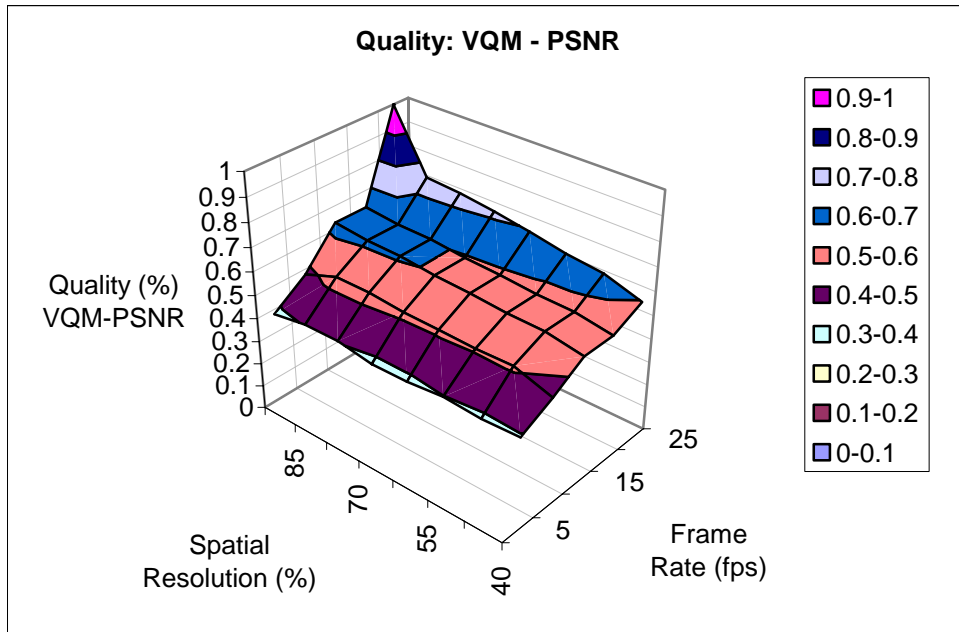
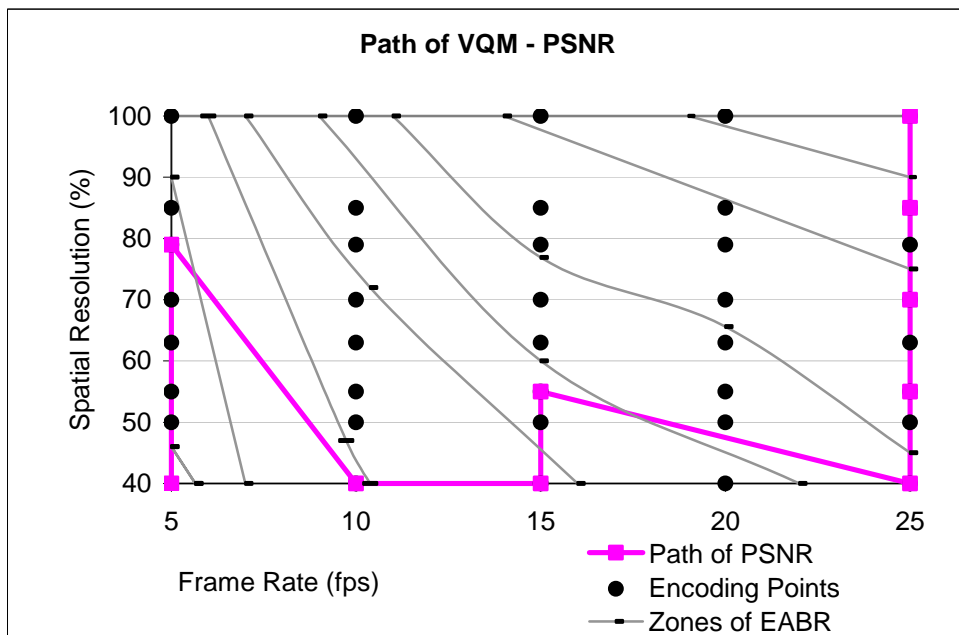


FIGURE 5.4: PLANE OF VQM PSNR



$$VQM_{PSNR} = \frac{1}{1 + \exp^{0.171(PSNR - 25.6675)}}$$

FIGURE 5.5: PATH OF MAXIMUM VQM PSNR

### 5.1.5. Results

The paths of maximum VQM using the different models are very similar and in the case of the  $VQM_{PSNR}$  and  $VQM_{GEN}$ , identical (Figure 5.6). There seems to be a tendency by each of the VQM metrics to maintain the frame rate.

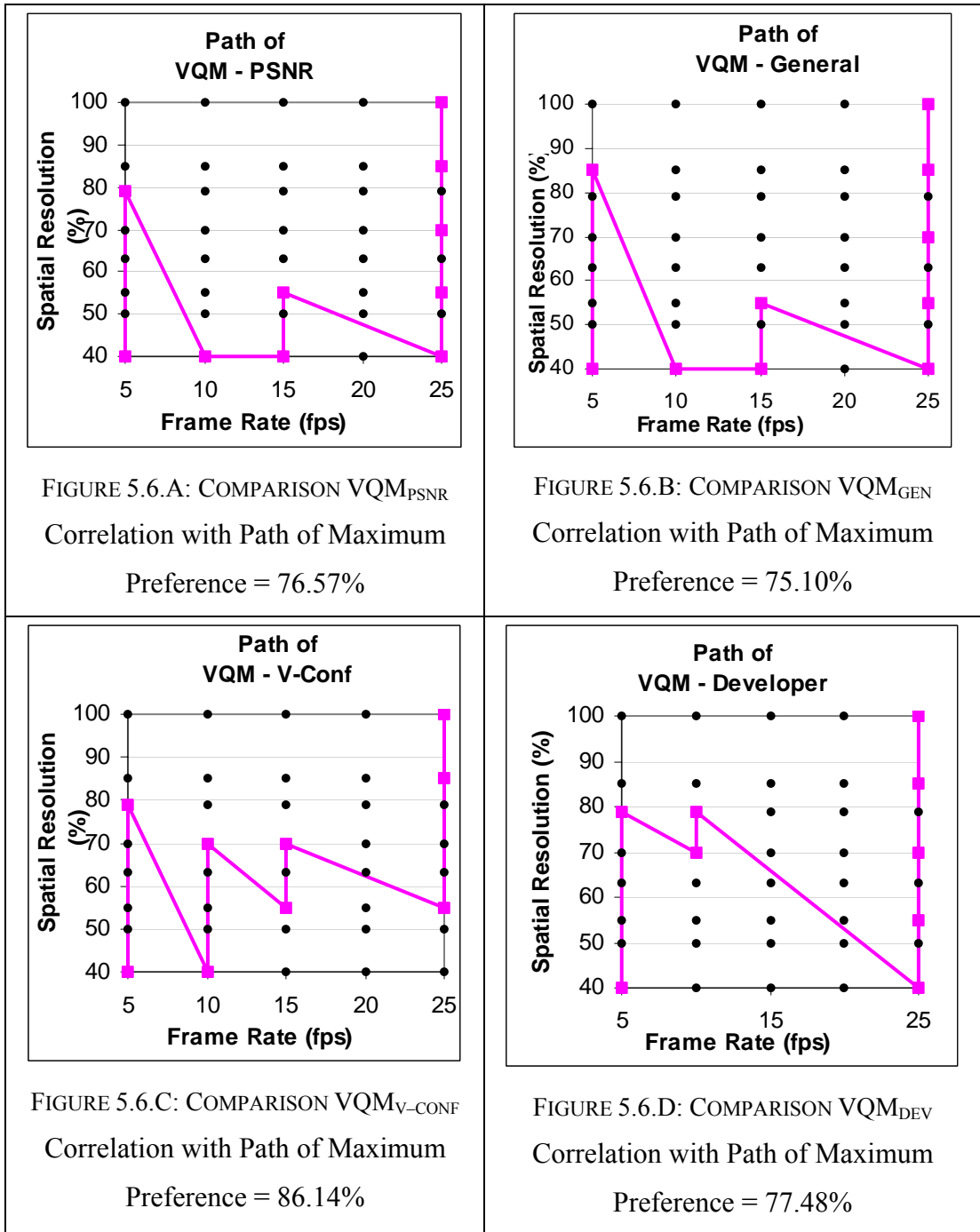


FIGURE 5.6: PATHS OF MAXIMUM VQM QUALITY

### 5.1.6. Discussion

The paths of maximum VQM using the different models are very similar and in the case of the  $VQM_{PSNR}$  and  $VQM_{GEN}$ , identical (Figure 5.6.A and Figure 5.6.B). There seems to be a tendency by each of the VQM metrics to maintain the frame rate. This is due to frame value comparisons in the reference and degraded clips. Consider, in Figure 5.7, the clip segment contained in frames 11-16 in the reference clip. These frames are temporally aligned with frames 3 and 4 in the degraded clip. Frames are aligned either by matching frames by their

playout time or by applying a best-match frame algorithm during the calibration phase of the analysis. Most objective metrics operate by comparing pixel values in consecutive frames for a particular time segment of the clip. For example, the standard deviation of some parameter calculated in the reference clip for this time segment would be much lower than that calculated in the degraded clip over the same time segment. Similarly, this effects calculations involving frame-to-frame comparisons. For example, if frame 1 in the reference clip was compared against frame 1 in the degraded clip. However, frame 2 in the reference clip would be compared against the closest matching frame in the degraded clip, which would be frame 1. Thus, frame comparisons lead the VQM metric to assume a lower quality when the frame rate is reduced. This effect is more significant for high action content clips where the inter-frame differences are higher.

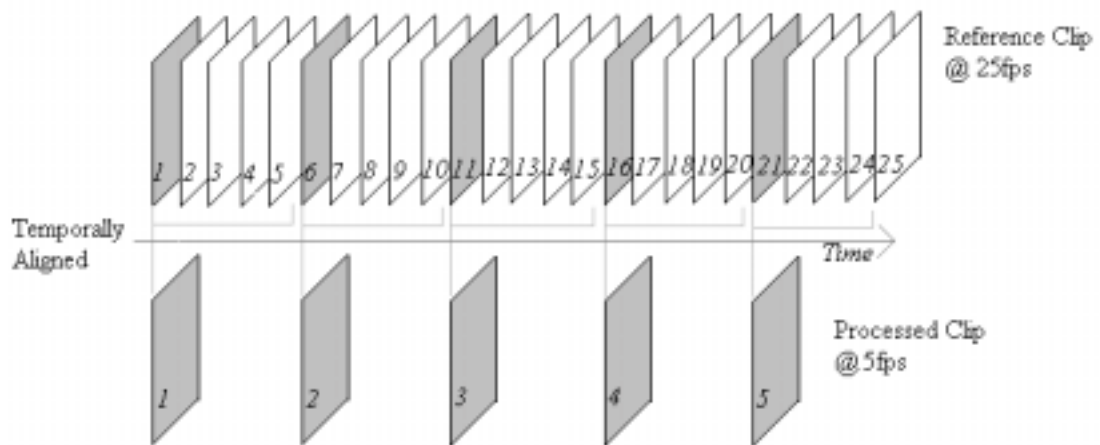


FIGURE 5.7: DIFFICULTIES IN OBJECTIVE TEMPORAL ANALYSIS

These objective metrics have been designed and calibrated for quantifying video quality under static conditions with impairments, an extremely difficult task in itself. This is different from determining the best way to adapt video with maximum user preference. These results highlight that measuring quality and adapting quality based on objective quality measurement are different tasks. Therefore, it can be seen that despite the sophistication of the objective metrics used, they are inefficient and inaccurate for determining the optimum adaptation trajectory. As can be seen from Figure 5.8, there is a little correlation between the path of maximum user preference and those obtained using objective metrics.



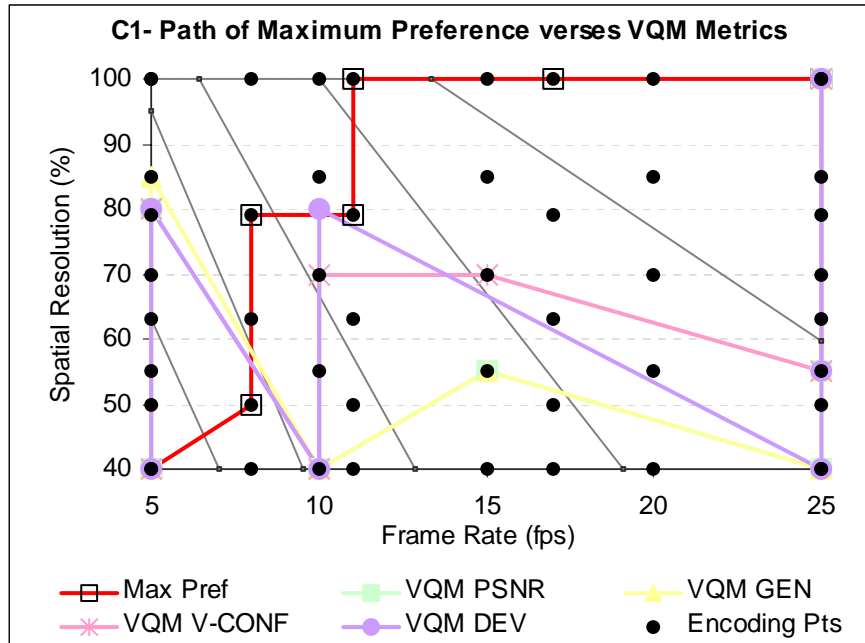


FIGURE 5.8: COMPARISON OF PATH OF MAXIMUM USER PREFERENCE AND MAXIMUM VQM PATHS

Users prefer a smooth graceful, gradual adaptation whereas current objective metrics tend to jump through encoding configurations without a sense of direction and continuity [164]. To discover an optimum adaptation trajectory using objective metrics, quality metrics need access to the adaptation history of the video in order to provide a smooth graceful trajectory of adaptation. The ITU-T has yet to find or recommend an objective metric that correlates adequately to human perception and recommend that subjective testing is required to complement objective metrics. Current objective metrics are designed to evaluate video quality, which is different from adapting video quality.

## 5.2. One-dimensional versus Two-dimensional Adaptation

The primary objective of these experiments is to evaluate the usefulness of the OAT by comparing subjective responses to time varying quality degradation in 1-dimension (either frame rate or spatial resolution) versus quality degradation in two-dimensions (both frame rate and spatial resolution) using the OAT (Figure 5.9).

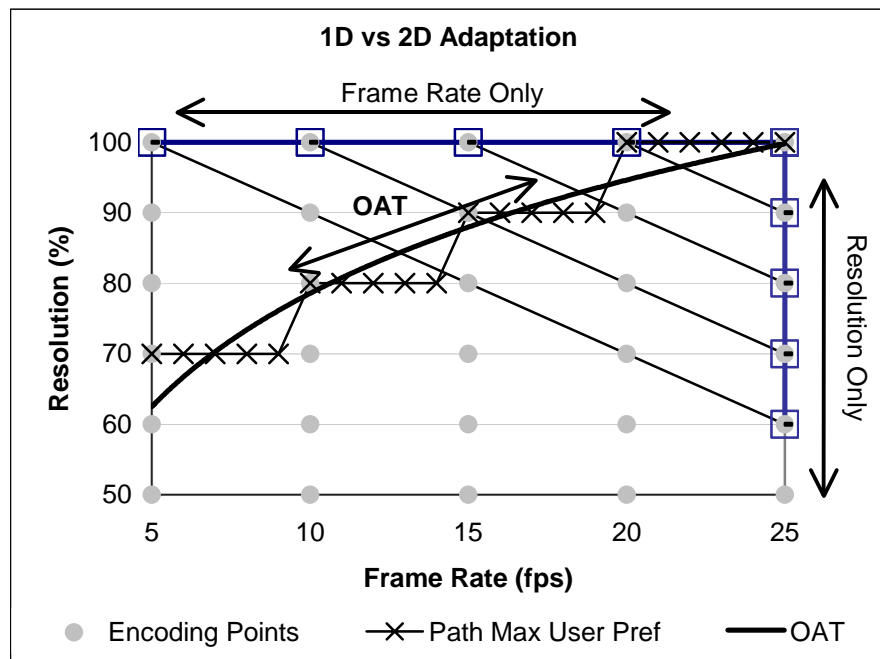


FIGURE 5.9: ONE-DIMENSIONAL VERSUS TWO-DIMENSIONAL ADAPTATION

### 5.2.1. Test Methodology

The Forced choice methodology is suitable for clips lasting not longer than 15 seconds long [165], [166], [167]. For video clips lasting longer than this duration, there are recency and forgiveness effects by the subject, which are a big factor when the subject must grade the overall quality of a video sequence. For example, the subject may forget and/or forgive random appearances of content-dependent artefacts when they are making their overall grade of the video sequence. To test clips of a longer duration a different test methodology to the forced choice method needs to be applied to overcome the forgiveness and recency effects and to ensure the subject can make an accurate judgement.

The Single Stimulus Continuous Quality Evaluation (SSCQE) is a methodology that is designed for the presentation of longer sequences, lasting several minutes long [168]. Continuous evaluation is performed using a slider scale that is used to record the subjects' responses. The choice of a slider is the simplest way for the subject to interact with the

system without introducing too much interference or distraction to the subject. This method is useful to trace the overall quality of the sequence [169].

The reference clip is played out at the beginning so that the subjects are aware of the reference highest quality sequence. As the sequence is played out the subject continuously rates the quality of the sequence using a slider. When the slider is moved, the quality grade of the slider is captured and related to the playout time of the media. Below in Figure 5.10 is a screen shot of the program built to perform and facilitate the subjective testing of 1-dimensional versus 2-dimensional adaptation.



FIGURE 5.10: SCREEN-SHOT OF SSCQE TESTING APPLICATION

### 5.2.2. Test Sequence Preparation

The bit rates of the various encoding configurations demonstrates the limitations in terms of bit rate range available when adapting in one dimension only (Figure 5.11).

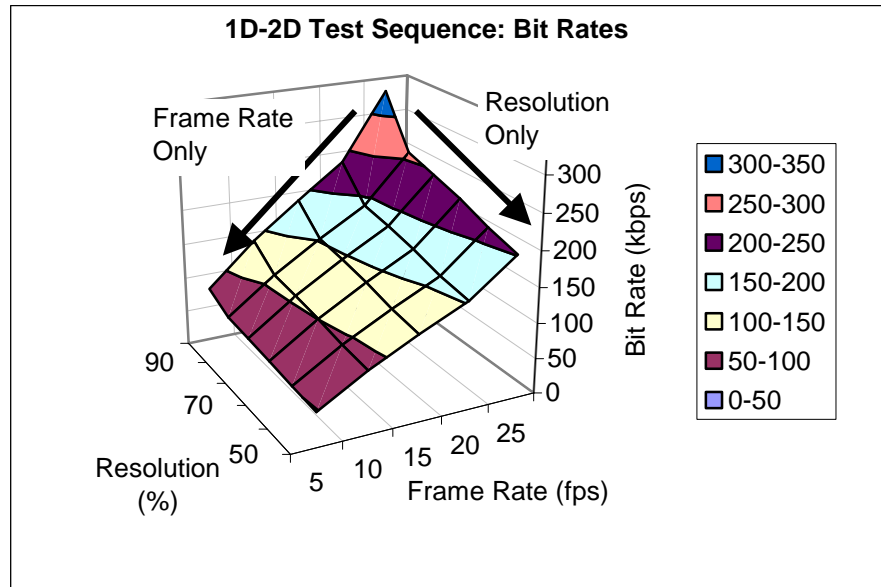


FIGURE 5.11: BIT RATES OF ONE-DIMENSIONAL VERSUS TWO-DIMENSIONAL ADAPTATION

For example,

Adapting only spatial resolution from 100% to 50%:

- Yields a bit rate range of 331kbps → 200kbps.

Adapting only the frame rate from 25fps to 5fps:

- Yields a bit rate range of 331kbps → 80kbps.

Adapting in two dimensions there is greater flexibility.

- Yields a bit rate range from 331kbps at 25fps and spatial resolution 100% to 48kbps at 5fps and 50% spatial resolution.

To prepare the test sequences, so that the bit rates are comparable, the spatial resolution had to be further spatially reduced.

The test sequences were carefully prepared, by segmenting the entire clip into segments of approximately 15 seconds each (Figure 5.12). Each segment was encoded at various encoding configurations of spatial resolution and frame rate. The purpose of segmentation is to ensure that there were no variations in the duration and therefore content at any one time at a particular encoding configuration. In this fashion each segment is comparable at each encoding configuration. These video segments were then pieced together again seamlessly reproducing the original reference sequence with adapting quality in either the frame rate dimension only, the spatial resolution dimension only or both frame rate and spatial resolution dimensions.

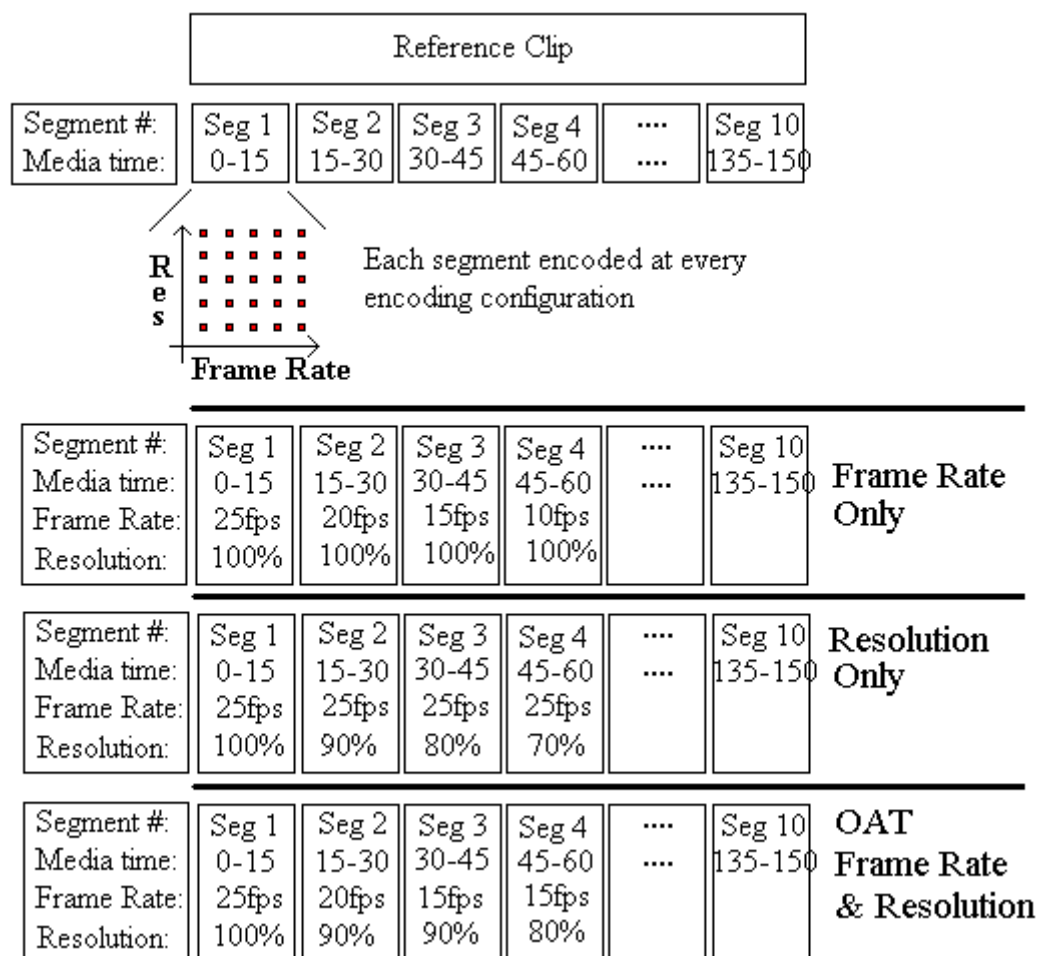


FIGURE 5.12: ADAPTIVE TEST SEQUENCE PREPARATION

### 5.2.3. SSCQE Data Analysis

During each test, the subject continually grades the quality of the clip they are watching. The media time and grade are written to a file. For each clip, the results are aggregated for all subjects (Figure 5.13). The mean opinion score (MOS) and standard deviation are calculated at each media time instant. In this case, each media time instant corresponds to one second of media. In [169], the authors state that, from experience, it is unnecessary to normalise the subject grades. Then, the MOS and standard deviation is calculated for each clip segment.

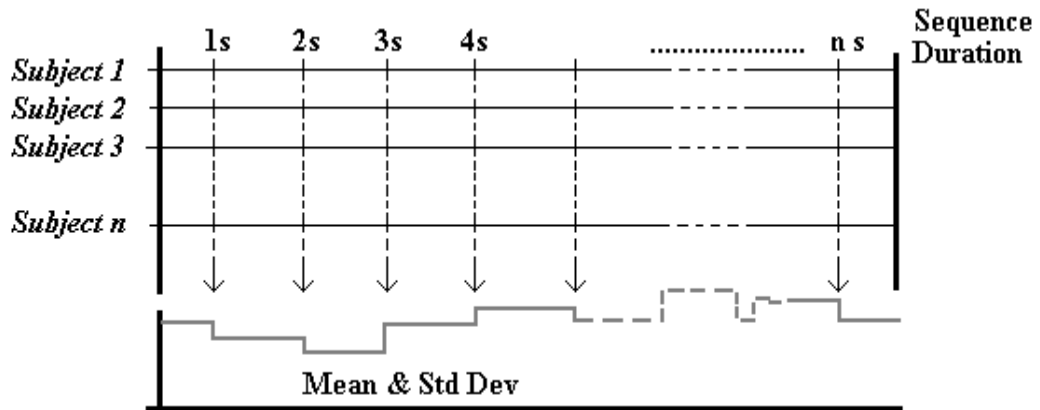


FIGURE 5.13: SSCQE DATA ANALYSIS

### 5.2.4. Test Results

Three scenarios tested, in the first, the quality is adapted down from the best to worst, in the second the quality is upgraded from worst to best and in the third, the quality varies in an additive increase/multiplicative decrease fashion. The first two tests are complementary and are designed to assess symmetrical perception, that is, do subjects perceive quality increases and quality decreases uniformly. The third test is designed to test quality perception in a typical adaptive network environment. Of particular interest are the MOS scores when the quality is decreased.

#### *Adapting Quality Down*

In this test, the quality of the clip degrades from the best quality to the worst quality. Figure 5.14.A shows the bit rate decreasing as the quality degrades. Figures 5.14.C and D show the encoding configuration of frame rate and resolution for each segment as the quality is adapting down in either the frame rate dimension only or the resolution dimension only or using the OAT adapting down in both the frame rate and resolution dimensions. Through segments 1-3 the resolution and frame rate dimensions are perceived the same (Figure 5.14.E). In segment 4 and 5, there appears to be imperceptibility between a decrease in resolution from 80% to 70%. Using the OAT there is a smooth decrease in the MOS scores, which outperforms both one-dimensional adaptation of frame rate and resolution. Segments 4-6 have a high action content which may explain the sharp decrease in the MOS scores for adapting the frame rate only. When there is high action content, subjects prefer smooth continuous motion. Further, when there is high action content, reductions in spatial resolution cannot be perceived as clearly as there is too much happening in the video clip for the detail to be perceived properly. At the lowest quality level in segment 6, the frame rate is perceived worst of all with a high standard deviation. The OAT is perceived best with the lowest standard deviation (Table 5.1).

	Frame Rate Only	Resolution Only	OAT
Sequence Average MOS Score	82%	88%	91%
Sequence Average Std Dev MOS Score	6.7%	5.7%	3.9%
Segment 6 Average MOS Score	61%	78%	82%
Segment 6 Average Std Dev MOS Score	8%	6%	5%

TABLE 5.1: ADAPT DOWN RESULTS

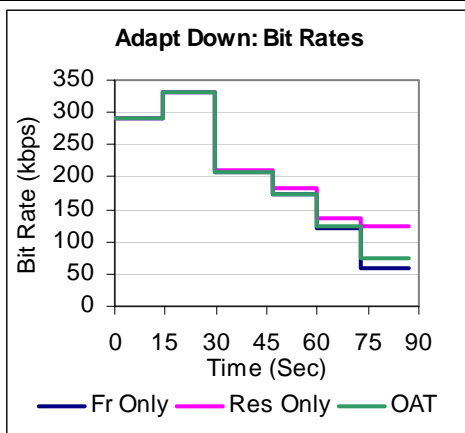


FIGURE 5.14.A: ADAPTING DOWN – BIT RATES

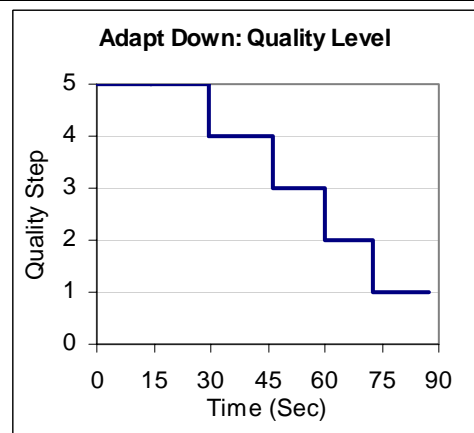


FIGURE 5.14.B: ADAPTING DOWN – QUALITY LEVELS

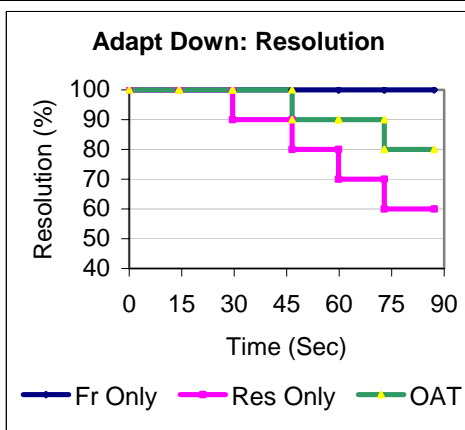


FIGURE 5.14.C: ADAPTING DOWN RESOLUTION

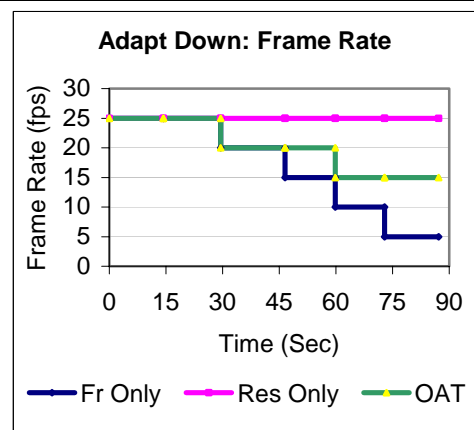


FIGURE 5.14.D: ADAPTING DOWN FRAME RATE

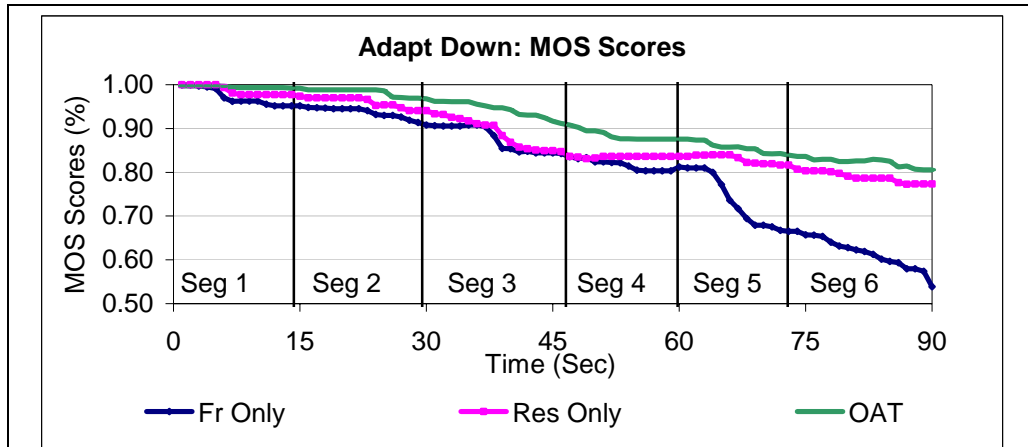


FIGURE 5.14.E: ADAPTING DOWN – MOS SCORES

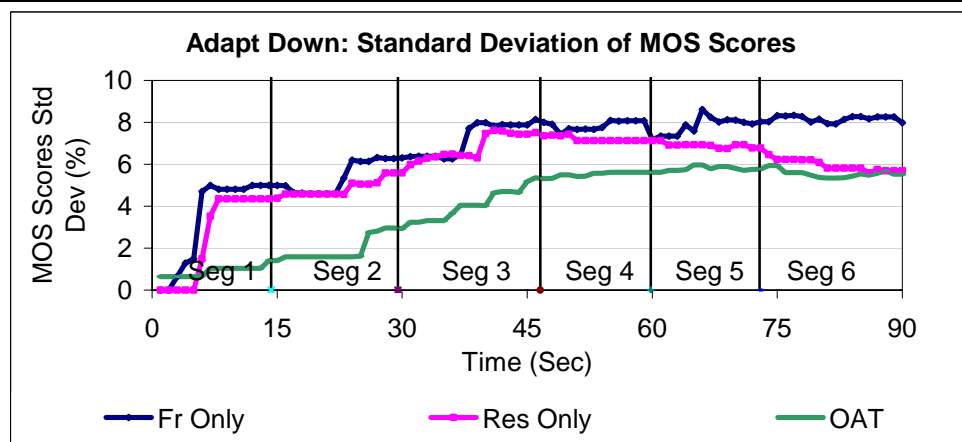


FIGURE 5.14.F: ADAPTING DOWN – STANDARD DEVIATION OF MOS SCORES

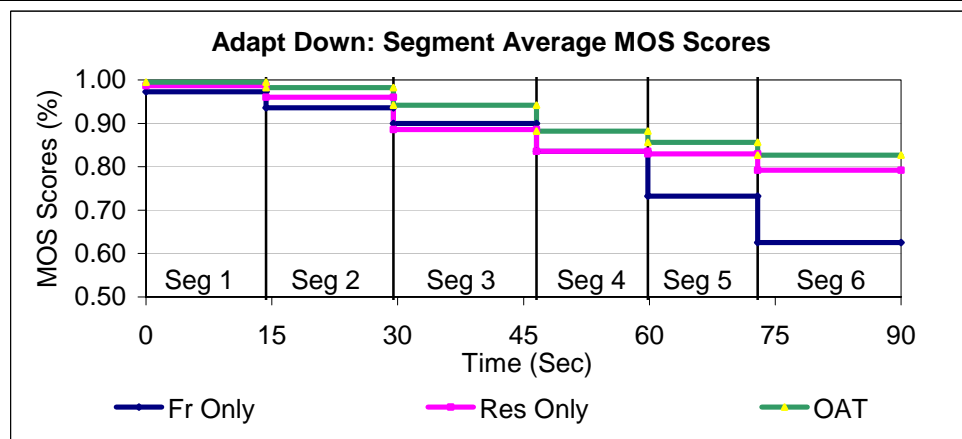


FIGURE 5.14.G: ADAPTING DOWN – SEGMENT AVERAGED MOS SCORES

FIGURE 5.14: ADAPTING DOWN RESULTS

*Adapting Quality Up*

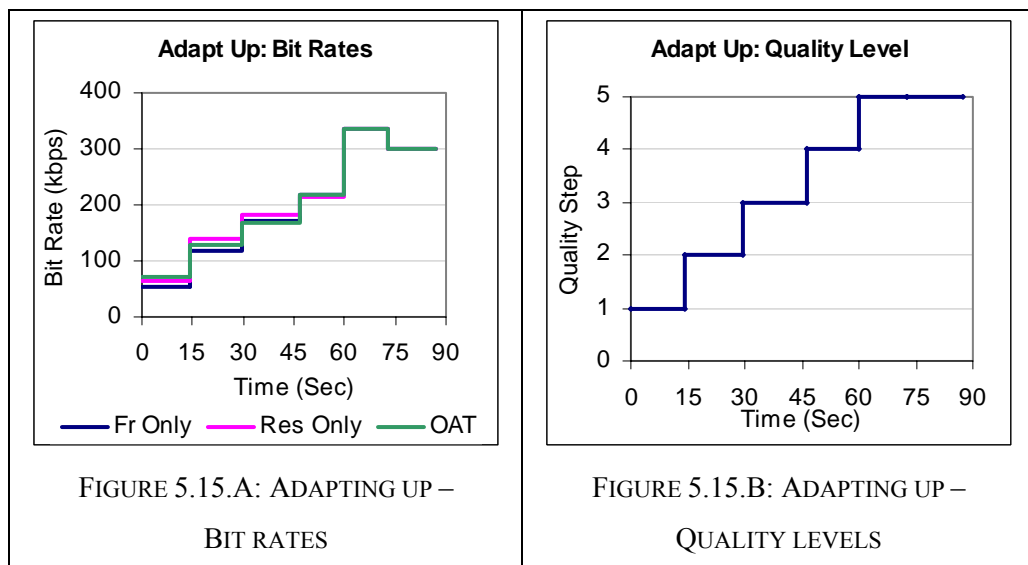
In this test, the quality of the clip upgrades from the worst quality to the best quality. The last two segments are encoded at the highest quality. Figures 5.15.C and D show the



encoding configuration of frame rate and resolution for each segment as the quality is adapting up in either the frame rate dimension only or the resolution dimension only or using the OAT adapting down in both the frame rate and resolution dimensions. During this experiment, the slider is placed at the highest quality value on the rating scale when the clip begins. It can be seen that it took subjects several seconds to react to the quality level and adjust the slider to the appropriate value (Figure 5.15.E). At low quality, subjects perceive adaptation using the OAT better than one-dimensional adaptation. The quality is slowly increasing however subjects do not seem to notice the quality increasing nor do they perceive it significantly differently indicating that subjects are more aware of quality when the quality is low (Figure 5.15.G).

	<b>Frame Rate Only</b>	<b>Resolution Only</b>	<b>OAT</b>
Sequence Average MOS Score	72%	74%	78%
Sequence Average Std Dev MOS Score	8%	7.4%	5.9%
Segment 2 Average MOS Score	62%	66%	72%
Segment 2 Average Std Dev MOS Score	11%	8.5%	7%

TABLE 5.2: ADAPT UP RESULTS



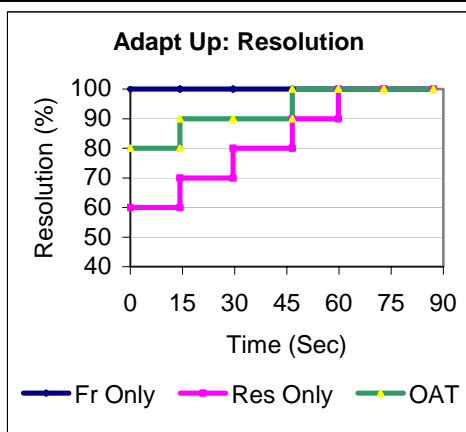


FIGURE 5.15.C: ADAPTING UP RESOLUTION

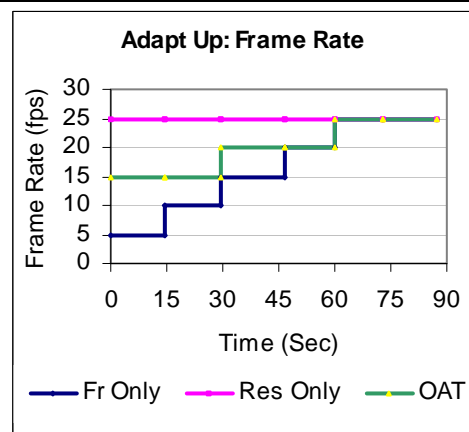


FIGURE 5.15.D: ADAPTING UP FRAME RATE

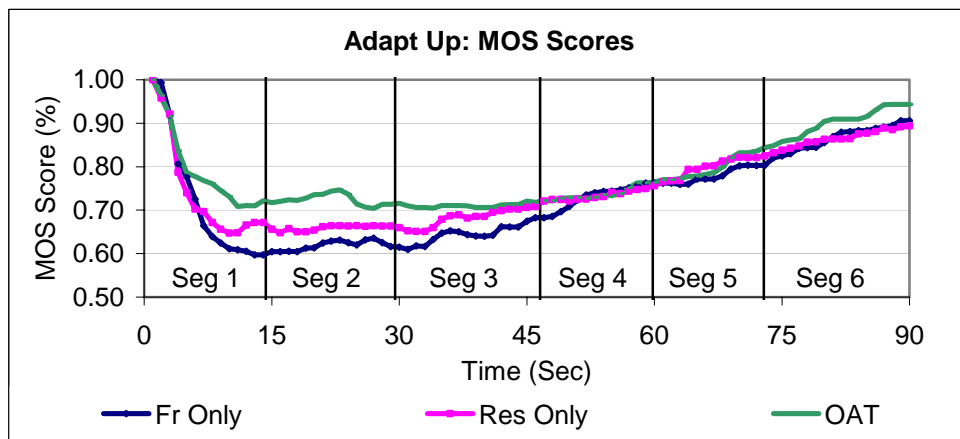


FIGURE 5.15.E: ADAPTING UP – MOS SCORES

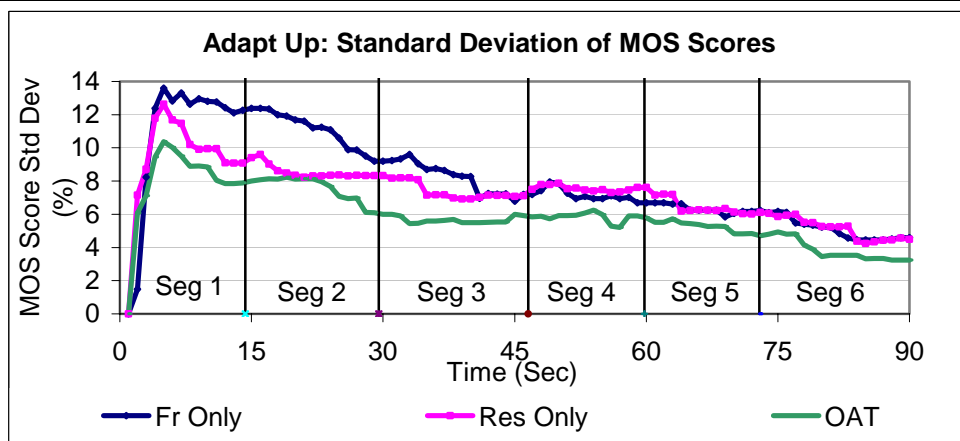


FIGURE 5.15.F: ADAPTING UP – STANDARD DEVIATION OF MOS SCORES

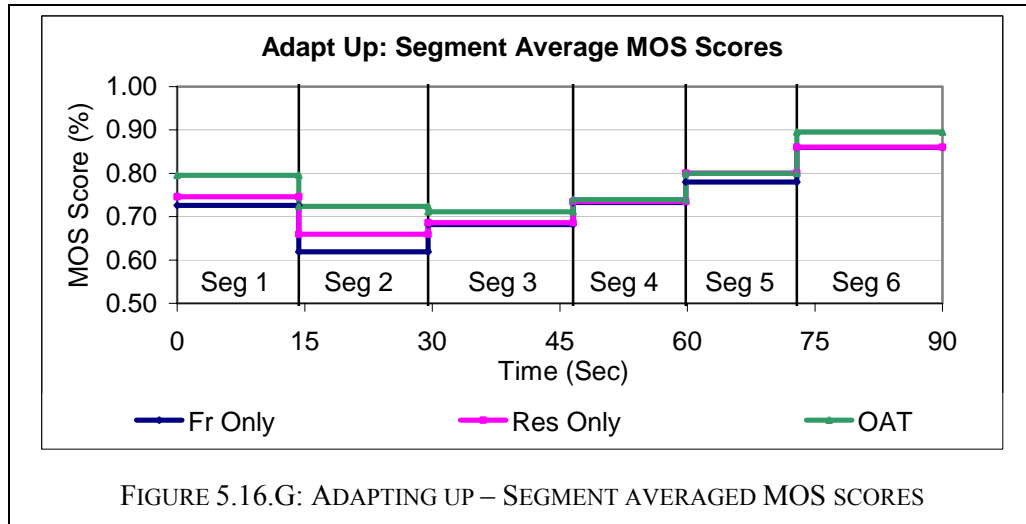


FIGURE 5.15: ADAPTING UP RESULTS

### *AIMD Quality Adaptation*

Most adaptation algorithms behave in an additive increase multiplicative decrease (AIMD) manner. In this test, the quality of the clip varies over time between the worst quality and the best quality (Figure 5.16.A). Figures 5.16.C and D show the encoding configuration of frame rate and resolution for each segment as the quality is adapting in either the frame rate dimension only or the resolution dimension only or using the OAT adapting down in both the frame rate and resolution dimensions.

It can be seen that the first MD to Quality Level 2 occurs in segment 2 but not perceived until segment 3 allowing for the time delay for subjects to react quality adaptation (Figure 5.16.E). Once again, subjects MOS scores are slow to react to an increase in quality. In segment 6, the second MD drop to Quality Level occurs. The MOS scores for adapting only the frame rate and spatial resolution are quick to reflect this drop however, using the OAT, it takes subjects do not perceive this drop in quality as strongly. The average MOS scores during this period using the OAT are 20% better than those adapting the frame rate only and 10% better than adapting the spatial resolution only. In addition the standard deviation using the OAT is also much lower. This is a high action part of the sequence and so the reduced frame rate is perceived more severely. There is imperceptibility between a spatial resolution of 60% and a spatial resolution 70%, as can be seen in segments 6 to 7 and again in segments 9 to 11 (Figure 5.17.G).

	Frame Rate Only	Resolution Only	OAT
Sequence Average MOS Score	73%	77%	80%
Sequence Average Std Dev MOS Score	8%	7%	5%
Segments 6&7 Average MOS Score	55%	66%	77%
Segments 6&7 Average Std Dev MOS Score	9.1%	8.1%	6.7%

TABLE 5.3: AIMD RESULTS

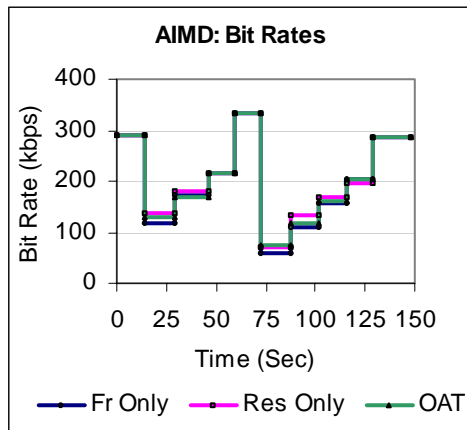


FIGURE 5.16.A: AIMD ADAPTATION – BIT RATES

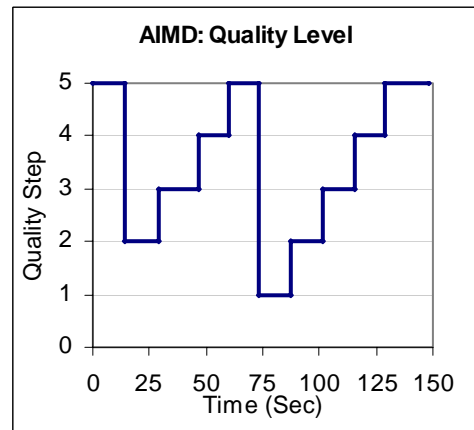


FIGURE 5.16.B: AIMD – QUALITY LEVELS

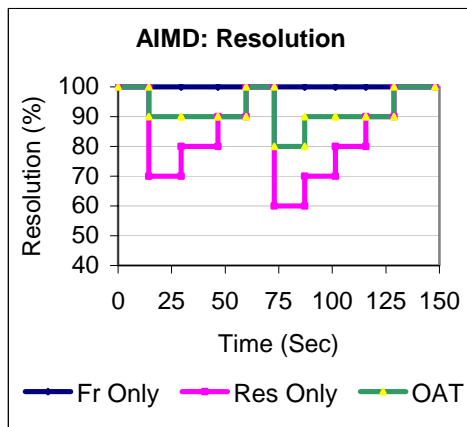


FIGURE 5.16.C: AIMD ADAPTATION RESOLUTION

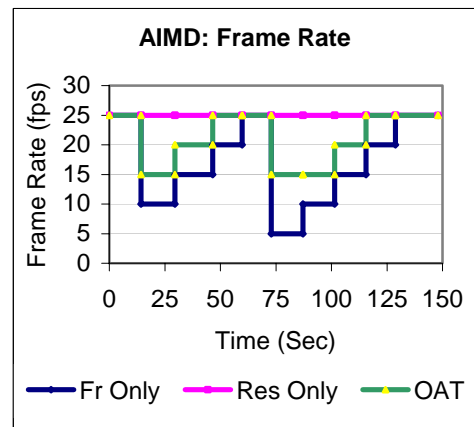


FIGURE 5.16.D: AIMD ADAPTATION FRAME RATES

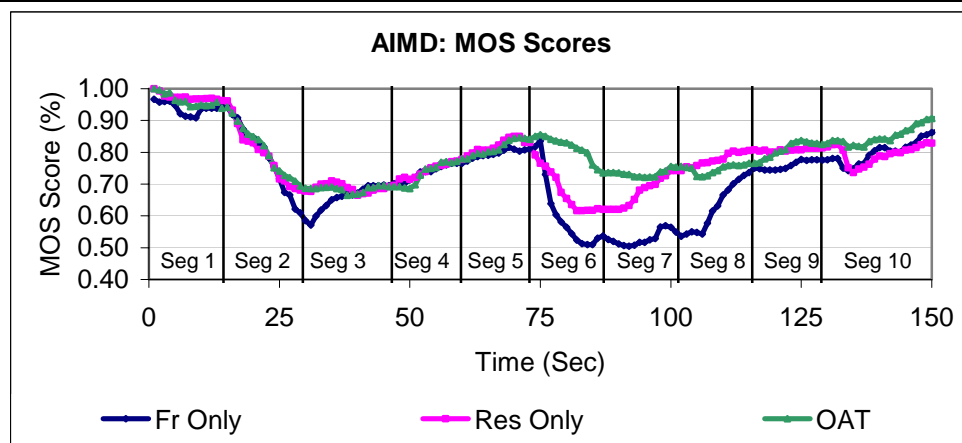


FIGURE 5.16.E: AIMD ADAPTATION – MOS SCORES

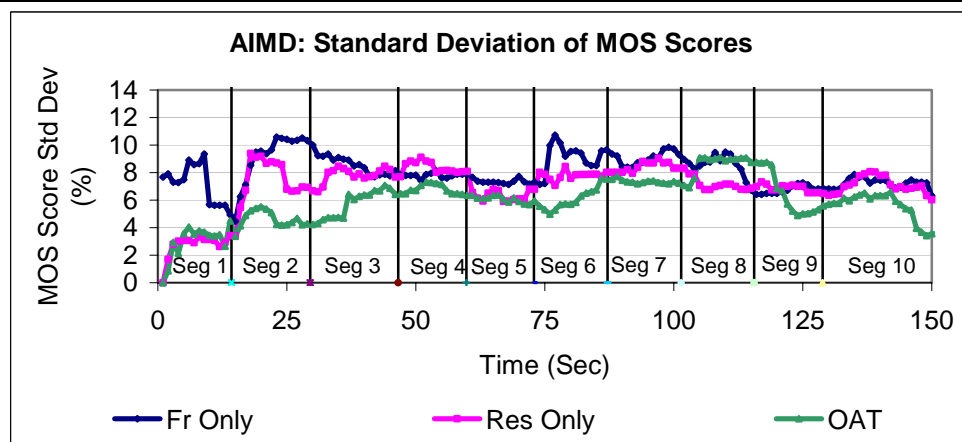


FIGURE 5.16.F: AIMD ADAPTATION – STANDARD DEVIATION OF MOS SCORES

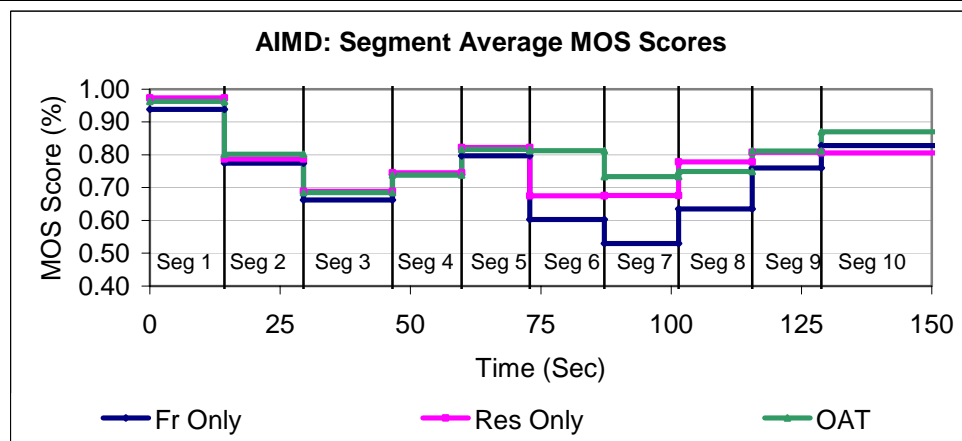


FIGURE 5.16.G: AIMD ADAPTATION – SEGMENT AVERAGED MOS SCORES

FIGURE 5.16: AIMD ADAPTATION RESULTS

### 5.2.5. Discussion

From the various experiments conducted, subjects perceived adapting frame rate the worst, followed by resolution and the OAT best of all. It was observed that there is a time delay for subjects to react to quality adaptations. It was also observed that quality perception is asymmetrical when adapting the quality down and adapting quality up. For example, during the worst quality level of the sequence in the adapting down and adapting up tests, there was a 1% difference in the MOS scores for adapting the frame rate only; a 12% difference in the MOS scores for adapting the resolution only and a 10% difference in the MOS scores adapting using the OAT. Perception is strongly dependent on the contents characteristics i.e. if there is high action or high detail in the clip. In some tests, despite the encoding configurations not changing, the segments have different MOS scores. In addition to the greater bit rate adaptation range achieved using the OAT, adapting using the OAT outperforms one-dimensional adaptation using frame rate and spatial resolution. The OAT can be dynamic if the contents spatial and temporal characteristics are known at a given instant, thus making it more flexible to adapt according to the contents characteristics and maximise user perceived quality.

### 5.3. Chapter Conclusions and Summary

In this chapter the possibility of using an objective metric to automate the discovery of the OAT was investigated. Two metrics were selected, the PSNR and VQM, to try and determine whether they yielded an OAT. Despite the sophistication of the metrics chosen, the paths of maximum quality measured by these metrics did not correlate in any way to the OAT discovered by subjective testing for the same content type.

The usefulness of the OAT in a real video adaptation scenario was investigated using the SSCQE methodology. Two modes of adaptation were investigated, Additive Increase Additive Decrease (AIAD) and Additive Increase Multiplicative Decrease (AIMD). For each mode the video was adapted in one of three ways, adapting the frame rate only, adapting the resolution only and adapting both the frame rate and resolution using the OAT. It can be concluded that the users prefer a two-dimensional adaptation policy with a trade-off of spatial resolution against frame rate proportional to the contents characteristics. In all cases, adaptation using the OAT out-performed one-dimensional adaptation of frame rate or spatial resolution currently employed for adaptive delivery of pre-encoded multimedia content. The next chapter will demonstrate how knowledge of the OAT can be used with sender-based adaptation algorithms for streaming multimedia over best-effort IP networks.

# Chapter 6

## Adaptive Streaming

### *Chapter Abstract*

In chapter 4, the concept of an OAT was proposed and proven to exist. In chapter 5, it was shown that the usefulness of this OAT lies in its application to adaptive streaming of multimedia. The server can make more appropriate adaptation decisions by using knowledge of the OAT, as it indicates how to adapt encoding quality in response to changes in network conditions to maximize user-perceived quality. This chapter will demonstrate how knowledge of the OAT can be used to complement any sender-based adaptation algorithm. Further, a new adaptation algorithm, Perceptual Quality Adaptation (PQA), has been developed that uses the OAT directly as a means for making adaptation decisions by more accurately attempting to match the clients' playout capabilities rather than the clients' network capacity.

Sections 1 and 2 provide a high level overview of the system developed, its adaptation capabilities and various design and development decisions made. In section 3 the OATs' ability to inter-work with and complement any sender-based adaptation algorithm are demonstrated. In section 4, a new adaptation algorithm, PQA, is proposed and several examples of its behaviour are presented. Section 5 discusses networking issues and factors that affect streaming multimedia over wireless networks. Various wireless network test scenarios have been chosen and are tested by both emulation and simulation in section 6.

### 6.1. Adaptive Streaming Overview

#### **6.1.1. System Overview**

The prototype system developed is a client-server architecture. Both client and server consist of the RTP/UDP/IP stack with RTCP/UDP/IP to relay feedback messages between the client and server. Both the client and server should be able to handle content encoded and hinted with an MPEG-4 codec. There are two open-source streaming servers available, Helix from Real [170] and Darwin Streaming Server from Apple [171][172][173][174]. The main characteristics of both are outlined in Table 6.1.



	<b>Helix</b>	<b>Darwin Streaming Server (DSS)</b>
Released	2003	1999
File Format	MP3, RealAudio, RealVideo (.rm, .ra, .rv))	ISO-compliant hinted MP4, 3GP, Mov, MP3 (using Icecast-compatible protocols over http)
Compatibility		Any ISO-compliant MP4 device (including Windows Media), 3GPP device, MP3 (iTunes, WinAmp, SoundJam)
Transport	RTSP, RTP streaming, HTTP Tunnelling, TCP, UDP. HTTP delivery supporting RealAudio and RealVideo delivery using RTSP/RDT via TCP, UDP unicast and UDP multicast transports.	RTSP, RTP streaming, RTCP, HTTP Tunnelling, TCP, UDP. Unicast and multicast support. Instant-on streaming option. Skip protection for fast-buffering option.
Broadcast	SDP announcement for RTP via UDP unicast and UDP multicast standards based live encodes. Live broadcasting for RealAudio and RealVideo from the Helix Producer 9.0	SDP announcement for RTP via UDP unicast and UDP multicast standards based live encodes or playlists. Allows creation of simulated live broadcasts with Playlist Broadcaster. Multiple playlist play options supported, sequential playback, sequential looped playback, and weighted Random playback. Acts as a reflector for live broadcasts, Provides relay functionality for setting up multiple servers.

TABLE 6.1: DARWIN STREAMING SERVER VERSUS REALPLAYER

The main advantage of Real, is that the Helix DNA platform provides source code for all three Producer, Server and Client, not just the server, like DSS. However, Helix does not follow ISMA guidelines and instead uses its own proprietary encoding (i.e. RealAudio, RealVideo and RealMedia) and a proprietary transport protocol, RDT, which is very similar

to RTP. The Internet Streaming Media Alliance (ISMA) promotes the adoption of a single technology, MPEG-4. However, to date, current releases of Helix do not support MPEG-4. One of the main concerns in the system design and development was heterogeneity. The reason for this is that the prototype streaming server should be able to inter-work with existing technologies i.e. a specific client player should not be tied a specific server nor should it employ any proprietary transport protocols. In the worst-case scenario, users would require a separate player for each different server. Thus, DSS was chosen as the streaming server for the system development.

DSS is an open-source, standards-based streaming server that runs on Windows NT and Windows 2000 and several UNIX implementations, including Mac OS X, Linux, FreeBSD, and the Solaris operating system. The prototype server complies with Internet Engineering Task Force (IETF) protocols:

- Real Time Streaming Protocol (RTSP)
- Real Time Transport Protocol (RTP)
- Real Time Transport Control Protocol (RTCP)
- Session Description Protocol (SDP)
- User Datagram Protocol (UDP)
- Transport Control Protocol (TCP)
- HyperText Transport Protocol (HTTP)

DSS consists of a number of distributed components, such as broadcasters and relay servers, which facilitate the deployment of a distributed multimedia streaming service in both unicast and multicast mode. The client can be any QuickTime Player or any player that is capable of playing out ISMA compliant MPEG-4 or .3pg content. The system developed is compliant to MPEG-4 standard profiles, ISMA streaming standards and all IETF protocols.

### **6.1.2. Heterogeneous and Server-Specific Clients**

The prototype system was developed and designed to be able to cater for both general clients as well as the clients that return the application specific feedback, called ‘server-specific clients’. Server-specific clients return both RTCP-RR feedback and application specific RTCP-APP feedback whereas general-clients only return RTCP-RR feedback. RTCP-APP feedback contains information about the quality of the clients’ playout of the received stream, including the receiver bit rate, average buffer delay, buffer occupancy, playable frame rate and the expected frame rate. This feedback enables the prototype server to evaluate the user-perceived-quality of the received stream (Figure 6.1).

Most adaptive streaming servers attempt to reduce the loss rate to zero, which could lead to an excessive degradation of the transmitted quality. The system developed aims to reduce the loss rate to the point of tolerable loss, which is determined by the robustness of the encoding. For example, most sender-based adaptation algorithms insist that the bit rate be reduced when the loss rate is above zero regardless of whether these errors/losses could be masked or recovered. However, if RTCP-APP feedback indicates that the quality played out by the client equals the expected quality, then, regardless of the loss rate reported, the prototype server should either not adapt or upgrade along the OAT as though the loss rate is zero. Another advantage of using the RTCP-APP feedback is that there may be factors other than network losses, which may contribute to frames being discarded or dropped by the client, for example, buffer overflow, so even if the loss rate reported by the RTCP-RR is zero, there is still a degradation of quality at the client side.

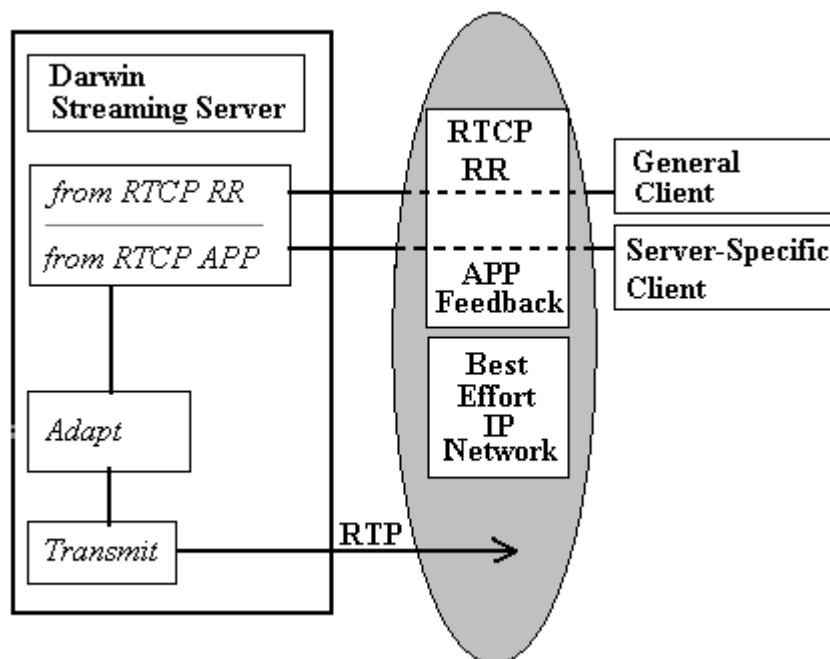


FIGURE 6.1: BASIC CLIENT-SERVER ARCHITECTURE

## 6.2. System Adaptation

### 6.2.1. System Adaptation for Pre-encoded multi-tracked content

When the prototype server receives feedback from the client using either RTCP-APP feedback or RTCP-RR feedback, the server adapts the transmission rate of the stream being delivered to the client. Adaptation can occur in a number of ways depending on the session type i.e. real-time encoded streaming session or pre-encoded streaming session. In general, most unicast real-time streaming over the Internet delivers pre-encoded files whereas real-time encoded streams are typically multicast over the network.

Real-time streams require an encoder to encode and prepare the content “on the fly”. The main difficulty with this method is that the adaptation is confined to the adaptation facilities of the encoder and thus adaptation is encoder-dependent. Further, multicast, by its nature, is limited in its powers of adaptation as it must aggregate the feedback from the client-base and make an adaptation decision for the group. In a typical pre-encoded unicast streaming system, the user selects the encoded version that best matches their network connection. This method is very constrained in its ability for adaptation, which usually involves various methods of flow control.

A design decision was made to make the system adaptation as generic as possible so that device-dependencies could be avoided. By using pre-encoded multi-tracked content, the prototype server can switch between video tracks with differing quality versions, effectively adapting the quality of the stream being delivered and its resulting transmission bit rate. In the system developed, adaptation occurs on two levels, a spatial level and a temporal level. Spatial adaptation is achieved by seamlessly switching between different video tracks with different spatial resolutions. Temporal adaptation occurs by frame dropping during the streaming process.

### 6.2.2. Multi-tracked MPEG-4 and 3GP Content

Third-generation (3G) mobile networks are being deployed and multimedia-enabled mobile devices are becoming popular and widely available, leading to a need for a wireless multimedia standard. In industry there are many competing and proprietary media formats. 3GP is the new developing standard defined by the 3rd Generation Partnership Project (3GPP), for the creation, delivery, and playback of multimedia over wireless networks [175],[176][177]. It enables multimedia files to be shared between a variety of devices, including mobile phones, PDA’s, and computers.

3GP format files are based on the ISO base file format, upon which the MPEG-4 format is based, which is a track-based file format. This allows the ability to mix different types of media, like video, audio and text, in a single file. 3GP allows for both MPEG-4 and H.263 video encoded content and AAC (Advanced Audio Coding) and AMR (Adaptive Multi-Rate) audio content and timed text tracks to be contained into one multimedia file. MPEG-4 files are made of data structures called atoms and each atom has a header, which includes its size and type [178][179]. The atom's type indicates what kind of data it contains. A parent atom is of type *moov* and contains the following child atoms: *mvhd* (the movie header), a series of *trak* atoms (the media tracks and hint tracks), and a movie user data atom *udta*. A *trak* represents a single independent data stream and an MPEG-4 file may contain any number of the following tracks:

- Video tracks
- Audio tracks
- Hint tracks
- Binary Format for Scenes (BIFS) track
- Object Descriptor (OD) track

Each video and audio track must have its own associated hint track. Hint tracks are used to support streaming by a server and indicate how the server should packetize the data. As with MPEG-4 streaming, 3GP files use the “hint track” mechanism for streaming the content. In 3GP files the BIFS and OD tracks are optional and can be ignored.

Server extensions enable a server to relate different tracks and use them for selection and adaptation. In particular, they enable a server to generate Session Description Protocol (SDP) descriptions with alternatives, select and combine tracks with alternative encodings of media before a presentation and switch between tracks with alternative encodings during a streaming session. 3GP files may conform to one or more profiles but it is not mandatory [180]. The *basic profile* is targeted to work with MMS (Multimedia Messaging Service) and requires that the file is self-contained with a single video track, audio track and text track. The *streaming server profile* allows for adaptation with the selection of alternative encodings of content and adaptation by switching between various tracks with alternate encodings during streaming (Figure 6.2).

- **Groupings of alternative tracks:** By default all enabled tracks in a 3GP file are streamed (played) simultaneously. However, tracks that are alternatives to each other can be grouped into an alternate group. Tracks in an alternate group that can be used for switching can be further grouped into a switch group.

- **Alternate group:** Only one track within an alternate group should be streamed or played at any time and must be distinguishable from other tracks in the group via attributes such as bit rate, codec, language, packet size etc.
- **Switch group:** Tracks that belong to the same switch group, belong to the same alternate group.
- **Hint tracks:** All media tracks must have their own associated RTP hint track.

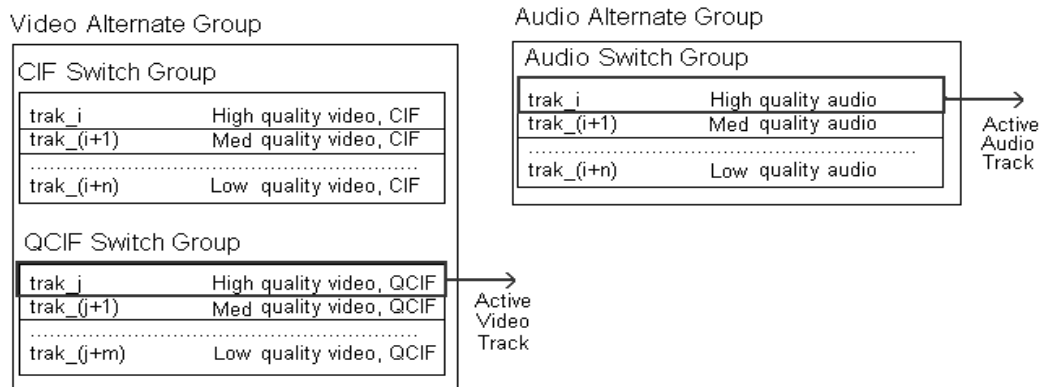


FIGURE 6.2: MULTI-TRACKED .3GP FILE

When the client first connects to the prototype server, the server reads the pre-encoded file, analyses and loads all the tracks and their associated hint tracks for the session. The prototype server sorts these tracks into appropriate audio and video alternate and switch groups. Switch groups can be chosen to target particular devices, networks, content preferences such as foreign languages etc.

In the system, all tracks in a video switch group have the same display size, that is, all tracks are QCIF display size and form the QCIF switch group. Tracks within the QCIF video switch groups have a different spatial resolution. Only one track may be sent from the video alternate group and one from the audio alternate group at any instant. By default, on connection establishment, the prototype server initially delivers the highest quality video track from the appropriate switch group i.e. spatial resolution = 100% and frame rate = 25fps. The prototype server de-activates all the other tracks within this alternate group.

Some players currently available are not able to play out multi-tracked MPEG-4 video files. Windows Media Player and Real Player do not support multi-layered video. For example, Windows Media Player version 7 opens a new window for each video track present in the multi-layered video file and then crashes. Windows Media Player version 9 and Real Player do not recognise the media type when there are multiple video tracks in the file despite the

fact that each track is a recognisable media type. These players currently do not support .3gp files but when this is implemented they will have to recognise multi-tracked content. However, QuickTime Player does support .3gp and multi-tracked MPEG-4 video files. Thus, the Quicktime player was chosen as the client player for system development.

### 6.2.3. Temporal Adaptation: Frame Dropping

Frame dropping is often used as a method to reduce bit rate in adaptive streaming servers [181]. In the MPEG-4 standard, there are a number of profiles, which determine the capabilities of the player to play out encoded content. The purpose of these profiles is that a codec only needs to implement a subset of the MPEG-4 standard whilst maintaining interworking with other MPEG-4 devices built to the same profiles. The most widely used MPEG-4 visual profiles are the MPEG-4 Simple Profile (SP) and the MPEG-4 Advanced Simple Profile (ASP). These are part of the non-scalable subset of visual profiles. The scalable profiles require a real-time encoder.

In MPEG-4 video frames are known as Video Object Planes (VOP's). These VOP's are encoded in a strict sequence, known as a Group Of VOP's (GOV). The main difference between MPEG-4 SP and ASP is that SP contains only I and P-VOP's whereas ASP contains I, P and B-VOP's.

For ASP, the GOV sequence is:      I P BB P B B P BB I

For SP, the GOV sequence is:      I P P P P P P P P I

VOPs are inter-dependent which restricts how the frames can be decoded. For example, B-VOP's are bi-directional and so in order to decode a B-VOP; the decoder needs the preceding I or P-VOP and the next I-VOP or P-VOP. The prototype server applies a strict frame dropping policy during the transmission process (Figure 6.3). Thus, B-VOP's are preferentially dropped, followed by P-VOP's. I-VOP's are the last to be discarded but are often not discarded as the I-VOP rate is typically very low in the order of three to five I-VOP's per second.

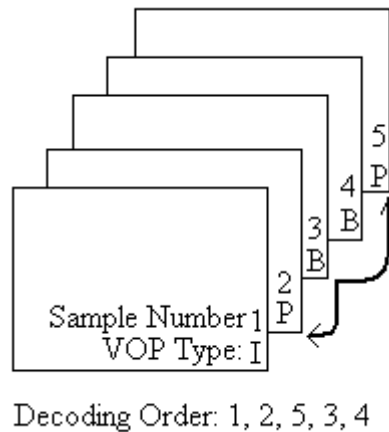


FIGURE 6.3: VOP INTER-DEPENDENCIES

When the prototype server is streaming from a video track, it reads in each VOP at a time. The server analyses the meta-information of each VOP to determine the VOP type, size etc. The Trak Meta information contains an atom called *stss*, which contains the sample numbers of all the sync frames (I-VOPs) contained in the video track whereas the *stsz* atom contains meta-information about all the samples or VOP's in the video/audio track. If the server needs to drop frames, the server uses knowledge of the MPEG-4 profile and GOV sequence to determine a target GOV sequence to achieve the desired frame rate. The server checks the VOP type of each sample to determine which frames can be discarded. In ASP, the server preferentially drops B-VOP's starting with the last in each GOV, followed by the P-VOP's starting with the last in each GOV. The frame rate is never reduced below 5fps, so no I-VOP's are dropped. In SP, the server uses the same frame dropping procedure as used in ASP. A further analysis of the frame dropping policy is in Volume 2 Appendix G.

#### 6.2.4. Spatial Adaptation: Track Switching

Spatial resolution is another means of adapting the bit rate and is typically implemented as a form of stream switching [182]. When the client first connects to the prototype server, the server reads the pre-encoded file, analyses and loads all the tracks and their associated hint tracks for the session. The server then makes a decision to initially deliver the highest quality track i.e. spatial resolution = 100% and frame rate = 25fps. The prototype server deactivates all the other tracks.

During the streaming process the server keeps a log of the playout time of the media samples. When the server needs to switch tracks, it records the current playout time and gets the media time,  $T$ , from the current track. All tracks are encoded with the same frame rate



and GOV sequence. All I-VOP's are aligned and the media timescale is common to all tracks regardless of their quality encoding. MPEG-4 provides random access allowing the server to jump to any point in the media and begin streaming from that point onwards. Sample tables facilitate random access to any time in a multimedia file (Figure 6.4).

To seamlessly switch tracks, the server locates the next nearest random access point of an I-VOP (or sync-sample) to the time,  $T$ , using the *stbl* atom and the sync sample table, *stss* in both the current track and the track that is to be activated. Once the sample number of the next nearest I-VOP required in the new track is known, the sample-to-chunk table, *stsc*, is used to locate the chunk in which this sample is located. The chunk-offset atom is used to determine where this chunk begins and finally the sample size atom, *stsz*, is used to determine where the requested I-VOP is located. The switch is then pending until the streaming media time of the current track equals that of the random access point in the new track. During the switch, the server seeks to this media sample time, de-activates the old track, activates the new track and begins streaming from the correct time in the new track.

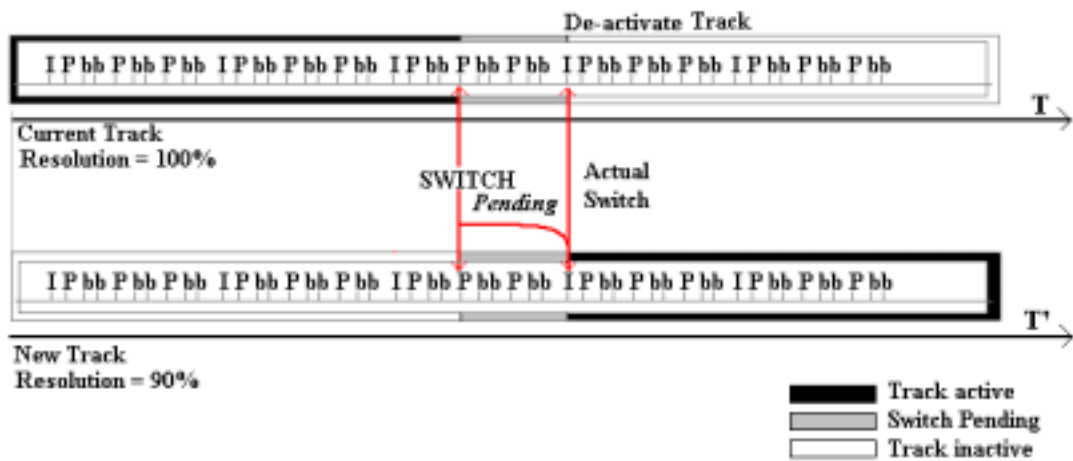


FIGURE 6.4: TRACK SWITCHING

### 6.3. Using OATs with Any Sender-Based Adaptation Algorithm

In general, adaptation policies (including sender-based, receiver-based and encoder-based schemes) address the problem of how to adapt in terms of adjusting the transmission rate or the window size. For example, the sender-based scheme Loss-Delay Adjustment (LDA) algorithm [183] and its variants [184, 185] indicate how to adjust the transmission rate of the sender in response to various network conditions of loss and delay. But for a given bit rate, there are several ways to encode the content. Similarly, the receiver-based scheme RLM (receiver-driven layered multicast [186]) devises a policy of join/leave experiments for additional layers of video to improve the perceived quality and make the most of the available bandwidth. Again, the base and enhancement layers can be composed in a large number of different ways. Knowledge of the OAT can be used to complement these adaptation algorithms. For example, the adaptation algorithm indicates the target bit rate that should be transmitted given certain network conditions and the OAT can then indicate how this new bit rate can be achieved in terms of real encoding parameters. Similarly for layered schemes, the OAT can be used to indicate how the base layer and enhancement layers should be composed.

#### 6.3.1. LDA using OATs

OATs can be used with any sender-based rate adaptation algorithm [187] and this shall be demonstrated using LDA. The LDA algorithm is a typical sender based Additive Increase Multiplicative Decrease (AIMD) adaptation scheme, which relies on RTCP-RR feedback information about the losses at the receivers to make adaptation decisions. During periods of no loss, the sender increases the transmission rate by an additive increase rate (AIR), which is estimated using the loss, delay and bottleneck bandwidth values from the RTCP feedback. Thus,

$$AIR_i = AIR(1 - r/b)$$

Where  $r$  is the current transmission rate,  $b$  is the estimated bottleneck bandwidth and  $AIR$  is the current value of additive increase rate. The bottleneck bandwidth,  $b$ , is estimated using a packet-pair probe approach, i.e.  $b$  is equal to the size of the probe packet divided by the arrival time separation between two probes packets sent sequentially [188]. If two packets are sent together, they are queued as a pair at the bottleneck, with no packets intervening them, then the inter-packet spacing is proportional to the time required for the bottleneck router to process the second of the packet pair. AIR is set initially to a small value and is then increased during periods without losses. From this, the new transmission rate is,

$$r_i = r + AIR_i$$

For the implementation of LDA used here, the maximum transmitted bit rate is bounded to correspond to 100% spatial resolution at the maximum frame rate. However, if the client reports loss, then the transmission rate is multiplicatively reduced,

$$r_i = r(1 - l * R_f)$$

Where  $l$  is the reported loss rate during the RTCP interval and  $R_f$  is the reduction factor and is set to a value between 2 and 5 but nominally set to 3. A fundamental flaw with this method of reducing the transmitted bit rate, is that if the  $(l * R_f) > 1$  then the new transmitted bit rate will be negative. To overcome this problem, the minimum transmitted bit rate,  $r_i$ , is bounded to 10kbps for these tests, which corresponds to the lowest quality that will be sent, that is, a frame rate of 5fps and spatial resolution of 60%.

To demonstrate the integration of the OAT with the LDA algorithm, the clip named, LDA\_TestSeq, was used. The tests shall demonstrate adaptive streaming of a pre-encoded multi-tracked content in the adaptation dimensions of frame rate and spatial resolution. The bit rate range at various encoding configurations of frame rate and spatial resolution for this clip is shown in Figure 6.5. The content is composed of 5 tracks each representing the content encoded with a different spatial resolution and a frame rate of 25fps. The OAT through adaptation space is continuous but must be quantized to allow for track switching as shown in Figure 6.6.

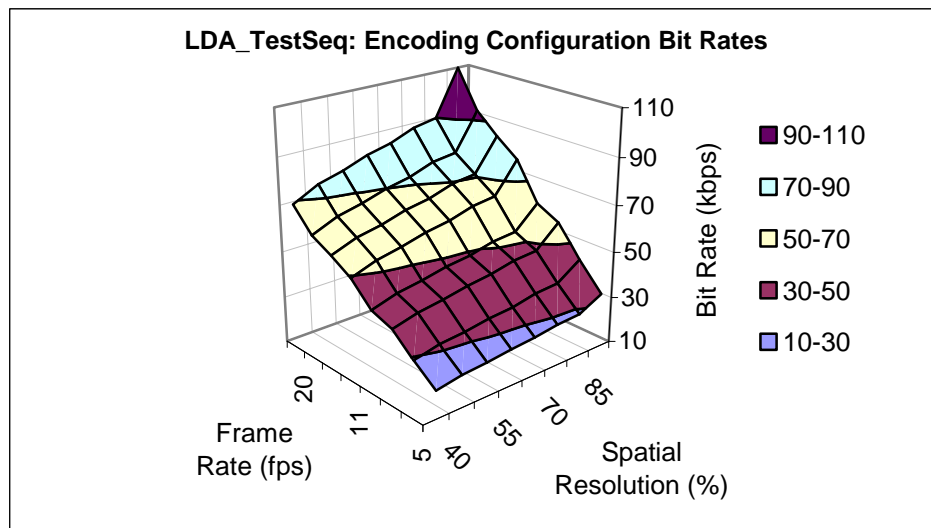


FIGURE 6.5: LDA\_TESTSEQ BIT RATE PLANE

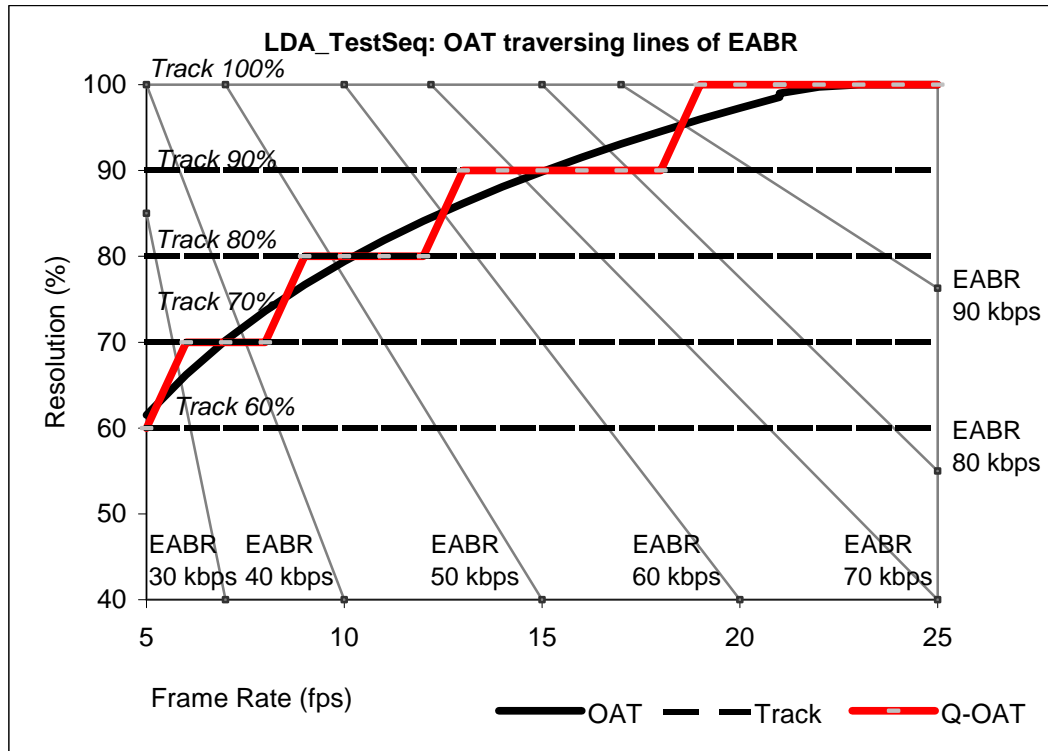


FIGURE 6.6: LDA\_TESTSEQ OAT AND EABR

### 6.3.2. LDA System Architecture

The prototype system consists of a client and server (Figure 6.7). The client returns standard RTCP-RR feedback containing information such as the loss, delay and bottleneck bandwidth values. When the server receives feedback from the client, the LDA algorithm indicates how the bit rate should be adjusted in response to fluctuations in the available end-to-end bit rate between client and server. The server finds the corresponding EABR zone, which contains this new transmission rate. Within this EABR zone, the server finds the corresponding encoding configuration on the OAT indicating the quality-encoding configuration that maximizes the user-perceived quality. Having found this new encoding configuration, the server adjusts the frame rate and/or adapts the resolution by switching tracks to achieve this new encoding configuration.

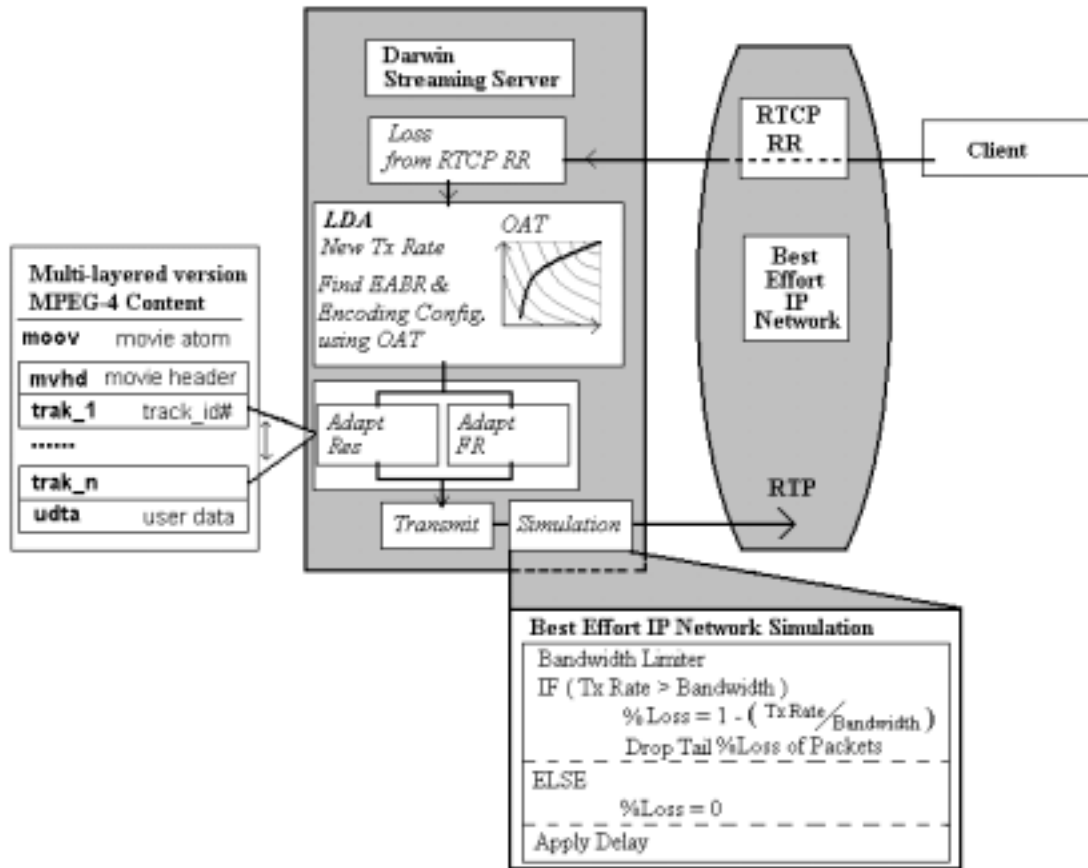


FIGURE 6.7: LDA SYSTEM ARCHITECTURE

### 6.3.3. LDA Simulations

Two very different network conditions have been selected to demonstrate the efficacy and use of OATs with LDA. The first is typical of a good fixed Internet connection where the client experiences relatively small fluctuations in their available bit rate. The second example is more typical of a wireless IP network where mobility can result in sudden and substantial changes in the available bit rate. In the simulations, RTCP feedback was fixed at every 5 seconds. In the examples below, Figures 6.8.A and 6.9.A show how the server's transmission rate (TX\_Rate) adapts in response to the available bandwidth (Avail\_BW) at the client using the LDA algorithm. This transmission rate gives the EABR zone required and the spatial resolution and frame rate are then determined from the OAT.

The simulation environment was built into the prototype server and so there is greater control and reproducibility of the simulated network tests. When the server is streaming the transmission bit rate is monitored. The server compares its transmission bit rate with the bandwidth at a particular instant. If the transmission bit rate is greater than the simulation bandwidth, the server drops a percentage of packets in groups to simulate a drop-tail queue

that will typically occur at the bottleneck router. The packets that are not dropped are passed to a separate thread, which will delay the actual transmission of the packets to simulate network delay. The client and server run on a local loop. The RTCP feedback frequency is fixed at 5-second intervals. In the simulations, the server has perfect knowledge of the bottleneck bandwidth and is determined from the available bandwidth at a particular instant.

Figures 6.8.C, 6.8.D, 6.9.C and 6.9.D show how the spatial resolution and frame rates are adapted. Adaptation occurs in both dimensions of frame rate and spatial resolution simultaneously as dictated by the OAT. In particular the quantised spatial resolution (Q\_Res) implies that track switching has been performed. Track switching is a computationally lightweight process and so does not hinder the performance of the server.

*Example 1: Relatively stable network conditions*

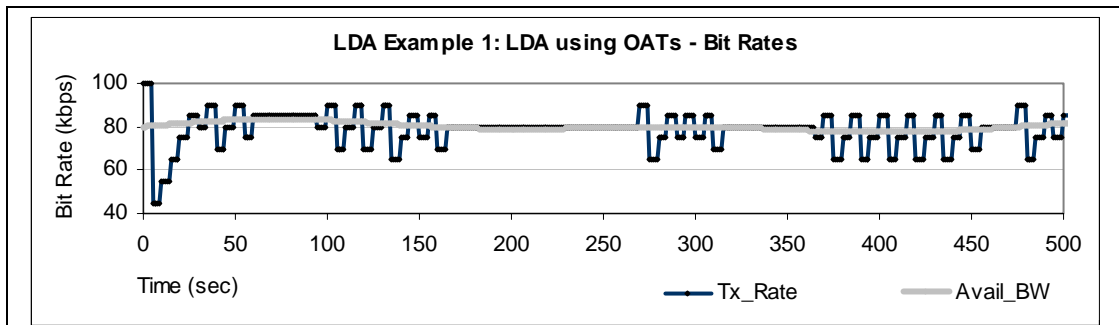


FIGURE 6.8.A: LDA EXAMPLE 1 - BIT RATE VARIATIONS

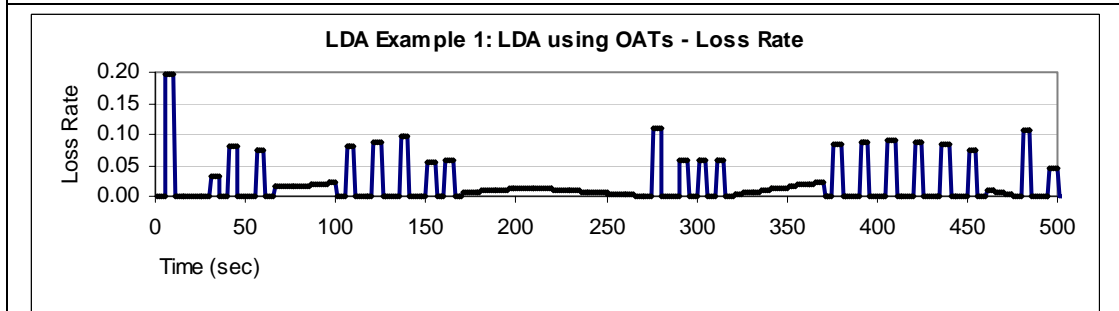


FIGURE 6.8.B: LDA EXAMPLE 1 - LOSS RATE VARIATIONS

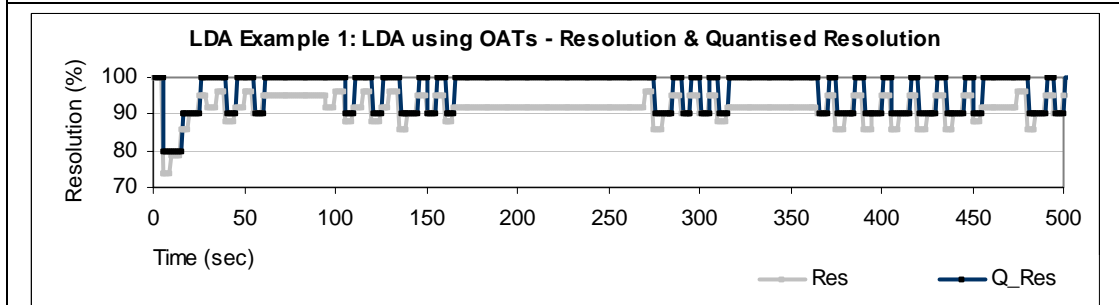


FIGURE 6.8.C: LDA EXAMPLE 1- ADAPTATIONS IN RESOLUTION USING THE OAT

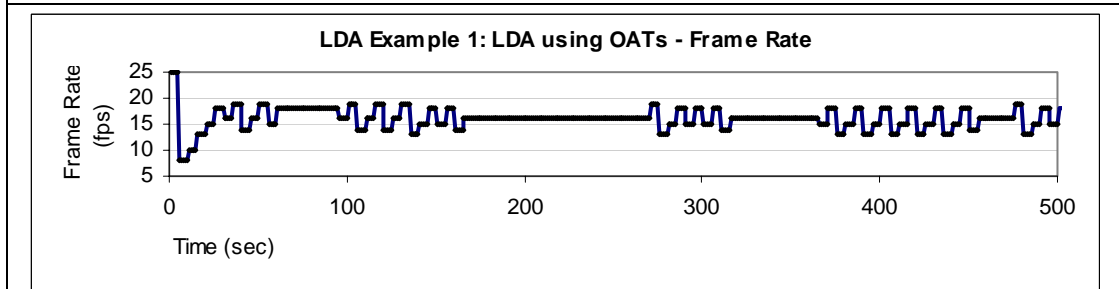


FIGURE 6.8.D: LDA EXAMPLE 1- ADAPTATIONS TO FRAME RATE USING THE OAT

FIGURE 6.8: LDA EXAMPLE 1

Example 2: Rapidly fluctuating network conditions

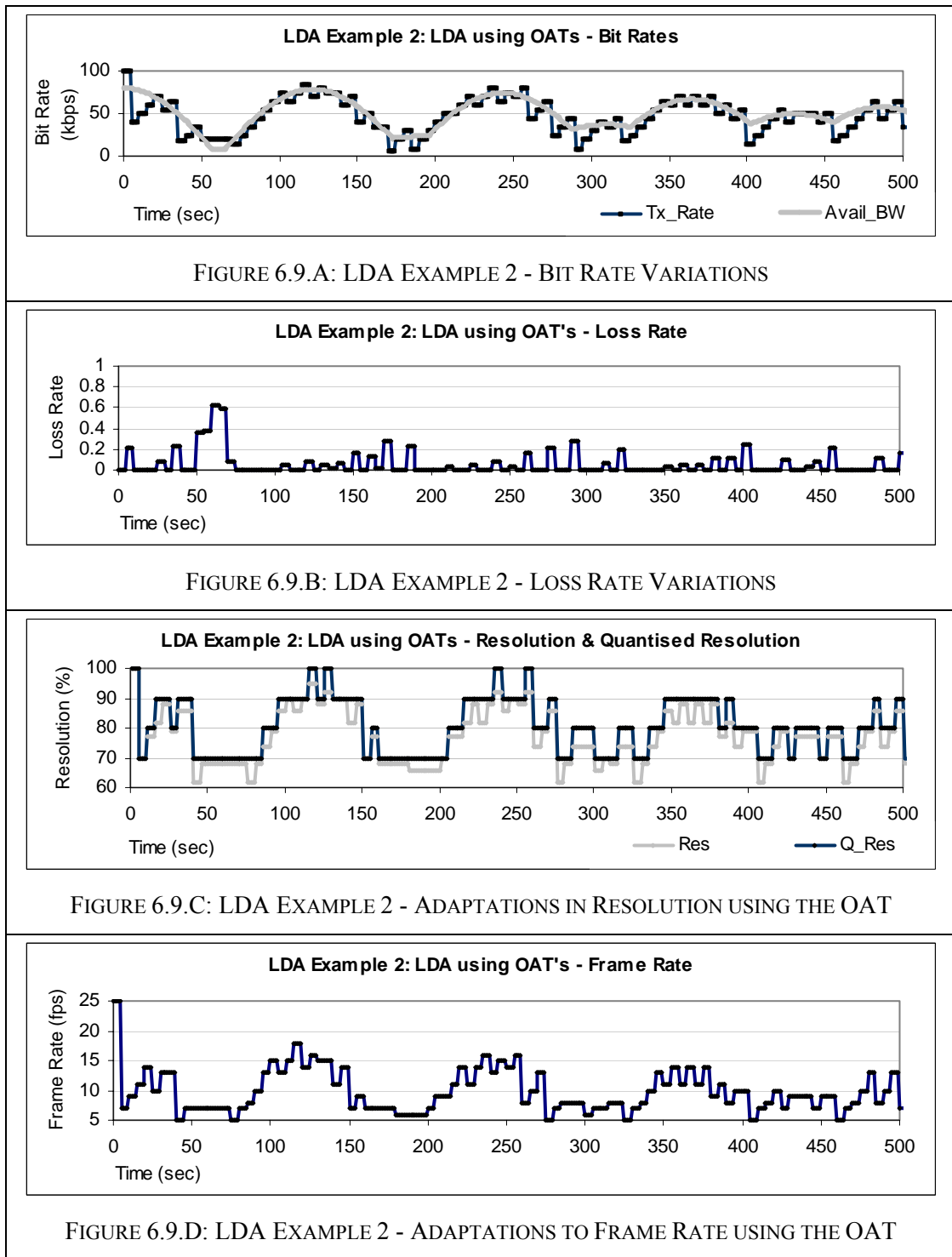


FIGURE 6.9.A: LDA EXAMPLE 2 - BIT RATE VARIATIONS

FIGURE 6.9.B: LDA EXAMPLE 2 - LOSS RATE VARIATIONS

FIGURE 6.9.C: LDA EXAMPLE 2 - ADAPTATIONS IN RESOLUTION USING THE OAT

FIGURE 6.9.D: LDA EXAMPLE 2 - ADAPTATIONS TO FRAME RATE USING THE OAT

FIGURE 6.9: LDA EXAMPLE 2



#### 6.3.4. Discussion

It can be seen from example 1, when there is no reported loss, the algorithm probes the available bandwidth by increasing the transmission rate, which causes periodic loss bursts. However, this behavior may be desirable when the network conditions are rapidly fluctuating as in example 2. In this case, LDA performs particularly well by matching the transmitted bit rate to the available bandwidth in the network. The efficacy of LDA depends on the algorithm control parameters, the reduction factor and the bottleneck bandwidth. The reduction factor has been empirically optimized for a value of 3. The bottleneck bandwidth is estimated using a packet-pair approach. The packet-pair approach assumes that the second packet is queued behind the first at the bottleneck link but if the probe packets were not sent fast enough, there might not be any queuing at the bottleneck link. This approach also assumes that both packets travel along the same route however this is not always the case as the probe packets can be dropped, take different routes and arrive out-of-order.

These examples demonstrate the feasibility of using the OAT to complement any existing sender-based transmission adaptation algorithm. This is expected to enhance the user perception of the adaptation as the quality is degraded and upgraded in a known and controlled manner that has the least negative impact on the perceptual quality of the content. However, in general most sender-based algorithms attempt to reduce the loss rate to zero regardless of the perceptual effects of any loss. In the next section, an algorithm, Perceptual Quality Adaptation (PQA) algorithm, has been developed that attempts to reduce the loss to the point of tolerable loss. PQA uses feedback to infer the clients' playout quality using RTCP-RR feedback or explicitly gets the clients' playout quality from RTCP-APP feedback. Using this knowledge, the server can make more intelligent adaptation decisions.

## 6.4. Perceptual Quality Adaptation (PQA)

Adaptation techniques should attempt to reduce network congestion and packet loss by matching the rate of the video stream to the available network bandwidth. Without adaptation, any data transmitted exceeding the available bandwidth could be discarded, lost or corrupted in the network. This has a devastating effect on the playout quality of the received stream. Adaptation techniques can be classified by the method in which the adaptation occurs. These are:

- Rate Control: Transport level
  - Sender-based
  - Client-based
  - Hybrid
- Rate Shaping: Transport and encoding level
  - Sender-based
  - Client-based
  - Hybrid
- Encoder-based adaptation: Encoding level
  - Sender-based
- Transcoder: Network level.

In this section a new type of adaptation algorithm, PQA, is proposed that uses the OAT as a basis for making adaptation decisions. This algorithm does not fall into any of the conventional adaptation technique classifications and as such could be termed a Perceptual Quality adaptation technique, which can take place at either the encoder level or sender level. Adaptation at the encoder level requires a real-time encoder capable of adapting the video stream in terms of spatial resolution and frame rate or the sender level. Adaptation at the sender level uses track switching and frame dropping implemented in the streaming server to adapt the quality of the delivered video. In this section, PQA is demonstrated to work at the sender level for pre-encoded multi-tracked MPEG-4 content. However, PQA could work equally well with a live encoder that can adapt the frame rate and spatial resolution in real-time. PQA attempts to match the transmitted stream to the playout capabilities of the client rather than the network capacity of the client.

### 6.4.1. PQA System Overview

The prototype system was developed and designed to adapt the transmitted video stream based on either RTCP-RR and/or RTCP-APP feedback received from the client (Chapter 6 Section 1.2). This feedback enables the server to evaluate the user-perceived-quality of the

received stream. The perceptual quality of the received stream is obtained directly from RTCP-APP feedback or else must be inferred from the standard RTCP-RR feedback, in particular, the loss rate. Using the loss rate, the server can estimate the playable frame rate of the received stream based on packet loss probabilities and how they relate to the encoding configuration of the content in terms of its GOV sequence and hint track settings.

Rather than reducing the loss rate to zero, which could lead to an excessive degradation of the transmitted quality, the aim is to gain a better insight into the quality of the clients' playout and reduce the loss rate to the point of tolerable loss, which is determined by the robustness of the encoding. For example, most sender-based adaptation algorithms insist that the bit rate be reduced when the loss rate is above zero regardless of whether these errors/losses could be masked or recovered. Thus, if RTCP-APP feedback indicates that the playable frame rate equals the expected frame rate, then, regardless of the loss rate reported, the server should upgrade along the OAT as though the loss rate is zero. Another advantage of using the RTCP-APP feedback is that there may be factors other than network losses, which may contribute to frames being discarded or dropped by the client, for example, buffer overflow, so even if the loss rate reported by the RTCP-RR is zero, there is still a degradation of quality at the client side.

- **RTCP-APP:** Server specific clients who send the RTCP-APP feedback containing information about the expected frame rate and the actual frame rate of the streamed content on the client player.
- **RTCP-RR:** RTCP-RR feedback contains information about the fraction of packets lost in the RTCP interval. Using this, the server makes inferences to approximate the playable frame rate of the clients.

When the server receives feedback from the client, the server determines the quality of the clients' playout, either by using RTCP-APP feedback directly or else by making inferences using RTCP-RR feedback (Figure 6.10). The server uses the PQA algorithm to make adaptation decisions. The PQA algorithm is essentially the OAT, with two operating points, the transmitted encoding configuration and the clients' playout capability. The server uses feedback to determine the clients' playout capability on the OAT. During adaptation, the server adjusts the transmitted encoding configuration to match the clients' playout capability. By adapting the encoding configuration being sent, this manifests itself in an increase/decrease in the transmitted bit rate.

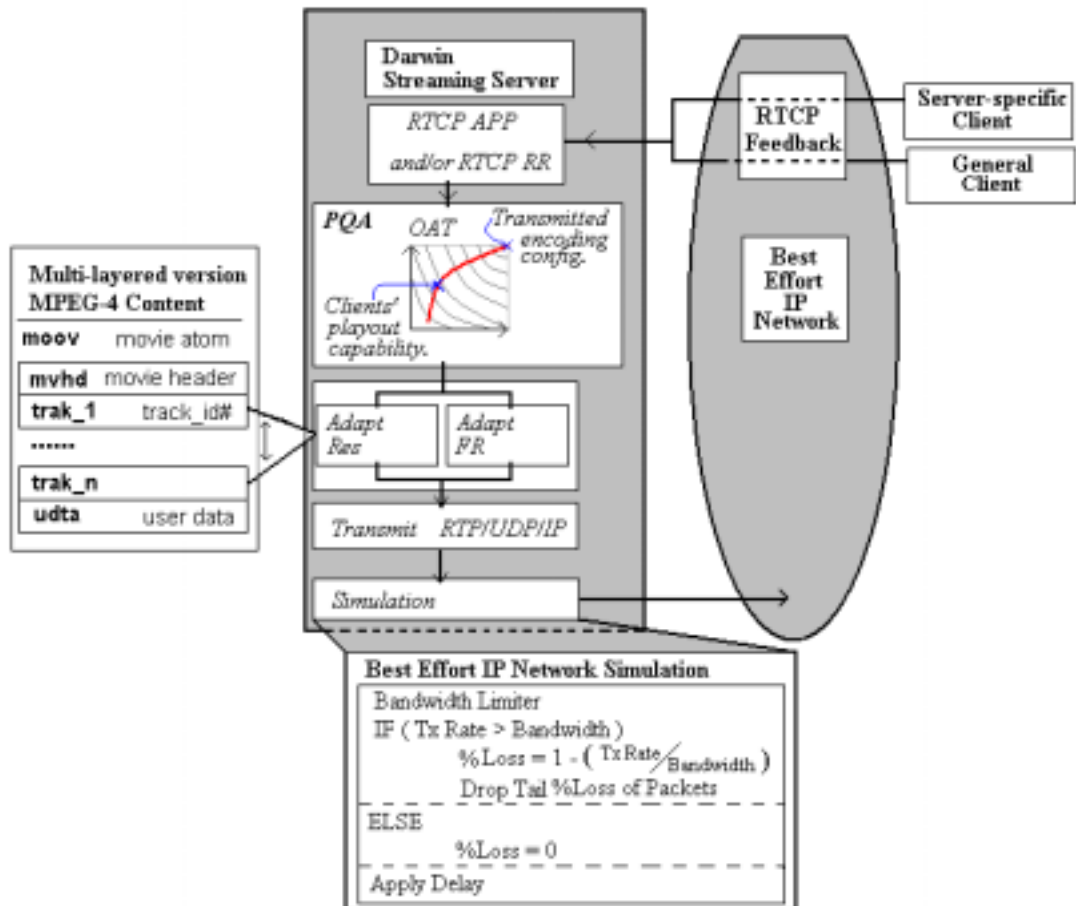


FIGURE 6.10: PQA SYSTEM ARCHITECTURE

#### 6.4.2. RTCP-APP: Explicit Quality Feedback

For server-specific clients, an RTCP-APP packet has been defined (Figure 6.11) [189]. The application specific data is transmitted in text form, which is then parsed by the DSS. The RTCP-APP packet is intended for experimental use as new applications and new features are developed, without requiring packet type value registration. RTCP-APP packets are transmitted periodically with RTCP-RR reports. The function of RTCP-APP is to convey application specific feedback to the server. This feedback gives the server a better insight into the level of QoS that the client is experiencing and make better, more intelligent adaptation decisions. For example, consider a client sending conventional RTCP-RR reports indicating that there are no packet losses in the network. The server may increase the transmission rate assuming that the client can handle and playout the entire stream being transmitted. However, there may be other sources of packet loss, for example, at the clients' buffer. RTCP-APP feedback can indicate to the server that the clients' buffer cannot handle the incoming traffic and packets are being dropped at the clients' buffer. Thus, the server should reduce the transmission rate to match the both network level and device level

constraints of the client.

The information fields are as follows:

- 'rr': Receiver Bit Rate
- 'lt': Average Late Milliseconds
- 'ls': Percent Packets Lost
- 'dl': Average Buffer Delay (Milliseconds)
- 'pr': Total Packets Received
- 'pd': Total Packets Dropped
- 'pl': Total Packets Lost
- 'bl': Client Buffer Fill
- 'fr': Frame Rate
- 'xr': Expected Frame Rate
- 'd#': Audio Count
- 'ob': Over-buffer Window Size

The RTCP-APP packet reports back the servers transmitted frame rate and the actual frame rate as played out by the client having accounted for errored and discarded frames.

0				1				2				3															
0							8							16							24						
V=2	P	Subtype		PT=APP=204				Length																			
SSRC/CSRC																											
Name (ASCII)																											
Application specific data																											

FIGURE 6.11: RTCP-APP PACKET STRUCTURE

### 6.4.3. RTCP-RR: Inferring the Playable Frame Rate

In MPEG-4 the I-VOP's (Video Object Planes) are intra-coded images, P-VOP's are predictively encoded from the I-VOP and B-VOP's are bi-directionally encoded frames. It is extremely difficult to take account of error correction techniques in a procedure to infer the frame loss rate from the packet loss rate. In the absence of RTCP-APP feedback, the following equations are used to infer the playable frame rate when network losses occur [190]. These are based on the assumption that if a frame contains one or more errored/lost packets, then that frame will be unplayable.

Let,

$F_0$  = Frame rate of the stream delivered.

$F$  = Playable frame rate discarding unplayable frames due to single packet losses.

$S_I$  = Average number of packets in an I-VOP.

$S_P$  = Average number of packets in a P-VOP.

$S_B$  = Average number of packets in a B-VOP.

$N_P$  = Number of P-VOP's in a GOV.

$p$  = Packet loss rate.

$\phi$  = Frame drop rate.

$P(i_f)$  = Probability of a VOP of type  $i$ .

$P(\bar{F} | f_i)$  = Probability of an error in a VOP of type  $i$ .

Then the playable frame rate is related to the original frame rate being sent and the frame drop rate,  $\phi$ .

$$F = F_0(1 - \phi)$$

Where the frame drop rate,  $\phi$ , is given by,

$$\phi = \sum_i P(f_i) \cdot P(\bar{F} | f_i)$$

The probability of an error in each VOP type is related to the number of packets needed to transmit the VOP; the probability and frequency of the VOP; any inter-dependencies on other VOP's for the successful decoding of this VOP and the probability of one of these packets being lost.

The probability of an error in an I-VOP is:  $P(\bar{F} | I) = 1 - (1 - p)^{S_I}$

The probability of an error in a P-VOP is:  $P(\bar{F} | P) = \frac{1}{N_P} \sum_{k=1}^{N_P} (1 - (1 - p)^{S_I + kS_P})$

Closed-form expression:

$$P(\bar{F} | P) = 1 - \left[ \frac{(1 - p)^{S_I}}{N_P (1 - (1 - p)^{S_P})} \right] \left[ 1 - (1 - p)^{S_P N_P} \right]$$

The probability of an error in a B-VOP is:  $P(\bar{F} | B) \leq \frac{1}{N_P} \sum_{k=1}^{N_P} (1 - (1 - p)^{S_I + (k+1)S_P + S_B})$

Closed-form expression:

$$P(\bar{F} | B) \leq 1 - \left[ \frac{(1 - p)^{S_I + S_P + S_B}}{N_P (1 - (1 - p)^{S_P})} \right] \left[ 1 - (1 - p)^{S_P N_P} \right]$$

**Example:**

$$F_0 = 24\text{fps.}$$

$$S_I = 3 \text{ packets.}$$

$$S_P = 2 \text{ packets.}$$

$$S_B = 1 \text{ packets.}$$

Frame sequence: I PBB PBB PBB I PBB PBB PBB I PBB

GOV Sequence: I PBB PBB PBB

$$N_P = 3 \text{ P-VOP's.}$$

$$p = 1\%$$

$$P(I) = \frac{3}{25} = 0.12 \text{ and } P(\bar{F} | I) = 0.029$$

$$P(P) = \frac{7}{25} = 0.28 \text{ and } P(\bar{F} | P) = 0.058$$

$$P(B) = \frac{15}{25} = 0.60 \text{ and } P(\bar{F} | B) = 0.086$$

$$\phi = \sum_i P(f_i) \cdot P(\bar{F} | f_i) = P(I) \cdot P(\bar{F} | I) + P(P) \cdot P(\bar{F} | P) + P(B) \cdot P(\bar{F} | B)$$

$$\phi = 0.071$$

$$F = F_0(1 - \phi) = 23.217 \text{ fps Playable frame rate after discarding errored frames.}$$

The server can predict the playable frame rate at the client using knowledge of the video being streamed such as the reported loss rate from the client, the number of packets per VOP type and the GOV sequence. As expected, the playable frame rate decays exponentially with increasing loss rates (Figure 6.12). Error propagation in the video at loss rates above 20% can render the video unplayable.

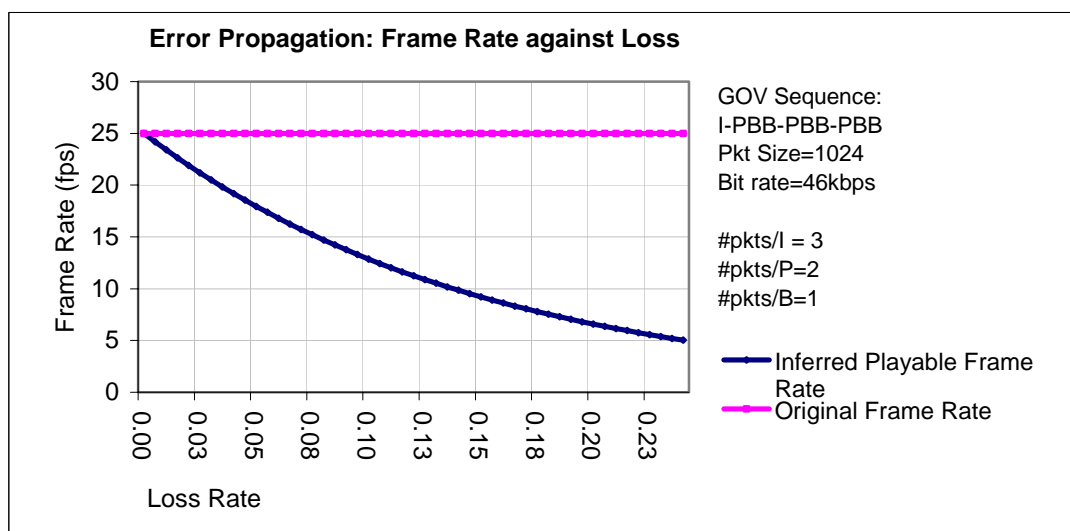


FIGURE 6.12: PLAYABLE FRAME RATE WITH LOSS

To increase the accuracy of this inference mechanism the server monitors the encoding characteristics and delivery of the content throughout the streaming process. When a file contains multiple video tracks, these may have different characteristics as indicated by their corresponding hint track. The server monitors the number of packets required to send each type of VOP, as a track with a hint track stating a maximum packet size of 512 bytes would require sending more packets per VOP than a track whose hint track indicated a maximum packet size of 1024 bytes.

- **Theoretical I-VOP Frequency:** The theoretical I-VOP frequency can be determined using the hint track associated with the video track. The Trak Meta information contains an atom called *stss*, which contains the sample numbers of all the sync frames (I-VOPs) contained in the video track. By dividing the movie duration by the number of sync samples in the track we can determine the theoretical I-VOP frequency.
- **Theoretical P and B-VOP Frequency:** The theoretical P and B VOP frequency can be determined using the *stsz* atom in the hint track. The *stsz* atom contains meta-information about all the samples or VOP's in the video track.

The problem with using the theoretical I, P and B VOP frequencies in the inference above is that videos may not have a uniform distribution of I, P or B frames and thus the frequency of I-VOP's may not be consistent throughout the video. For example, some encoders allow for an I-VOP to be generated only when there is a scene change. The video track may not use the standard GOP structure. In the worst-case scenario the above assumptions are used for inferring the playable frame rate using the theoretical VOP frequencies. Similarly, hint tracks may have different parameters, for example, different MTU's that would in turn affect the number of packets required to send each VOP.

- **Actual VOP Frequency:** As packets are sent from the active video track, the server monitors the first packet in a group of packets for each VOP and gets the VOP frequency by counting the number of VOP's sent for this track since the track was activated.
- **Actual Number of Packets Per VOP:** As the server packetizes VOP's, it keeps a count of the actual number of packets used to send each VOP type.

In general, at low loss rates, the theoretical inference correlates well to the playable frame rates as reported in the RTCP-APP feedback and estimates the frame rate as being approximately 2 frames less than the reported frame rate. This is to compensate for various error correction and error resilience schemes of the encoding, which can help recover frames even if some data is lost. However, as the loss rates increase to about 20%, the inference begins to differ greatly from the reported frame rate in RTCP-APP feedback.



#### 6.4.4. PQA Algorithm

When the server receives feedback from the client, the server determines the playable frame rate, either by using RTCP-APP feedback directly or else by making inferences using RTCP-RR feedback (Figure 6.13). The server checks if the playable frame rate equals that being sent, if so, the server increases the frame rate by 1fps. However, if the playable frame rate is less than that being sent, the server sets the new frame rate to be equal to the playable frame rate. The OAT is used by the prototype server to calculate the new encoding configuration in terms of spatial resolution and frame rate that has the least negative impact on the quality perceived by the client. Using the new frame rate, the server finds the track that has this corresponding value for spatial resolution by referring to the OAT. If it is different to the current track, the server simply switches tracks.

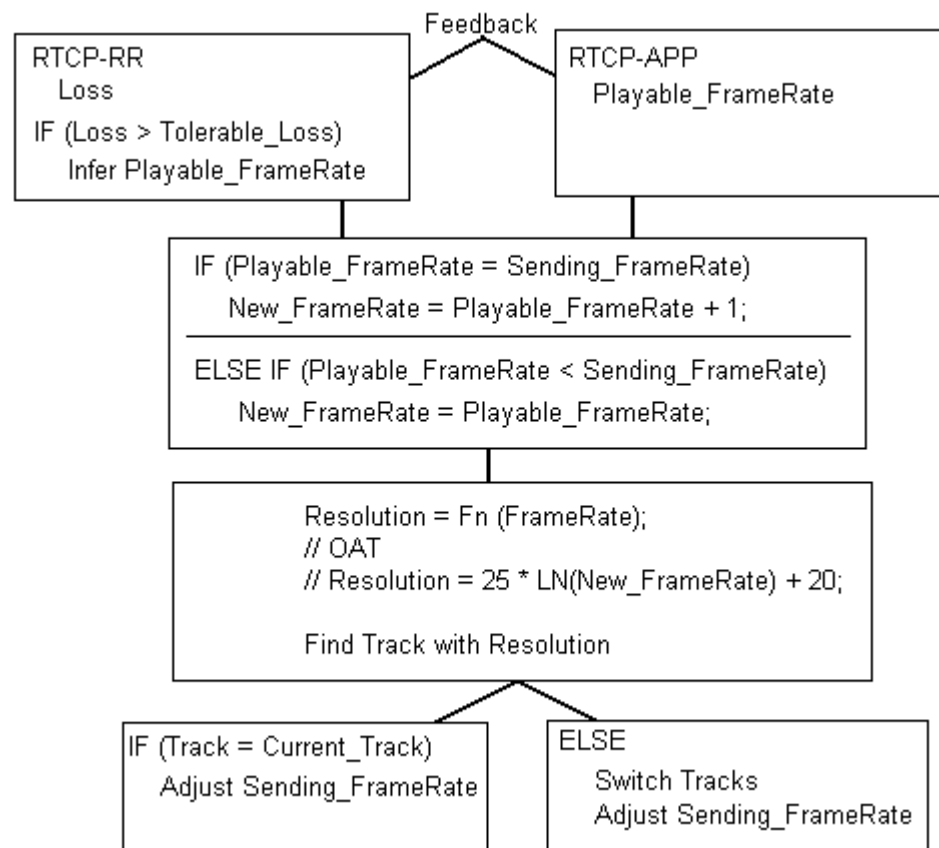


FIGURE 6.13: PQA ALGORITHM

### 6.4.5. PQA Example Operation

*Loss = 0%*

If the loss rate is zero, there are two scenarios in which the quality can be increased (Figure 6.14.A)

**[A]** The server is currently delivering a stream at 23fps from the 100% spatial resolution track, T1. RTCP feedback indicates that there is no loss. The server increases the frame rate to 24fps and finds the intersection point with the OAT at a spatial resolution of 98%. The server then finds the track from the appropriate switch group that most closely matches the spatial resolution i.e.T1. But as the current track is the same as that found by the PQA algorithm, no track switching is required.

**[B]** The server is currently delivering a stream at 13fps and spatial resolution of 80% in T3. RTCP feedback indicates that there is no loss, so the server increases the frame rate to 14fps and finds the intersection point with the OAT at a spatial resolution of 86%. The track that most closely matches this spatial resolution is T2. In this case, the server switches tracks from T3 to T2 seamlessly.

*Loss > 0%*

If the loss rate is greater than tolerable loss rate, for example, zero, there are two scenarios in which the quality can be decreased (Figure 6.14.B)

**[C]** The server is currently delivering at 24fps and spatial resolution of 100% in T1. The client reports back either through RTCP-APP or inferred using RTCP-RR that the playable frame rate is just 22fps and the server reduces the frame rate being delivered to 22fps. The server finds the intersection of 22fps with the OAT at 97% spatial resolution. The track with the closest spatial resolution is T1, which is the current track, so no track switching is required.

**[D]** The server is streaming from T1 at 22fps and 100% spatial resolution, however, the loss persists and this time, the server determines the playable frame rate is 20fps, which intersects the OAT at a resolution of 94%. The track with a resolution nearest to 94% is the track with 90% resolution, T2. This is different to the current track, so the server switches tracks seamlessly to T2.

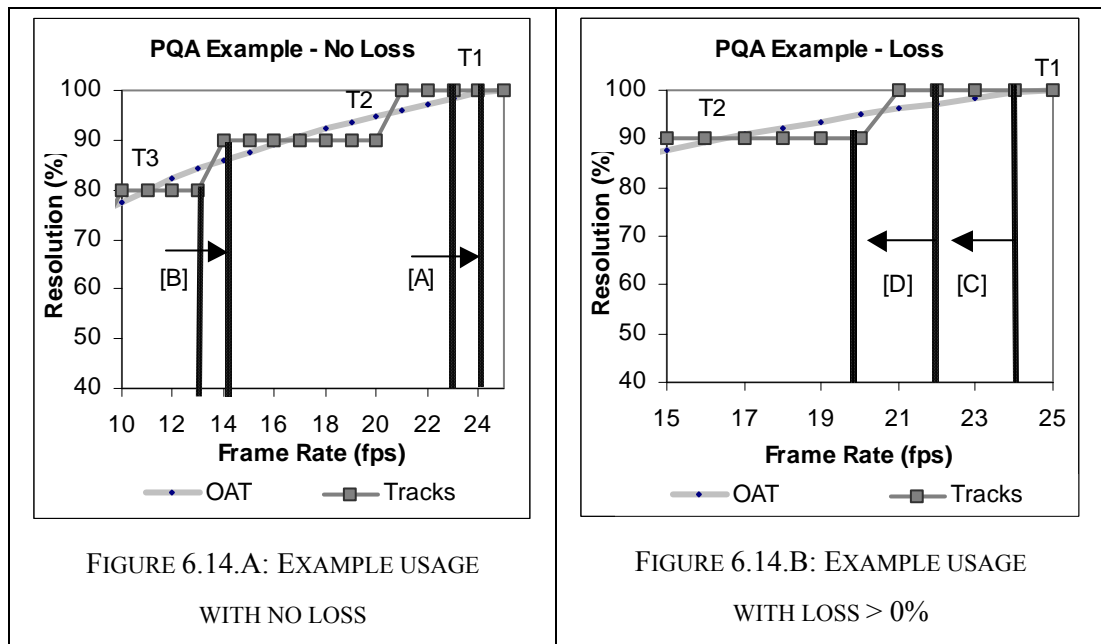


FIGURE 6.14: EXAMPLE USAGE OF PQA

#### 6.4.6. PQA Simulation

The benefits of using the PQA algorithm, is that the reduced bit rate is achieved in terms of real encoding parameters. Consider a system using another sender-based adaptation algorithm, Loss-Delay Algorithm (LDA); in instances of loss, the server is required to adjust its transmission bit rate by a very precise value. For example, if the server was sending at a bit rate of 50kbps but received feedback indicating loss, the algorithm might suggest a new bit rate of 49kbps. This reduction of 1kbps may not be achievable in terms of the available encoding parameters.

Two very different network conditions have been selected to demonstrate the behavior of the PQA algorithm. The first is typical of a good fixed Internet connection where the client experiences relatively small fluctuations in their available bit rate. The second example is more typical of a wireless IP network where mobility can result in sudden and substantial changes in the available bit rate. In the simulations, RTCP feedback was fixed at every 5 seconds. In the examples below, Figures 6.15.A and 6.16.A show how the server's transmission rate (TX\_Rate) adapts in response to the available bandwidth (Avail\_BW) at the client using the PQA algorithm. This transmission rate gives the EABR zone required and the spatial resolution and frame rate are then determined from the OAT. The simulation environment setup is the same as that used in the LDA tests (Chapter 6, Section 3.3).

Figures 6.15.C, 6.15.D, 6.16.C and 6.16.D show how the spatial resolution and frame rates are adapted. Adaptation occurs in both dimensions of frame rate and spatial resolution

simultaneously as indicated by the OAT. The spatial resolution (Res) is obtained from the OAT and is a function of the playable frame rate. The content used for this test has 5 tracks and so the spatial resolution is quantised (Q\_Res). Track switching matches the resolution determined by the OAT (Res) to the track that most closely matches the spatial resolution (Q\_Res).

Example 1: Relatively Stable Network Conditions

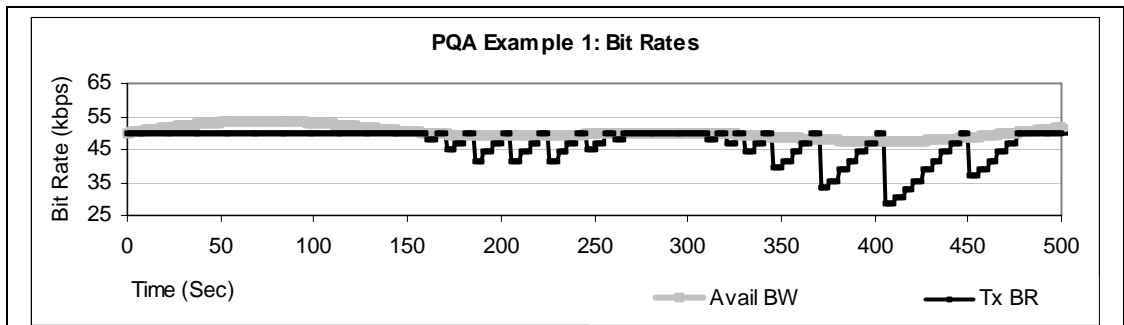


FIGURE 6.15.A: PQA EXAMPLE 1: BIT RATES VARIATIONS

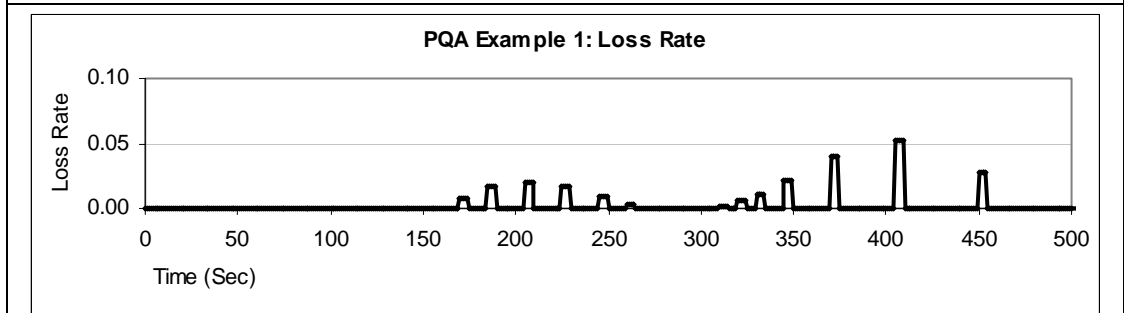


FIGURE 6.15.B: PQA EXAMPLE 1: LOSS RATES VARIATIONS

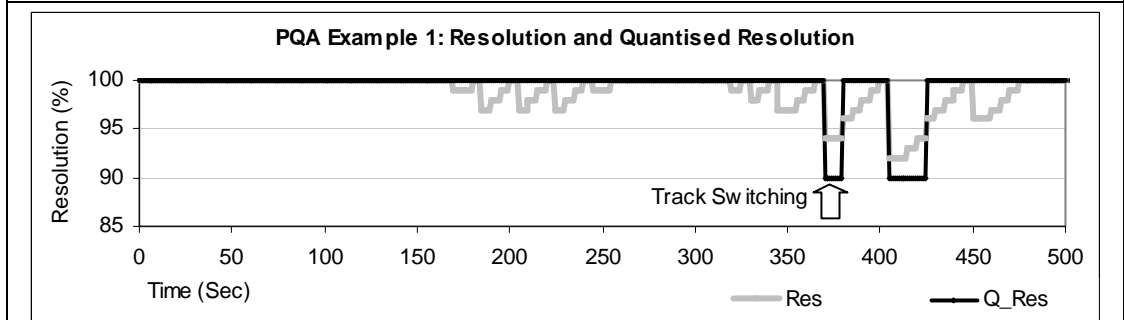


FIGURE 6.15.C: PQA EXAMPLE 1 - SPATIAL RESOLUTION ADAPTATIONS

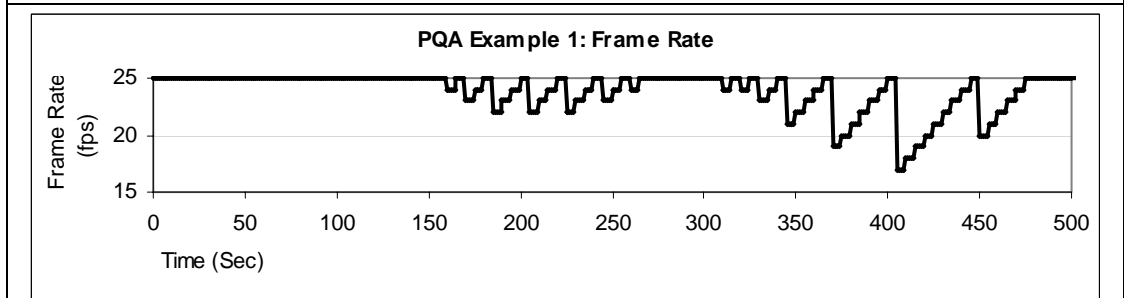


FIGURE 6.15.D: PQA EXAMPLE 1 - FRAME RATE ADAPTATIONS

FIGURE 6.15: PQA EXAMPLE 1

*Example 2: Rapidly fluctuating network conditions*

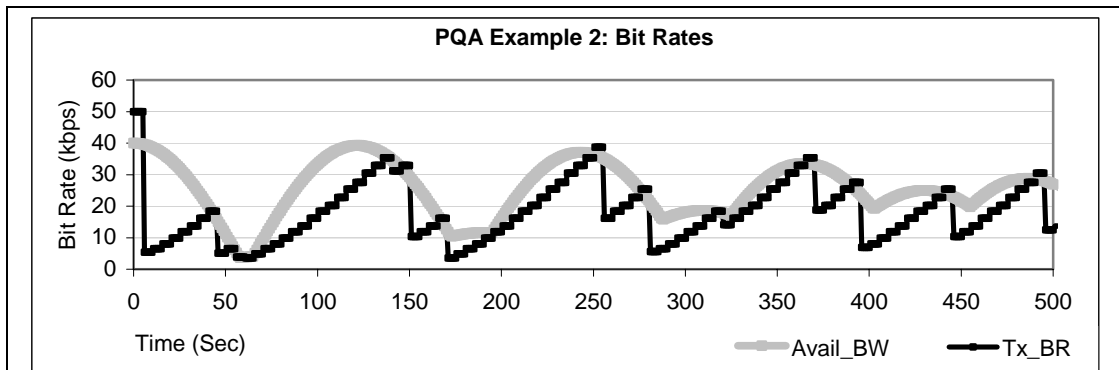


FIGURE 6.16.A: PQA EXAMPLE 2 - BIT RATES VARIATIONS

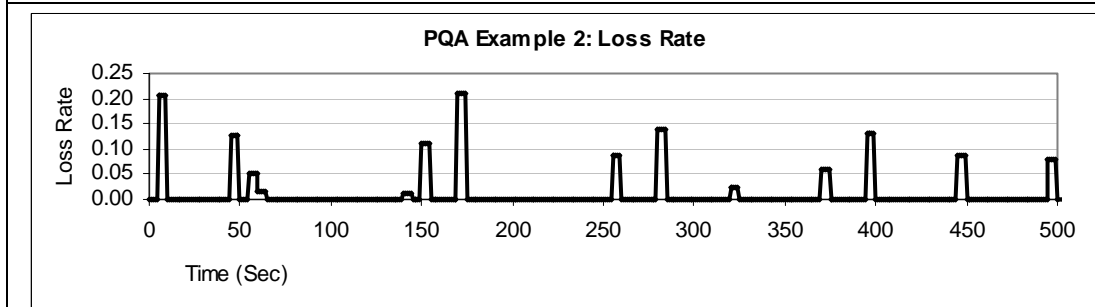


FIGURE 6.16.B: PQA EXAMPLE 2 - LOSS RATES VARIATIONS

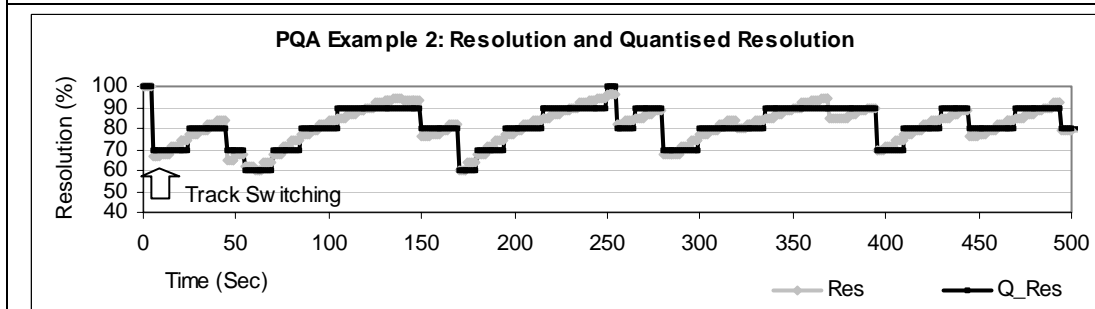


FIGURE 6.16.C: PQA EXAMPLE 2 - SPATIAL RESOLUTION ADAPTATIONS



FIGURE 6.16.D: PQA EXAMPLE 2 - FRAME RATE ADAPTATIONS

FIGURE 6.16: PQA EXAMPLE 2

### 6.4.7. Discussion

It can be seen from the simulation results that the PQA algorithm operates like a typical Additive Increase Multiplicative Decrease (AIMD) algorithm whereby the transmitted encoding configuration is reduced dramatically when the clients' playout capabilities are reduced and increases tentatively over time as the clients' capabilities improve. The main benefit of the PQA algorithm is that it is aware of the perceived quality and adapts using the OAT. PQA suggests a reduced quality-encoding configuration, which results in a reduced bit rate. As PQA is using the OAT as a basis for quality adaptation, quality adaptation occurs in a manner, which has the least negative impact on the users perception. PQA behaves in quite a different way to LDA. LDA adapts the transmitted bit rate in response to bandwidth fluctuations. PQA is not bit rate focused, but rather quality oriented and adapts the transmitted quality to match the playout capabilities of the client application.

The PQA algorithm reacts severely when loss occurs by reducing the transmitted encoding configuration in an AIMD manner. The transmitted encoding configuration is reduced to match the estimated clients playout capability. The PQA algorithm can be improved and extended in many ways. The responsiveness of the algorithm is entirely dependent on the can be improved to allow the system to react quicker to changes in the network conditions. The system reacts on immediate feedback and reacts quite harshly and severely to loss. It may be more efficient to monitor long-term and short-term network conditions and make a softer adaptation decisions.

**Algorithm Responsiveness:** The responsiveness of the algorithm is dependent on the RTCP feedback frequency. One possibility to improve the responsiveness of the algorithm is to increase the feedback frequency. There is work in progress to standardise a method to increase the RTCP feedback frequency to accommodate very low bit rate streaming sessions. SDP bandwidth modifiers can be used to increase the flexibility in the frequency of RTCP feedback so applications can adapt more quickly and readily. With increased feedback frequencies, the server can adapt more readily and quickly to congestion in the network.

**Feedback monitoring:** Monitoring the long term and short network connection characteristics may prevent the system over-reacting on bursty losses. For example, if a connection experiences a random bursty loss, the current system will react regardless based on this loss assuming that the quality of the connection has been degraded. However, this bursty loss may be caused some short-lived effect that is not related to congestion, for

example, the user moving to an area of low connectivity or coverage. There are several adaptation algorithms that use this notion of long-term and short-term monitoring to make more efficient adaptation decisions. Rate Adaptation protocol (RAP) uses both short-term and long-term estimates of a connections' RTT to make adaptation decisions. Similarly, the Direct Adjustment Algorithm, (DAA), uses an EWMA to avoid reactions to sudden bursty losses. PQA behaves in an AIMD fashion. It additively increases its encoding configuration and is slow to return to its maximum encoding, however, upon adaptation it multiplicatively decreases the encoding configuration. To prevent unnecessary adaptation which may be sudden and abrupt particularly when multiplicatively decreasing the quality, filtered responses using short and long-term feedback should improve the systems adaptation strategy.



## 6.5. PQA in Wireless Networks

PQA performs particularly well for large bandwidth fluctuations making it more amenable to operate in wireless networking environment. A typical user of a wireless streamed video application will be either static, for example, sitting down in a café, waiting in a queue etc. or the user will be travelling in some form of transport such as in a car, train etc. This means that the velocities of the potential users will be range from very low velocity to very high velocities. At high velocities there will be many handovers and the channel quality will fluctuate greatly.

### 6.5.1. Issues in Wireless Streaming

Streaming applications face several challenges when it comes to the wired and wireless environment [191][192].

Wired networking issues include:

1. Bandwidth fluctuations.
2. Loss.
3. Timeliness (strict delay constraints and jitter).
4. Reliability.

Wireless networking issues include:

1. The coverage area of the base stations being used.
2. Speed of mobility supported by the given technology.
3. Increased delays due to ARQ retransmissions.
4. The available spectrum is a scarce resource. Each channel uses a given range of frequencies and shares the usage with those frequencies with other users within the same cell. Increased usage leads to an increase in congestion and a decrease in communication quality (e.g. increased bit error rates) and reduced bit rates.
5. Size and energy consumption depends on the device being used. For example, laptops, PDAs, cellular phones affect the applications and service requirements. Energy consumption is a major problem for mobile devices. Heavy load applications such streaming services have a much higher energy consumption, which affects the devices ability to maintain the application.

*Wireless Network Testing Assumptions*

1. Energy consumption is not an issue.
2. Roaming, mobility and handover difficulties are solved.
3. There is total coverage in the cell i.e. there are no black spots.
4. There is a simplified inverse relationship between bit rate and delay (Figure 6.17).  
As the bit rate decreases, packets are more likely to be lost or errored, resulting in increased retransmissions increasing the overall mean delay.

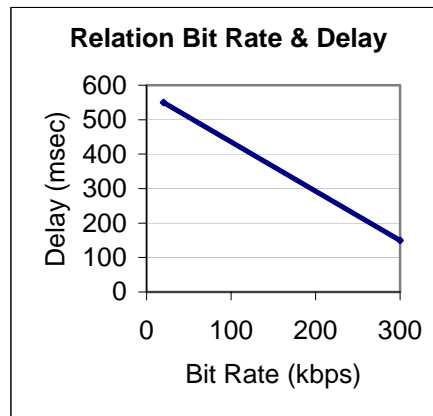


FIGURE 6.17: INVERSE RELATIONSHIP BETWEEN BIT RATE AND DELAY.

5. Maximum bit rate: The upper bound on the available bitrate is fixed at the maximum bit rate of the user in terms of their position in the cell and the network type.
6. Minimum bit rate: It is likely that the minimum bitrate could be 0bps when the network is heavily congested or when there is no coverage in an area of the cell. Equally, a user could be allocated 10kbps for one 20ms period in 100 such periods, so the aggregate bit rate is 100bps. To simplify, the testing process the lowest bit rate is defined as the lowest bit rate at which video can be streamed with a reasonable quality. A user with a bit rate lower than the lowest rate at which video can be streamed, cannot be serviced by the video server.
7. There is equal allocation of resources between users. If there is a maximum bit rate of maxBR kbps, and there are N users, each user is equally allocated maxBR/N kbps. Users are affected by their location and the increase in congestion.
8. The diameter of the cell is 1 km, which is typical in urban areas.

**6.5.2. Wireless Network Test Scenarios**

Modelling wireless networks is extremely complicated, as it must encompass user activity levels; user mobility; user velocity; cell dimensions, service and application requirements.

To date there are no typical models of wireless networks. A number of different scenarios were chosen to demonstrate the robustness and behaviour of PQA algorithm in wireless networks.

- User Location: Each network test scenario shows how a users' location affects their bit rate and delay, which will in turn affect the playout quality levels of a streaming application.
- Worst-case scenario: This test checks the robustness of the algorithm and system to provide reasonable service under seriously challenging conditions. In this test, the bit rate and delay oscillate between the maximum and minimum values every 5 seconds
- Multiple static users: When there are a number of active users within a cell, the available resources are divided between the users. Despite the connectivity and proximity of a user to the BS, the user will be affected by the level of congestion in the cell caused by other users. Two cases were chosen, a build-up of congestion and a decay of congestion.
  - Gradual congestion build-up: As congestion in the cell builds up, the resources available to each user are reduced in response to the increased congestion resulting in a reduced bit rate and increased delay.
  - Gradual congestion decay: As the congestion levels in the cell decay, users will gain more of the available resources and so their bit rate will increase and the delays will decrease.
- Single mobile user: This test demonstrates the effect of a users location and mobility through cells.

### *User Location*

The user can be at any location within a cell (Figure 6.18). The distance a user is from the base station (BS) affects the effective bandwidth of the user. Sample locations are chosen at positions Pa, Pb, Pc and Pd. The bit rate for each location is the maximum bit rate that this user will receive when there is no congestion in the cell.

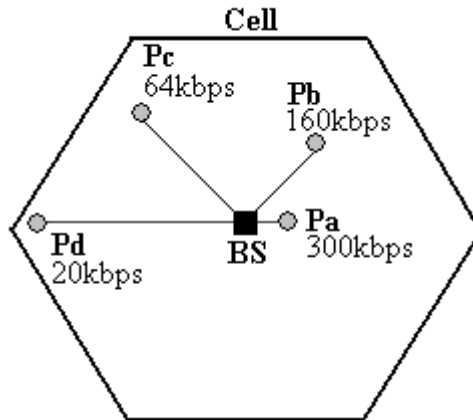


FIGURE 6.18: WIRELESS CELL

<i>Location</i>	<b>Pa</b>	<b>Pb</b>	<b>Pc</b>	<b>Pd</b>	<i>Minimum Service</i>
<b>Bit Rate (kbps)</b>	300	160	64	20	10
<b>Delay (msec)</b>	150	350	487	550	565

TABLE 6.2: WIRELESS CHANNEL CHARACTERISTICS AT DIFFERENT LOCATIONS

*Worst Case Scenario*

The worst possible case is when the bit rate and delays incurred oscillate between the minimum and maximum values (Figure 6.19).

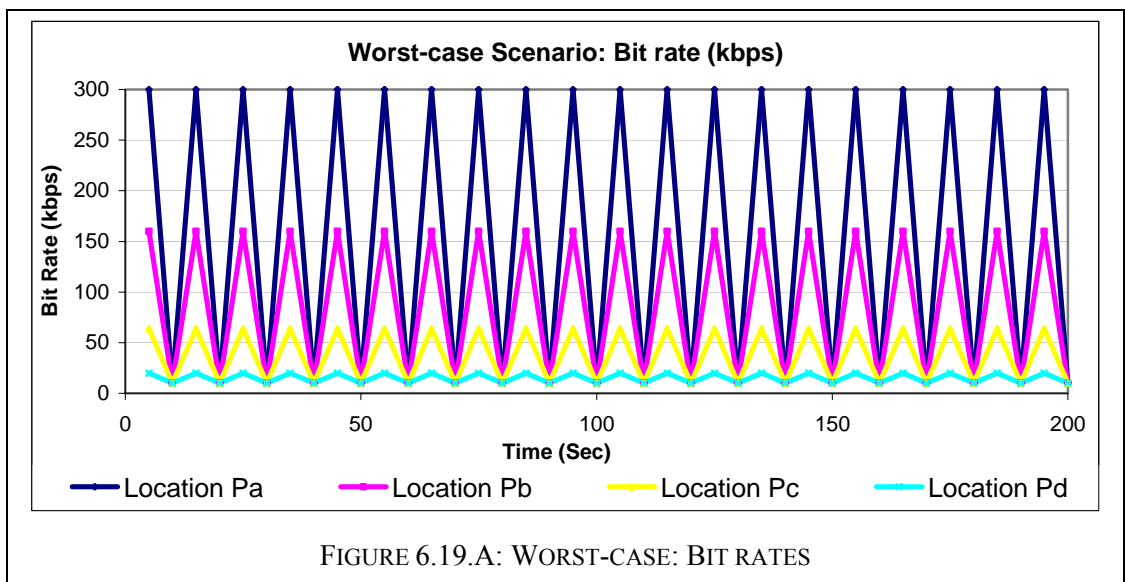


FIGURE 6.19.A: WORST-CASE: BIT RATES

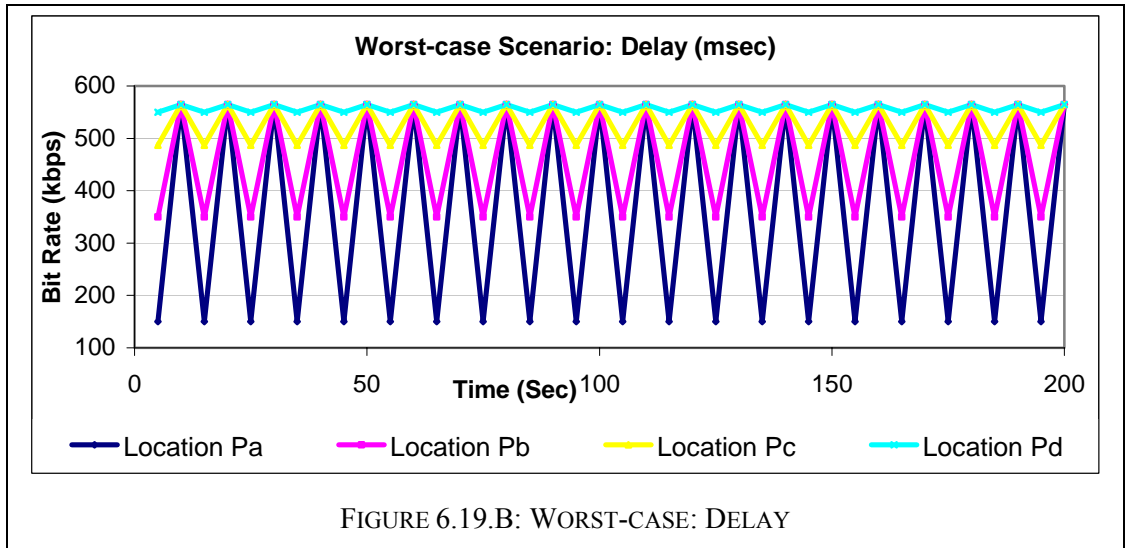


FIGURE 6.19.B: WORST-CASE: DELAY  
 FIGURE 6.19: WORST-CASE NETWORK CONDITIONS

*Multiple Static Users*

The purpose of this scenario is to show how the build up and decay of congestion can affect the users bandwidth.

*Gradual Congestion: Build up*

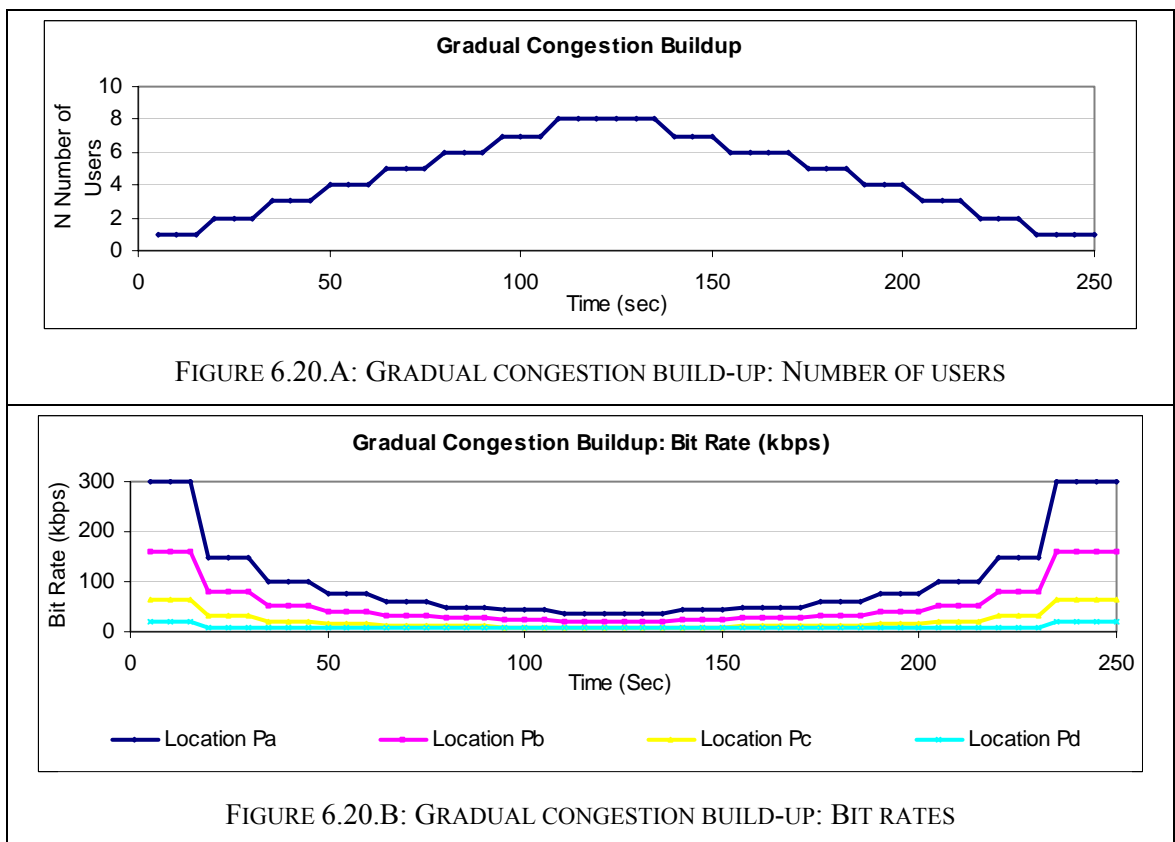


FIGURE 6.20.A: GRADUAL CONGESTION BUILD-UP: NUMBER OF USERS

FIGURE 6.20.B: GRADUAL CONGESTION BUILD-UP: BIT RATES

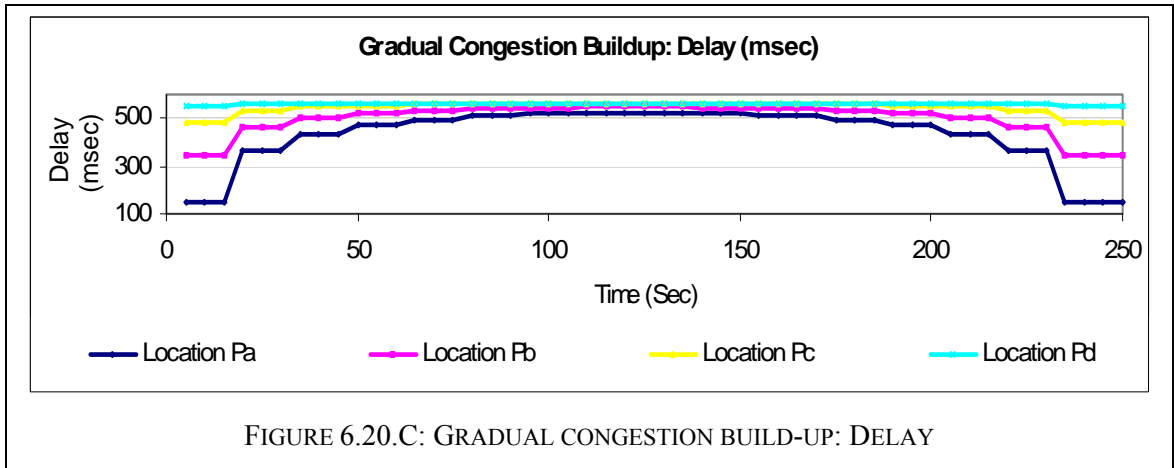


FIGURE 6.20: NETWORK CONDITIONS WITH GRADUAL CONGESTION BUILDUP

*Gradual Congestion: Decay*

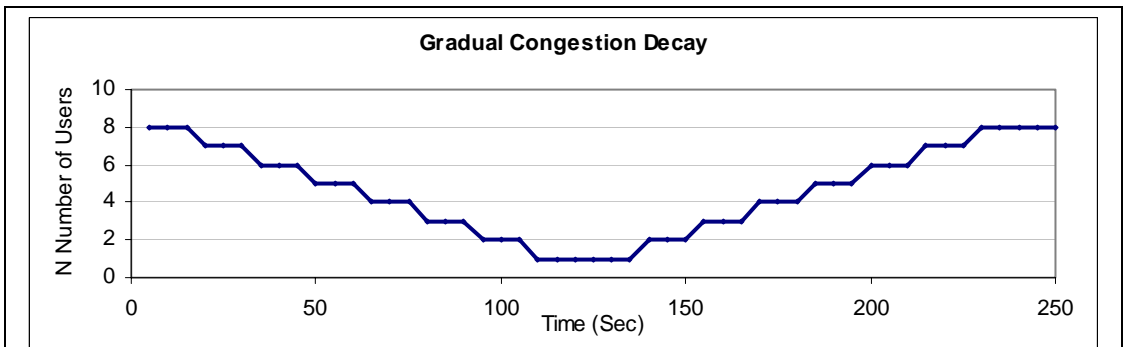
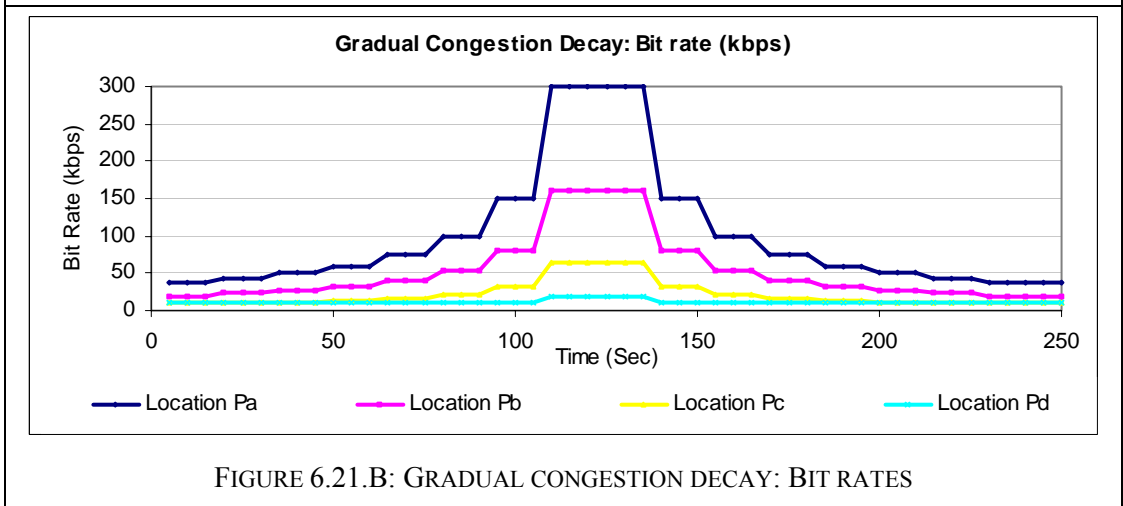


FIGURE 6.21.A: GRADUAL CONGESTION DECAY: NUMBER OF USERS



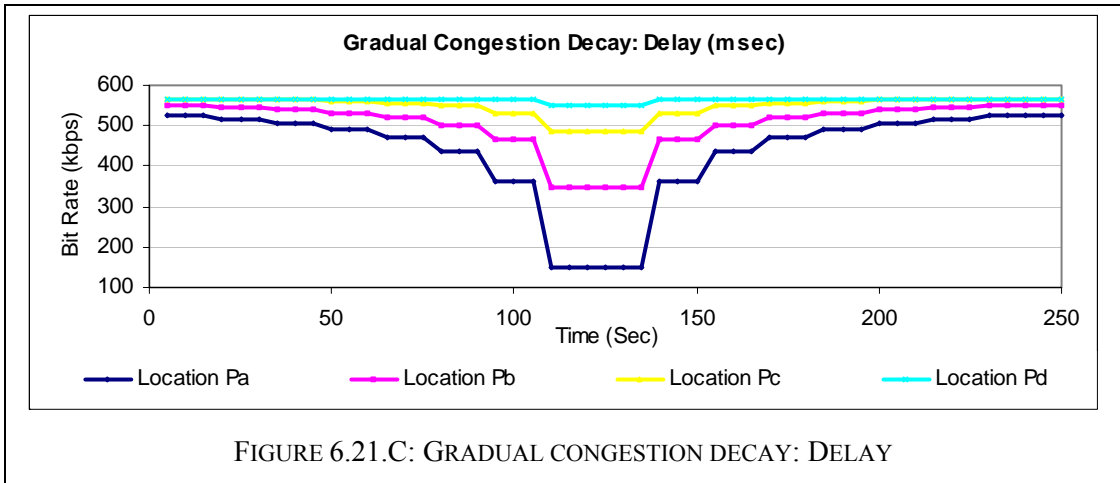


FIGURE 6.21.C: GRADUAL CONGESTION DECAY: DELAY

FIGURE 6.21: NETWORK CONDITIONS WITH GRADUAL CONGESTION DECAY

*Mobile Users*

A typical mobile user will be travelling at a high velocity and as such their distance relative to the BS will vary greatly. The rate at which the bandwidth changes depends on the size of the cell and the velocity at which the user is travelling. This test considers a user travelling a 10m/s and in a cell of 1km diameter (Figure 6.22.A). The bit rate will vary from 10kbps to 300kbps over 50sec, the time it takes the user to travel from the edge of the cell to the centre of the cell (Figure 6.22.B). During the handover from one cell to another, the bit rate is a minimum value for a period of the handover duration, 10sec.

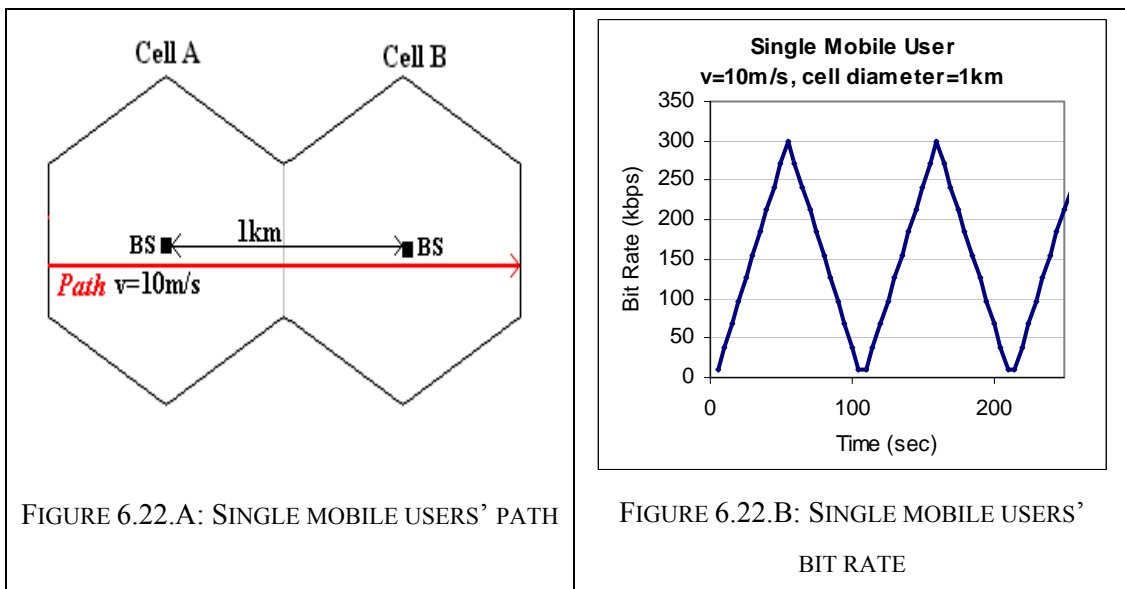


FIGURE 6.22.A: SINGLE MOBILE USERS' PATH

FIGURE 6.22.B: SINGLE MOBILE USERS' BIT RATE

FIGURE 6.22: NETWORK CONDITIONS OF A SINGLE MOBILE USER

## 6.6. Simulation Environments

### 6.6.1. Simulation Setup

The simulation environment is the same as that used for the simulations of PQA in Chapter 6, Section 4.6. The simulation was built into the server application and so there is greater control and reproducibility of the simulated network tests. The server compares its transmission bit rate with the bandwidth at a particular instant. If the transmission bit rate is greater than the simulation bandwidth, the server drops a percentage of packets in groups to simulate a drop-tail queue that will typically occur at the bottleneck router. The packets that are not dropped are passed to a separate thread, which will delay the actual transmission of the packets to simulate network delay. The RTCP feedback frequency is sent at 5-second intervals. Both RTCP-RR and RTCP-APP feedback were sent at the same time but only the RTCP-RR feedback was used in the PQA algorithm to infer the playable frame rate. As soon as the client connects to the server, the network simulation is started. The content is a constant bit rate stream with an MTU of 512 bytes and zero loss tolerance.

### 6.6.2. Content Preparation for Simulation Tests

In the wireless emulations, the content being streamed was encoded according to the ISMA Mobile MPEG-4 Profile [193][194]. Using the globally averaged OAT, there are 6 video tracks with a frame rate of 25fps and with decreasing spatial resolution as determined by the quantised OAT (Figure 6.23).

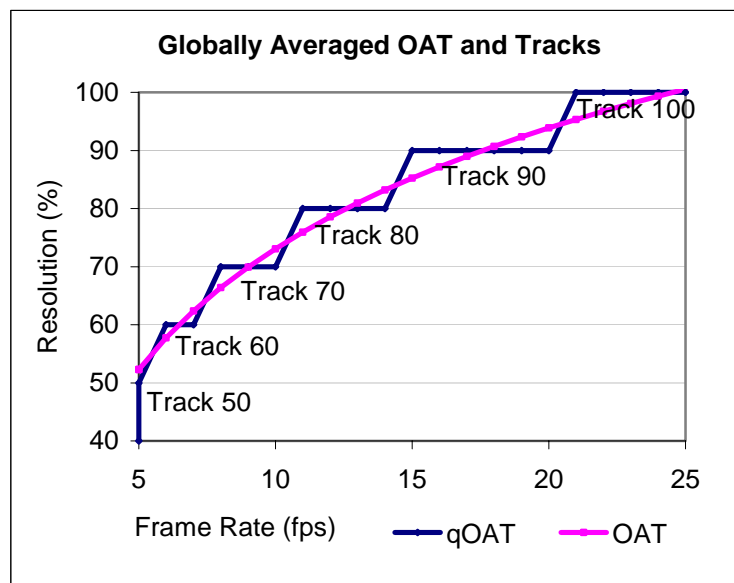


FIGURE 6.23: TEST CONTENT TRACK STRUCTURE



Using the MP4creator tool from the MPEG4IP developer group [195], the encoding configuration of the various tracks within the test file is listed. Each video track is encoded at a frame rate of 25fps using the MPEG-4 Simple Profile at Level 1. Each track has a duration of 298.88 seconds and has a QCIF (176 x 144) display size. In the test scenarios investigated, each hint track is the exactly the same.

<b>Encoding configuration:</b>	
Mobile MPEG-4	
Key Frame every 24 frames with resynchronisation markers.	
<b>Track Type</b>	
Video	MPEG-4 Simple @ L1, 298.880 secs, 63 kbps, 176x144 @ 25.00 fps
Video	MPEG-4 Simple @ L1, 298.880 secs, 11 kbps, 176x144 @ 25.00 fps
Video	MPEG-4 Simple @ L1, 298.880 secs, 23 kbps, 176x144 @ 25.00 fps
Video	MPEG-4 Simple @ L1, 298.880 secs, 31 kbps, 176x144 @ 25.00 fps
Video	MPEG-4 Simple @ L1, 298.880 secs, 41 kbps, 176x144 @ 25.00 fps
Video	MPEG-4 Simple @ L1, 298.880 secs, 52 kbps, 176x144 @ 25.00 fps
OD	Object Descriptor
Scene	BIFS

TABLE 6.3: ENCODING CONFIGURATION FOR TEST SEQUENCE

Figure 6.24.A shows the bit rate plane for the various encoding parameters whilst Figure 6.24.B shows how the bit rate varies through track switching and decreasing frame rate using the OAT.

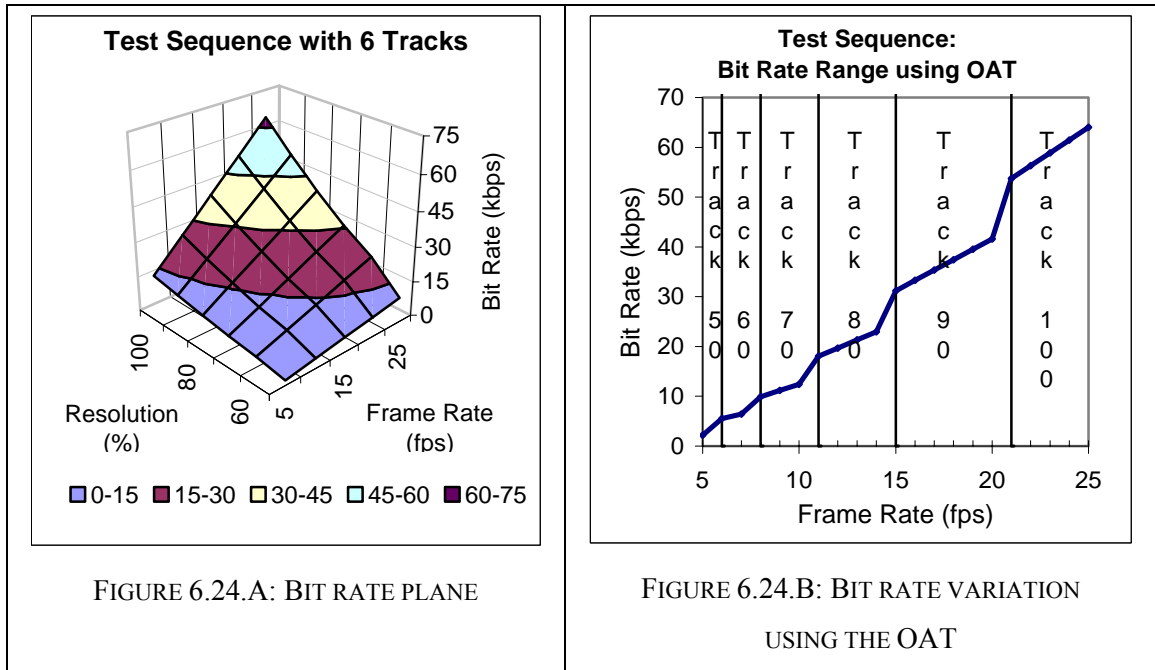


FIGURE 6.24: BIT RATE FOR TEST SEQUENCE

The test sequence has an average bit rate of 64kbps for the track encoded at 25fps and 100% spatial resolution (Figure 6.25). During the streaming process the test sequence has a transmission variation ranging from 55kbps to 80kbps, caused by scene changes and increased content activity. This bit rate variation is mirrored in each of the 6 tracks in the test sequence.

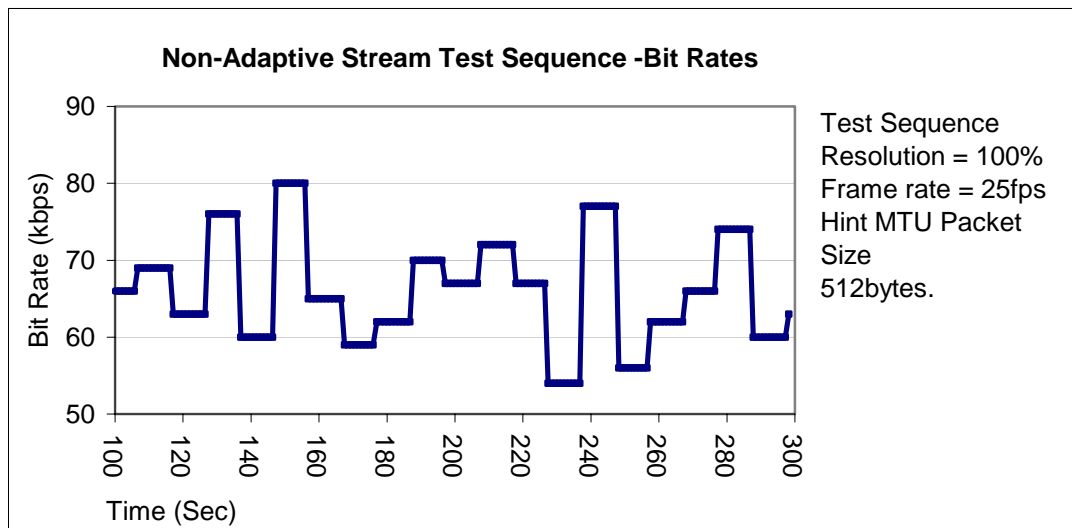


FIGURE 6.25: BIT RATE FLUCTUATIONS OF NON-ADAPTIVE TEST SEQUENCE

## 6.7. PQA in Wireless Network Test Results

The server received both RTCP-RR and RTCP-APP feedback. The server adapted every second using the PQA algorithm using the most recent network values from the RTCP-APP feedback. Presented here are the results for the most demanding location scenario, Pa. This location is very challenging because the available bit rate can vary between 300 and 10kbps, which is a dramatic reduction. Each of the different location scenarios (i.e. Pb, Pc and Pd) were tested and their results are presented in Volume 2 Appendix H.

### 6.7.1. Congestion Build-up Simulation Results

It can be seen that there is a two loss bursts, one after 60 seconds and another at 90 seconds. The first causes the system to adapt down slightly to users playout capacity. In the interval from 60-90 seconds, there is no loss and the clients' playout capability is not affected so the system recovers every 5 seconds slowly increasing the quality. However, just as the system recovers to the maximum quality, the bit rate exceeds the available bandwidth causing the second loss burst. This occurs as the available bandwidth is approaching its lowest value causing the large loss burst of 20%. This loss burst causes the system to transmit at the lowest quality (5fps and 50% resolution). The degradation in quality results in the transmitted bit rate being far below the available bandwidth of the network yet is the playout capacity of the client. Once this dramatic adaptation has taken affect, the system recovers to the maximum quality. The network delay has no effect on the play out of the multimedia file as there is a 3 second pre-buffering delay which ensures that packets that have not been dropped arrive at the player in time for playout.

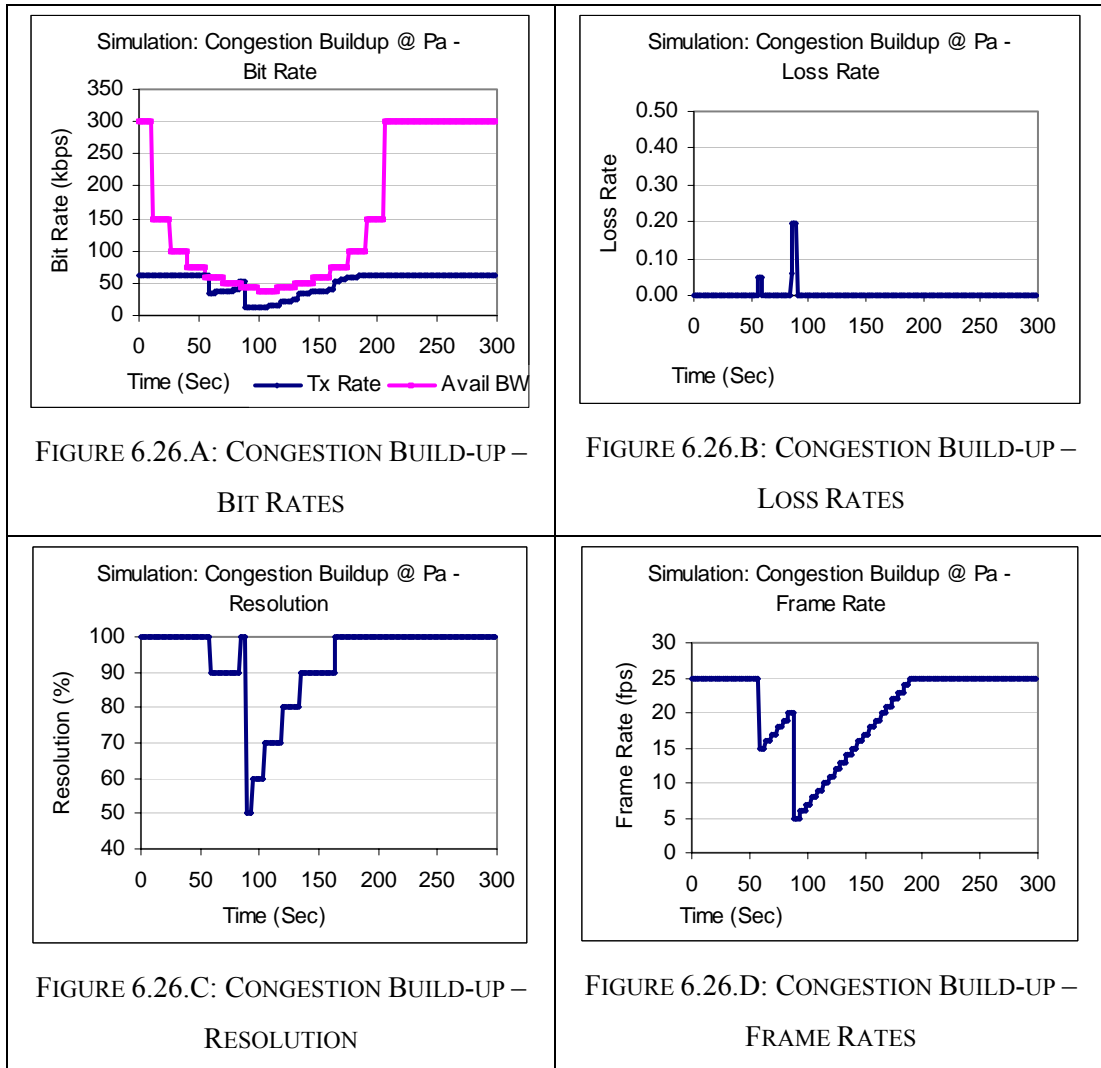


FIGURE 6.26: CONGESTION BUILDUP SIMULATION RESULTS

### 6.7.2. Congestion Decay Simulation Results

It can be seen that shortly after the system begins, there is a large loss bursts. This is due to the fact that the server automatically begins streaming at the maximum quality without any knowledge of the network congestion levels. This large loss burst causes the system to reduce its quality to 50% resolution and 5fps. The system recovers quickly to the maximum transmission rate coinciding with the peak in the available network bandwidth. As the available network bandwidth decreases due to increased congestion, there is a small loss burst followed by a larger loss burst. The system adapts to transmit the lowest quality at 50% resolution and 5fps. This adaptation causes the transmitted bit rate to be below the available network bandwidth. However, upon each feedback report indicating no loss, the server attempts to increase the transmitted quality causing periodic loss bursts. These loss bursts indicate that the clients' playout capability is 90% resolution and 15-17fps.

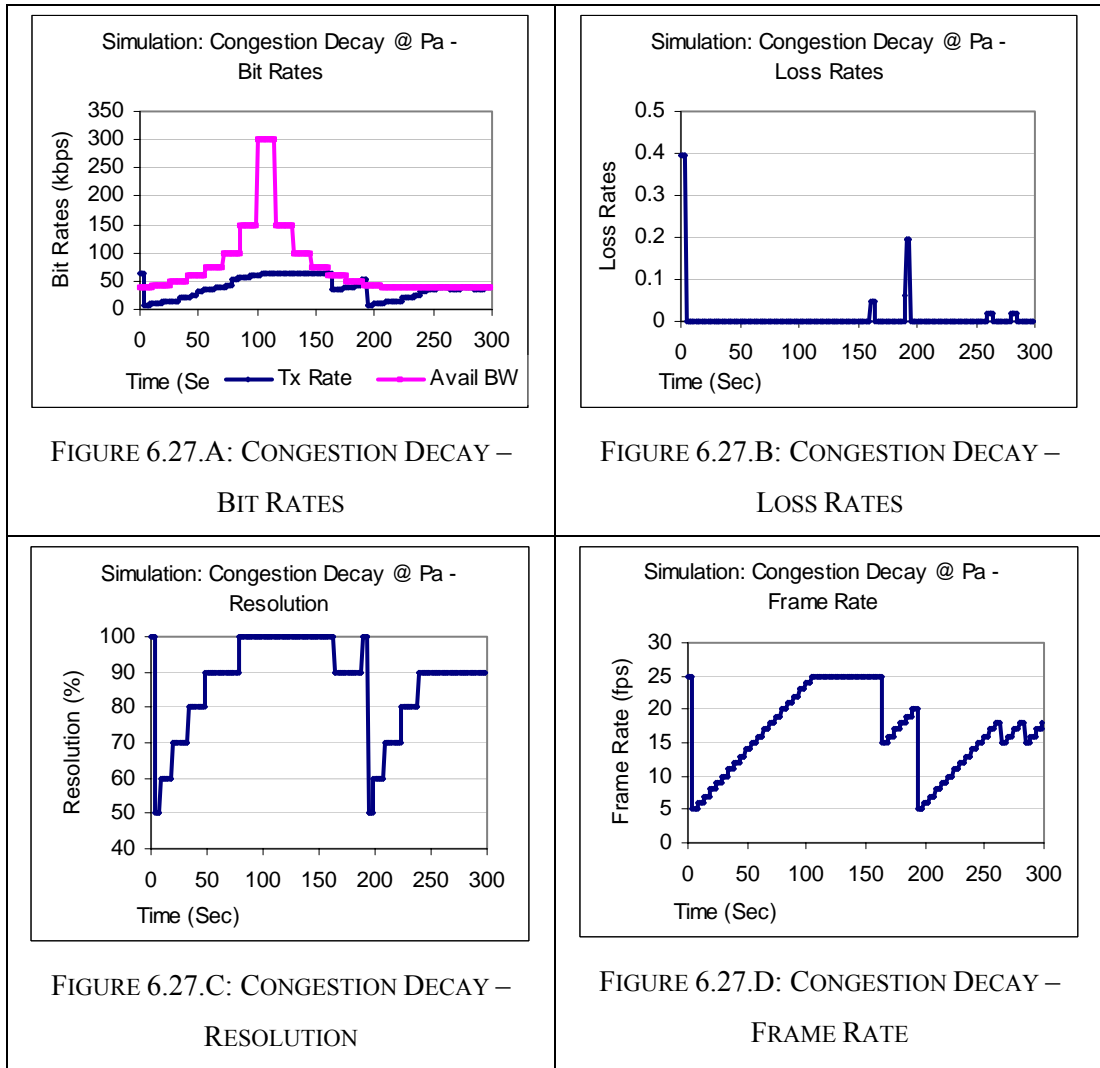


FIGURE 6.27: CONGESTION DECAY SIMULATION RESULTS

### 6.7.3. Worst-case Scenario Simulation Results

Once again there is a large loss burst after the system begins, there is a large loss bursts due to the fact that the server automatically begins streaming at the maximum quality. This large loss burst causes the system to reduce its quality to 50% resolution and 5fps. The system attempts to increase the quality being transmitted until it encounters another loss feedback message. This AIMD behaviour results in periodic adaptation and loss bursts.

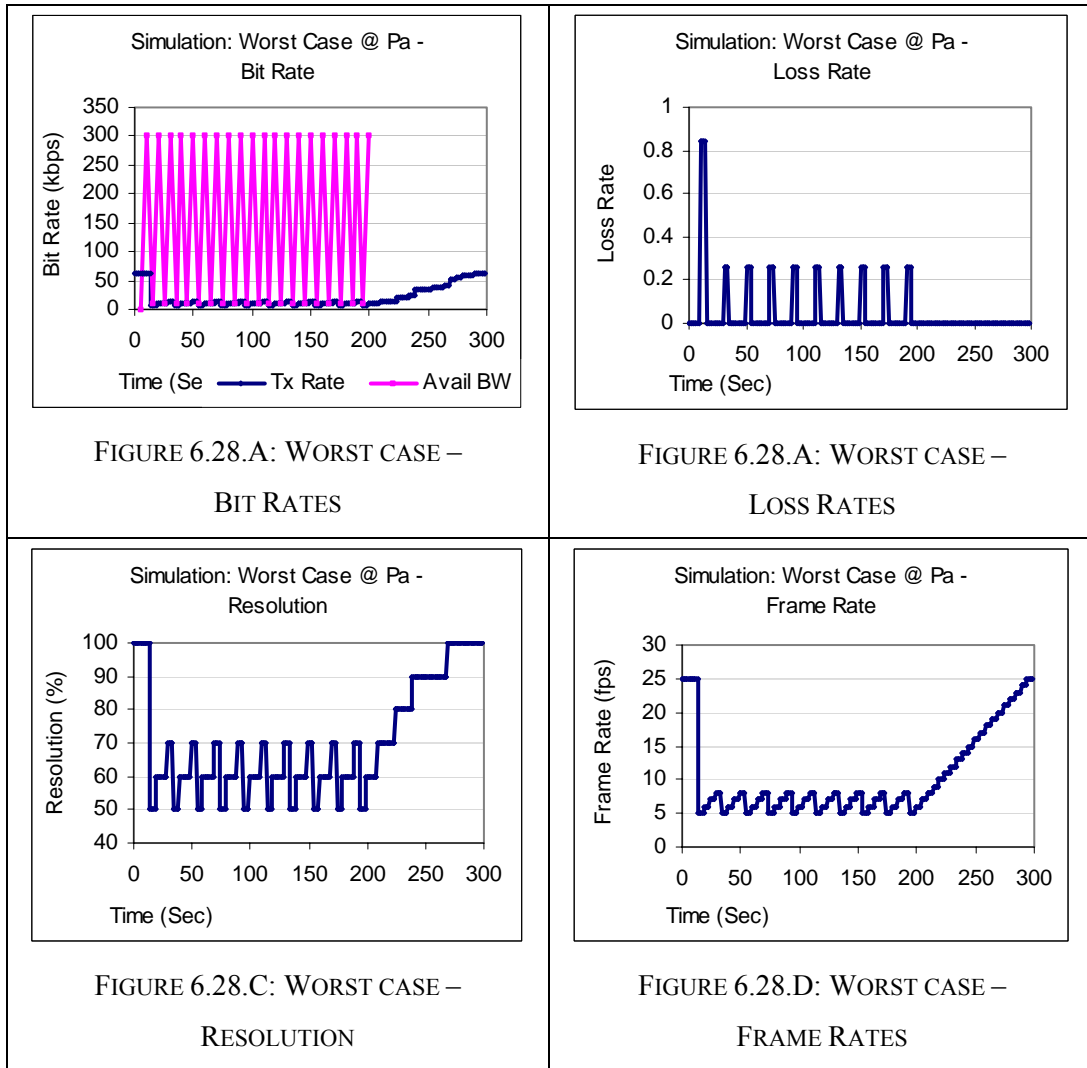


FIGURE 6.28: WORST-CASE SCENARIO SIMULATION RESULTS

### 6.7.4. Mobile Path Simulation Results

In this simulation, the quality is decreased and increased periodically whenever the transmitted bit rate exceeds the available bandwidth. The loss bursts are quite high which causes the server to adapt down to the lowest quality at 50% resolution and 5fps.

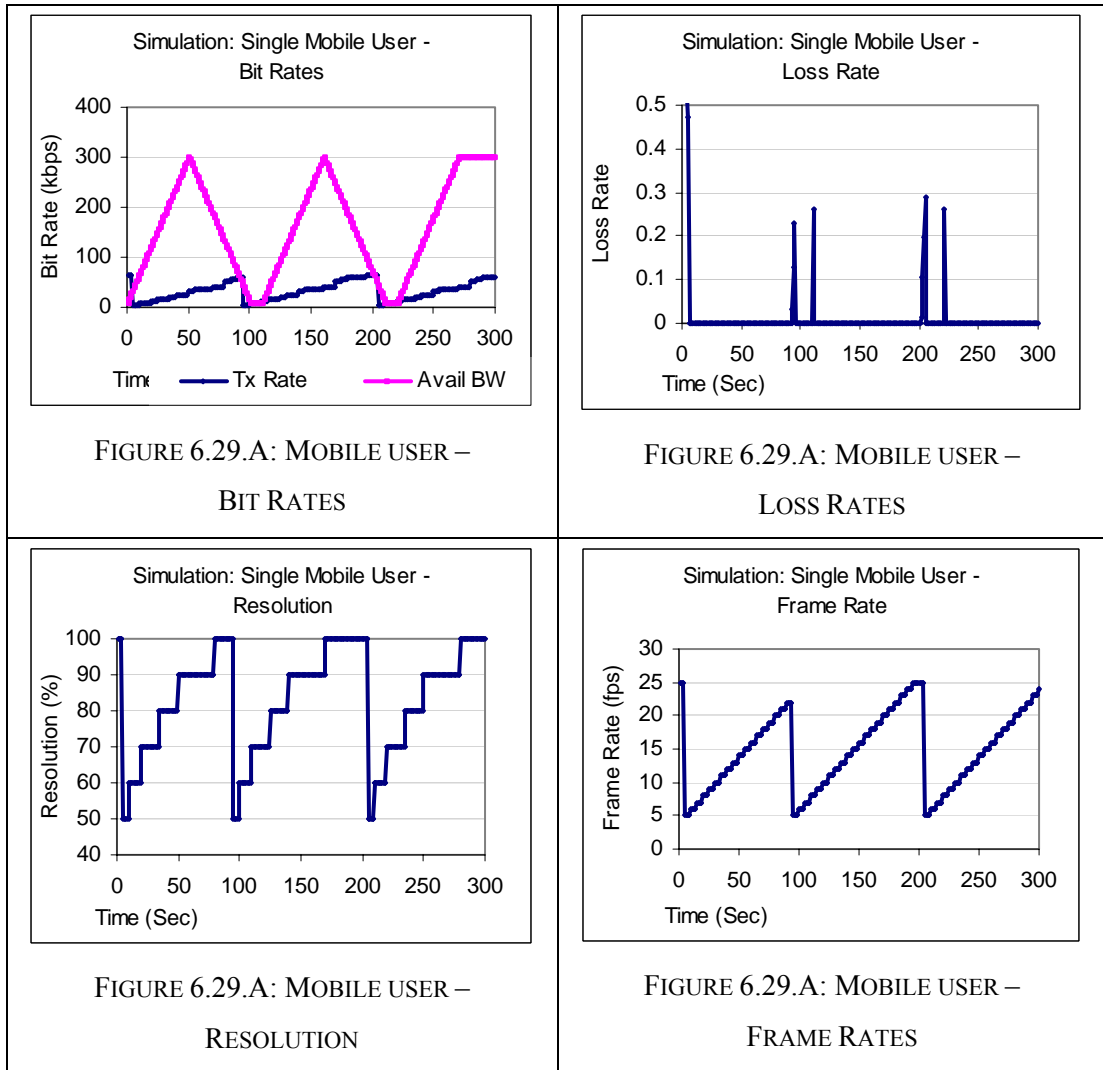


FIGURE 6.29: MOBILE USER SIMULATION RESULTS

Overall, the PQA algorithm performs in difficult network conditions. As discussed in Chapter 6, Section 4.7, there are a number of ways the PQA algorithm can be improved. In addition to this, wireless-specific feedback can improve the performance of the algorithm. For a wireless network, there are other factors other than application related factors, which may affect the playout of the stream. For example, the device may be able to signal to the server, its battery consumption levels. The battery on a mobile device will be drained very quickly for streaming applications and this affects its ability to playout the stream. Further, when the device is handing off from one cell to another, there may be methods in which the server could be aware of this and use a combination of fast-streaming and pre-emptive adaptation to mask the effects of handoff such as reduced bit rates and increased delays.

## 6.8. Summary and Conclusions

This chapter demonstrates how OAT's can be used in an adaptive streaming system. The system targets unicast streaming of pre-encoded content. The system developed is fully compliant with IETF transport protocols, ISMA streaming profiles and MPEG-4 standard profiles. The system achieves two dimensions of adaptation using a combination of frame dropping and track switching to switch between tracks encoded with different spatial resolutions. These dimensions of adaptation correspond to the adaptation dimensions of the OAT. Multi-tracked content can be applied to MPEG-4 files and the emerging wireless multimedia standard, 3GP file format.

The OAT can be used with any sender-based algorithm. To demonstrate this claim, the OATs have been shown to complement a widely known sender based adaptation algorithm, Loss Delay Adjustment (LDA) algorithm. The server receives RTCP-RR feedback from the client and LDA uses the reported loss rate to indicate how the bit rate should be adjusted to the network conditions. The server finds intersection of this new transmission bit rate with the OAT, to indicate what encoding configuration should be sent to achieve this new bit rate. The OAT could also be used to complement layered adaptation schemes, encoder and transcoder-based schemes.

A new type of adaptation algorithm, Perceptual-Quality Adaptation (PQA) algorithm, has been developed. This PQA algorithm integrates well to the framework of the developing 3GP standard for streaming multimedia to wireless devices. This algorithm uses the OAT directly to make adaptation decisions. Based on feedback from the client, the server determines the clients' playout ability. The server has two operating points on the OAT, the clients' playout ability and the transmitted encoding configuration. The server adjusts the transmitted encoding configuration to match the clients' playout ability. There two types of client considered, clients who send standard RTCP-RR and those who send both RTCP-RR and RTCP-APP feedback. RTCP-APP feedback has been designed to return useful application specific information to the server, so that the server can determine directly the clients' playout ability. If the client returns only standard RTCP-RR feedback, then the server uses probabilistic inference mechanism to determine the clients' playout ability. The PQA algorithm behaviour was demonstrated in typical wireless best-effort IP simulation environment where it performed well under harsh operating conditions.



## Chapter 7

### Conclusions and Future Work

The main goal of this research was to gain a better understanding of and insight into user perception in the context of adaptive multimedia streaming. Current adaptation policies address the problem of how to adapt only in terms of adjusting the transmission rate, window size or encoding parameters, they do not consider the impact of perception nor do they indicate how this bit rate adaptation can be achieved in terms of actual video encoding parameters. Several objective metrics of video quality have been proposed, but they are limited and are not satisfactory in quantifying human perception. A key issue addressed in this research is how to adapt a video transmission in order to maximize the resulting user-perceived quality.

This research makes the following novel contributions:

- 1. Proposes the concept of an OAT, which exists in an adaptation space defined by spatial resolution and frame rate.***

The OAT has been discovered within the dimensions of frame rate and spatial resolution. These were chosen as they map to a contents spatial and temporal characteristics. Indeed, the OAT has many dimensions and can be discovered using the same subjective test methodology described.

- 2. Presents a subjective methodology that can be used for discovering the OAT of any content type contained.***

The OAT is a new concept and as such there was no pre-defined method for its discovery. The OAT exists in an adaptation space that may have many dimensions and additionally, each content type has its own unique OAT in adaptation space. To limit the ambiguity and interpretation of quality and quality assessment, the forced choice method was used. Over the course of this research over 300 subjective tests were performed.

Two valid paths of adaptation were discovered, the path of maximum user preference and the weighted path of preference for various different content types. Further subjective testing suggested that these paths are perceptually equivalent implying that either the path of maximum user preference or the interpolated weighted path of preference could be used as the OAT.

**3. *Suggests that the OAT cannot be discovered using objective metrics.***

The possibility of discovering the OAT using objective metrics was investigated. Two metrics were selected, the PSNR and VQM, to try and determine whether they yielded an OAT. The paths of maximum quality measured by these metrics did not correlate in any way to the OAT discovered by subjective testing for the same content type. Despite the sophistication of quality metrics developed to date, they have been designed for assessing quality impairments of a fixed quality video and not for assessing the quality of an adaptive video transmission. They are, therefore, not appropriate for discovering the OAT.

**4. *Suggests that 2-dimensional adaptation using the OAT out-performs one-dimensional adaptation of spatial resolution and frame rate.***

The usefulness of the OAT in a real video adaptation scenario was investigated using subjective methods. In all cases tested, adaptation using the OAT out-performed one-dimensional adaptation of frame rate or spatial resolution currently employed for adaptive delivery of pre-encoded multimedia content.

**5. *Demonstrates how the OAT can be used to complement any sender-based adaptation algorithm.***

The OAT can be used with any sender-based algorithm. To illustrate this, the OATs have been shown to complement a widely known sender based adaptation algorithm, Loss Delay Adjustment (LDA). LDA uses the reported loss rate to indicate how the bit rate should be adjusted. A system was developed that operated by finding the intersection of this new transmission bit rate with the OAT, to indicate what encoding configuration should be used to achieve this new bit rate.

**6. *Proposes and develops a new algorithm Perceptual Quality Adaptation (PQA) algorithm that directly uses the OAT as a means of making adaptation decisions.***

Network feedback can give a warped view of an applications' performance. For example, there may be no loss in the network but the clients' application cannot support the incoming traffic and many packets are lost at the clients' buffer. A typical streaming server would interpret the zero network loss as an indicator that the transmission quality should be further increased and therefore aggravate the problem further. In the PQA system, regardless of the loss feedback, the server attempts to determine the clients' playout abilities, either directly or using inferences and match its transmission to the clients' capabilities. Wireless network test scenarios are explicitly considered to test the PQA algorithm. The results show that the system performs well under these conditions. They also highlight several issues that should be considered to improve the performance and behaviour of PQA.

## 7.1. Future Work

This research has only touched the surface of the many potential uses and applications of the OAT so some future work and developments are outlined.

### *n-Dimensional OAT*

The work presented here has shown the OAT for the adaptation of MPEG-4 video streams within a two-dimensional adaptation space defined by frame rate and spatial resolution.

Adaptation space is multi-dimensional. Other dimensions include:

- Quantization parameter
- Audio

In this research, the audio context has been ignored by assuming that the audio should remain at a fixed quality. However, there is a relationship between audio and video and the quality of the audio stream can affect the perception of the video stream.

### *Content-based Adaptation using a Dynamic OAT*

The usefulness of the OAT relies on the contents' spatial and temporal characteristics being known by the adaptive streaming server. However, the contents' characteristics will vary over time. These spatial and temporal complexity metrics are highly computationally intense and not feasible in real-time. One solution is to use a globally averaged OAT (Chapter 4, Section 8). Another solution is to apply a dynamic content specific OAT. A dynamic content specific OAT predicts the OAT in real-time in response to variations in the contents' characteristics.

### *Enhancing the PQA Algorithm*

The PQA algorithm can be improved and extended in many ways. The responsiveness of the algorithm can be improved to allow the system to more quickly to changes in the network conditions. The system reacts on immediate feedback, however it may be more efficient to monitor long-term and short-term network conditions and make more accurate decisions.

- **Algorithm Responsiveness:** The responsiveness of the algorithm is dependent on the feedback frequency. One possibility to improve the responsiveness of the algorithm is to increase the feedback frequency.
- **TCP-Friendliness:** The PQA algorithm behaves in an AIMD manner. There should be fair sharing of bandwidth with competing connections, particularly TCP connections. This has not been investigated in the development of the PQA algorithm.

- **Feedback monitoring:** Monitoring the long term and short network connection characteristics may prevent the system from over-reacting to bursty losses. To prevent unnecessary adaptation that may result in a sudden and abrupt adaptation particularly when multiplicatively decreasing the quality, filtered responses using short and long-term feedback should improve the systems adaptation strategy.
- **Wireless-specific Feedback:** For a wireless network, there are other factors other than application related factors, which may affect the playout of the stream. In addition to application-specific feedback, device-specific feedback may be necessary. For example, the device may be able to signal to the server, its battery consumption levels. The battery on a mobile device will be drained very quickly for streaming applications and this affects its ability to playout the stream. Further, when the device is handing off from one cell to another, there may be methods in which the server could be aware of this and use a combination of fast-streaming and pre-emptive adaptation to mask the effects of handoff such as reduced bit rates and increased delays.

#### *Layered OATs for Multicast*

The OATs have been shown to work with any sender-based adaptation algorithm. They can also be applied to multicast environments. Each multicast group in a session streams a layer at a particular bit rate or quality. In layered adaptation algorithms, there is no definition of what constitutes a base layer and its enhancement layers. Further there is an inter-layer dependency; for example, the base layer needs to be correctly decoded before subsequent enhancement layers can be decoded. However, if the base layer is not correctly decoded, the enhancement layers cannot be decoded. Even if layers are correctly decoded, as they may have different paths and different round-trip times, the receivers need to resynchronise the arriving data, which can further complicate the playout of the stream.

This problem can easily be addressed by using the OATs. In the unicast scenario, each track in an MP4 file corresponded to a different quality version as defined by the OAT. Similarly, in a multicast scenario, each multicast group corresponds to a different track (e.g. Figure 7.1). Each track is independent of the other tracks, thus eliminating the inter-layer dependencies. Clients subscribe to a multicast group, which most closely match their allocated resources and can switch between groups seamlessly as their needs change. Within a multicast group there is the possibility for group adaptation by adapting the frame rate within a certain range of values defined by the layers' range. If the OAT changes for the content, this will be reflected in the new ranges of the defined layers.

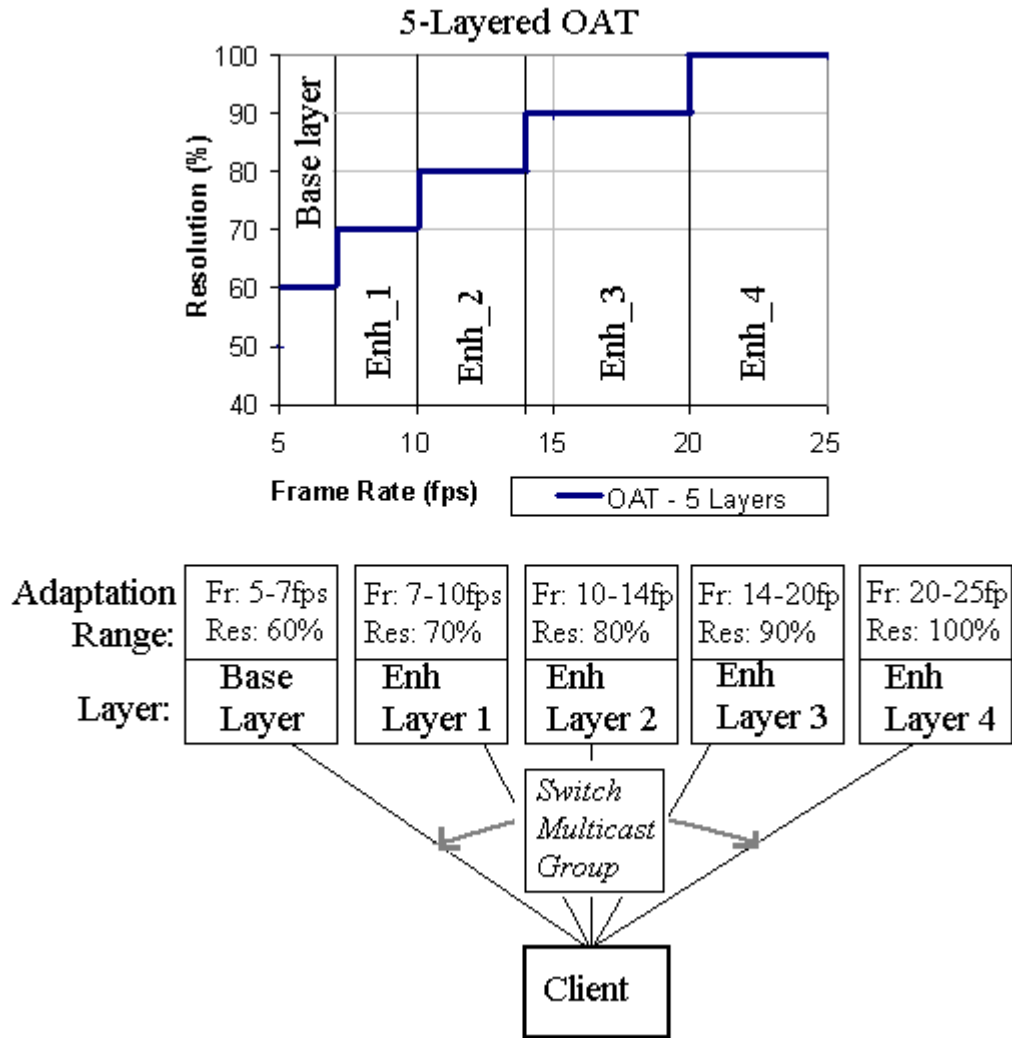


FIGURE 7.1: OATs FOR MULTICAST SESSIONS

## **Refereed Publications**

### **International Conference Papers**

N. Cranley, L. Murphy, P. Perry, “User-Perceived Quality Aware Adaptive Delivery of MPEG-4 Content”, NOSSDAV 2003, Monterey, California, June, 2003.

N. Cranley, L. Murphy, “Adaptive Quality of Service for Streamed MPEG-4 over the Internet”, ICC 2001, IEEE International Conference on Communications, Helsinki, Finland, June 2001.

N. Cranley, L. Murphy, “Adaptive Streaming of Multimedia over the Internet” (Experimental Results), UKTTS 2001, UK Teletraffic Symposium, Dublin, Ireland, May 2001.

### **National Conference Papers**

M. Searles, N. Cranley, L. Murphy, “Adaptive Multicast Architecture for Streamed MPEG-4 over IP Networks”, ISSC 2002, Irish Signals and Systems Conference, UCC, Cork, Ireland, June 2002.

N. Cranley, L. Murphy, L. Fiard, “Quality of Service for Streamed Multimedia over the Internet” ISSC 2000, Irish Signals and Systems Conference, Belfield, UCD, Dublin, Ireland, June 2000.

## References

- 
- [1] ITU Telecommunication Standardisation Sector, ITU-T, <http://www.itu.int/ITU-T/>
- [2] The MPEG Home Page, <http://www.chiariglione.org/mpeg/>
- [3] H. Schulzrinne, Tutorial “Wireless IP Multimedia”, MOBICOM Multimedia Tutorial, September, Atlanta, Georgia, <http://www.cs.columbia.edu/~hgs/papers/2002/mobicom.ppt>
- [4] A. S. Tanenbaum, “Computer Networks”, 4<sup>th</sup> Edition, Prentice-Hall, ISBN: 0130661023
- [5] Vcodex, “H.264/MPEG-4 Tutorial”, <http://www.vcodex.fsnet.co.uk/h264.html>
- [6] R. Koenen, “MPEG-4 Overview” Version 21, ISO-IEC JTC1/SC29/WG 11: N4668 Coding of Moving Pictures and Audio, MPEG, [www.m4if.org/resources/Overview.pdf](http://www.m4if.org/resources/Overview.pdf)
- [7] O. Avaro, A. Eleftheriadis, C. Herpel, G. Rajan, L. Ward, “MPEG-4 Systems: Overview”, [http://vilab.hit.edu.cn/~bozhang/MPEG/leonardo.telecomitalia.com/icjfiles/mpeg-4\\_si/3-systems\\_overview\\_paper/3-systems\\_overview\\_paper.htm](http://vilab.hit.edu.cn/~bozhang/MPEG/leonardo.telecomitalia.com/icjfiles/mpeg-4_si/3-systems_overview_paper/3-systems_overview_paper.htm)
- [8] ISO/IEC 14496-1. “Information Technology – Coding of Audio-visual Objects, Part 1: Systems”, January 1998, ISO/IEC JT1/SC 29/WG 11 International Standard.
- [9] F. Pereira, T. Ebrahimi, “The MPEG-4 Book”, Prentice-Hall, ISBN: 0-13-061621-4
- [10] S.N. Fabri, S.T. Worrall, A. H. Sadka, A.M. Kondoz, “Real-time Video Communications over GPRS”, IEE 3G2000, London, March 2000, pp. 426-430
- [11] Request for Comments: 1889, “RTP: A Transport Protocol for Real-Time Applications”, <http://www.ietf.org/rfc/rfc1889.txt?number=1889>
- [12] RTP Working Group, <http://www.cs.columbia.edu/~hgs/rtp/>
- [13] Request for Comments: 3016, “RTP Payload Format for MPEG-4 Audio/Visual Streams”, <http://www.ietf.org/rfc/rfc3016.txt?number=3016>
- [14] Request for Comments: 2326, “RTSP: Real-Time Streaming Protocol”, <http://www.ietf.org/rfc/rfc2326.txt?number=2326>
- [15] Request for Comments: 2327: "SDP: Session Description Protocol", <http://www.ietf.org/rfc/rfc2327.txt?number=2327>
- [16] M. Li, M. Claypool, R. Kinicki, J. Nichols, “Characteristics of Streaming Media Stored on the Internet”, WPI-CS-TR-03-18, Computer Science Technical Report Series, Worcester Polytechnical Institute, Massachusetts, May 2003
- [17] J. Liu, "Signal Processing for Internet Video Streaming: A Review", Proceedings of SPIE Image and Video Communications and Processing, January 2000.
- [18] B. Braden, D. Black, J. Crowcroft, B. Davie, S. Deering, D. Estrin, S. Floyd, V. Jacobson, G. Minshall, C. Partridge, L. Peterson, K. Ramakrishnan, S. Shenker, J. Wroclawski, and L. Zhang, “Recommendations on queue management and congestion avoidance in the Internet,” RFC 2309, Internet Engineering Task Force, April 1998.
- [19] ITU-T, Study-Group 12, COM 12-D98-E, “Analysis, measurement and modeling of Jitter”, Internal ITU-T Report

- 
- [20] S. Moon, J. Kurose, P. Skelly, D. Towsley. "Correlation of packet delay and loss in the Internet". Technical report, University of Massachusetts, January 1998.
- [21] H. Sanneck, G. Carle, R. Koodli. "A framework model for packet loss metrics based on loss run-lengths". In Proceedings of SPIE/ACM SIGMM Multimedia Computing and Networking Conference, January 2000.
- [22] M. Yajnik, S. Moon, J. Kurose, D. Towsley, "Measurement and modelling of the temporal dependence in packet loss". In Proceedings of IEEE Infocom, New York, March 1999.
- [23] J-C. Bolot, S. Fosse-Parisis, D. Towsley, "Adaptive FEC-Based error control for interactive audio in the internet". In Proceedings of IEEE Infocom, New York, March 1999.
- [24] W. Jiang, H. Schulzrinne, "Modelling of packet loss and delay and their effect on real-time multimedia service quality", In Proceedings of NOSSDAV 2000.
- [25] B. Sklar, "Rayleigh fading channels in mobile digital communication systems Part I: characterization," IEEE Communications Magazine, Vol. 35, No. 7, pp. 90-100, July 1997.
- [26] J. Villasenor, Y.-Q. Zhang, J. Wen, "Robust video coding algorithms and systems," in Proceedings of the IEEE, Vol. 87, No. 10, pp. 1724-1733, Oct. 1999.
- [27] M. Li, M. Claypool, R. Kinicki, "MediaPlayer versus RealPlayer - A Comparison of Network Turbulence", IMW2002 (Internet Measurement Workshop) Marseille, France, November 2002
- [28] Apple Quicktime home page, <http://www.apple.com/mpeg4/>
- [29] A. Gupta, "Rate based flow and congestion control schemes", Technical Report, Dept. of Computer Science, University of Arizona  
<http://www.cs.arizona.edu/people/amit/rate.pdf>
- [30] X. Wang, H. Schulzrinne, "Comparison of Adaptive Internet Applications", In Proceedings of IEICE Transactions on Communications, Vol. E82-B, No. 6, pp.806-818, June 1999.
- [31] D. Wu, Y. T. Hou, W. Zhu, H.-J. Lee, T. Chiang, Y.-Q. Zhang, H. J. Chao, "On end-to-end architecture for transporting MPEG-4 video over the Internet," IEEE Transactions on Circuits and Systems for Video Technology, vol. 10, no. 6, Sept. 2000.
- [32] D. Wu, T. Hou, W. Zhu, H.-J. Lee, T. Chiang, Y.-Q. Zhang, H. J. Chao, "MPEG-4 Video Transport over the Internet: A Summary," IEEE Circuits and Systems Magazine, vol. 2, no. 1, pp. 43-46, 2002.
- [33] V. Jacobson, "Congestion avoidance and control", in Proceedings of ACM SIGCOMM 1988, pp.314-329, August 1988.
- [34] T. Turletti and C. Huitema, "Videoconferencing on the Internet," IEEE/ACM Transactions on Networking, vol. 4, no. 3, pp. 340-351, June 1996.
- [35] J-C. Bolot, T. Turletti, and I. Wakeman, "Scalable feedback control for multicast video distribution in the Internet," in Proceedings of ACM SIGCOMM 1994, pp. 58-67, London, UK, September 1994.
- [36] N. Venkitaraman, T. Kim, K.-W. Lee, S. Lu, and V. Bharghavan, "Design and Evaluation of Congestion Control Algorithms in the Future Internet," In Proceedings of ACM SIGMETRICS'99, Atlanta, Georgia, May 1999.



- 
- [37] S. Floyd, "TCP and Explicit Congestion Notification", ACM Computer Communication Review, Vol. 24 No. 5, October 1994, p. 10-23.
- [38] D.-M. Chiu and R. Jain, "Analysis of the Increase and Decrease Algorithms for Congestion Avoidance in Computer Networks," Elsevier Journal of Computer Networks and ISDN, Vol. 17, No. 1, June 1989, pp. 1-14.
- [39] R. Rejaie, M. Handley, and D. Estrin, "RAP: An end-to-end rate-based congestion control mechanism for real-time streams in the Internet," in Proceedings of Infocom'99, New York, NY, March 1999.
- [40] Yang Richard Yang and Simon S. Lam, "General AIMD Congestion Control", Technical Report TR-00-09 Department of Computer Sciences, The University of Texas, Austin, Texas, U.S.A., Sept. 2000.
- [41] D. Bansal and H. Balakrishnan. "Binomial Congestion Control Algorithms", In Proceedings Infocom 2001, April 2001.
- [42] S. Floyd, K. Fall, "Promoting the use of end-to-end congestion control in the Internet," IEEE/ACM Trans. on Networking, vol. 7, no. 4, pp. 458-472, Aug. 1999.
- [43] IETF Request for Comments: 1191, J. Mogul and S. Deering, "Path MTU discovery," Nov. 1990.
- [44] J. Padhye, V. Firoui, D. Towsley, J. Kurose, "Modelling TCP throughput: A simple model and its empirical validation", In Proceedings of ACM SigComm '98, Vancouver, Oct. 1998
- [45] X. Wang and H. Schulzrinne, "Comparison of adaptive Internet multimedia applications," IEICE Trans. on Communications, vol. E82-B, no. 6, pp. 806-818, June 1999.
- [46] T.V. Lakshman, U. Madhow, "Performance analysis of window – based flow control using TCP/IP: the effect of high bandwidth-delay products and random loss," IFIP Transactions C26, High Performance Networking V, pp. 135-150, Grenoble, France, 1994.
- [47] IETF RFC 3448, "TCP Friendly Rate Control (TFRC): Protocol Specification", Jan. 2003, <ftp://ftp.rfc-editor.org/in-notes/rfc3448.txt>
- [48] J. Padhye, J. Kurose, D. Towsley, R. Koodli. "A model based TCP-friendly rate control protocol". In Proceedings of NOSSDAV'99, June 1999
- [49] D. Sisalem, A. Wolisz, "LDA+: A TCP-Friendly Adaptation Scheme for Multimedia Communication", In Proceedings of IEEE International Conference on Multimedia and Expo III, 2000
- [50] Dorgham Sisalem and Henning Schulzrinne, "The loss-delay based adjustment algorithm: A TCP-friendly adaptation scheme," in Proceedings of NOSSDAV '98, July 1998.
- [51] D. Sisalem, F. Emanuel, H. Schulzrinne. "The direct adjustment algorithm: a TCP-friendly adaptation scheme", Technical Report GMD-FOKUS August 1997. <http://www.focus.gmd.de/usr/sislem>
- [52] J. C. Bolot, "End-to-End packet delay and loss behaviour in the Internet", In Proceedings of ACM SigComm'93, pp289-298, San Francisco, California, Sept 1993.
- [53] S. McCanne, V. Jacobson, M. Vetterli, "Receiver-driven layered multicast," in Proceedings

---

ACM SIGCOMM'96, pp. 117-130, Aug. 1996.

[54] L. Wu, R. Sharma, and B. Smith, "Thin streams: An architecture for multicasting layered video", In Proceedings of NOSSDAV'97, St. Louis, USA, May 1997.

[55] X. Li, S. Paul, and M. H. Ammar, "Layered video multicast with retransmissions (LVMR): evaluation of hierarchical rate control," in Proceedings IEEE Infocom'98, vol. 3, pp. 1062-1072, March 1998.

[56] L. Vicisano, L. Rizzo, and J. Crowcroft, "TCP-like congestion control for layered multicast data transfer," in Proceedings IEEE Infocom'98, vol. 3, pp. 996-1003, March 1998.

[57] I. Rhee, V. Ozdemir, Y. Yung, "TEAR: TCP emulation at receivers - flow control for multimedia streaming", Technical Report, Department of Computer Science, North Carolina State University, Raleigh, North Carolina, U.S.A., April 2000.

[58] R. Rejaie, M. Handley, D. Estrin. "Quality Adaptation for Unicast Audio and Video", In Proceedings ACM SIGCOMM, September 1999.

[59] R. Rejaie, M. Handley, D. Estrin, "Layered Quality Adaptation for Internet Video Streaming", IEEE Journal on Selected Areas of Communications (JSAC), Winter 2000. Special issue on Internet QOS.

[60] H. M. Smith, M. W. Mutka, E. Tornig, "Bandwidth allocation for layered multicasted video," in Proceedings IEEE Int. Conference on Multimedia Computing and Systems, vol. 1, pp. 232-237, June 1999.

[61] S. Y. Cheung, M. Ammar, X. Li, "On the use of destination set grouping to improve fairness in multicast video distribution", In Proceedings IEEE Infocom '96, San Francisco, CA, Mar. 1996, pp. 553-560.

[62] S. Cen, C. Pu, J. Walpole, "Flow and congestion control for Internet streaming applications," in Proceedings of Multimedia Computing and Networking 1998, Jan. 1998.

[63] D. Sisalem, F. Emanuel, "QoS control using adaptive layered data transmission", In Proceedings of IEEE Multimedia Systems, (Austin, Texas), June 1998.

[64] B. J. Vickers, C. Albuquerque, T. Suda, "Adaptive multicast of multi-layered video: rate-based and credit-based approaches ", In Proceedings of IEEE Infocom '98, San Francisco, 1998.

[65] R. Jain, S. Kalyanaraman, R. Goyal, S. Fahmy, and R. Viswanathan. "ERICA switch algorithm: a complete description ", in ATM Forum'96, Aug.1996.

[66] M. Hemy, U. Hengartner, P. Steenkiste, and T. Gross, "MPEG system streams in best-effort networks," in Proc. IEEE Packet Video'99, New York, April 26-27, 1999.

[67] K. Sripanidkulchai and T. Chen, "Network-adaptive video coding and transmission," in SPIE Proceedings Visual Communications and Image Processing (VCIP'99), San Jose, CA, Jan. 1999.

[68] Z.-L. Zhang, S. Nelakuditi, R. Aggarwa, R. P. Tsang, "Efficient server selective frame discard algorithms for stored video delivery over resource constrained networks," in Proceedings IEEE Infocom'99, pp. 472-479, New York, March 1999.

[69] A. Eleftheriadis, D. Anastassiou, "Meeting arbitrary QoS constraints using dynamic rate shaping of coded digital video," in Proceedings NossDav'95, pp. 95-106, April 1995.

- 
- [70] Z.-L. Zhang, S. Nelakuditi, R. Aggarwa, and R. P. Tsang, "Efficient server selective frame discard algorithms for stored video delivery over resource constrained networks," in Proceedings IEEE Infocom'99, New York, March 1999, pp. 472-479.
- [71] N. Yeadon, F. Garcia, D. Hutchison, D. Shepherd, "Filters: QoS support mechanisms for multipeer communications," IEEE Journal on Selected Areas in Communications, vol. 14, no. 7, pp. 1245-1262, Sept. 1996.
- [72] Z.-L. Zhang, S. Nelakuditi, R. Aggarwa, and R. P. Tsang, "Efficient server selective frame discard algorithms for stored video delivery over resource constrained networks," in Proc. IEEE Infocom'99, pp. 472-479, New York, March 1999.
- [73] A. Eleftheriadis, D. Anastassiou, "Meeting arbitrary QoS constraints using dynamic rate shaping of coded digital video," in Proc. Nossdav'95, pp. 95-106, April 1995.
- [74] J.C Bolot, T. Turletti, "A rate control mechanism for packet video in the internet", In IEEE Infocom, November 1994.
- [75] P. Nee, K. Jeffay, G. Danneels, "The Performance of Two-Dimensional Media Scaling for Internet Videoconferencing",. In Proc. of NOSSDAV, May 1997.
- [76] W. Zeng, B. Liu, "Rate shaping by block dropping for transmission of mpeg-precoded video over channels of dynamic bandwidth", In ACM Multimedia, 1996.
- [77] S. Ramanathan, P.V. Rangan, H.M. Vin, S.S Kumar. "Enforcing application-level QoS by frame-induced packet discarding in video communications", Elsevier Journal of Computer Communications, vol. 18, no. 10, pp.742-754, Oct 1995.
- [78] F. C. Martins, W. Ding, E. Feig, "Joint control of spatial quantization and temporal sampling for very low bit-rate video," in Proceedings of IEEE Int. Conference on Acoustics, Speech, and Signal Processing (ICASSP'96), vol. 4, pp. 2072-2075, May 1996.
- [79] W. Ding, "Joint encoder and channel rate control of VBR video over ATM networks," IEEE Trans. on Circuits and Systems for Video Technology, vol. 7, pp. 266-278, April 1997.
- [80] J. Lee, B.W. Dickenson, "Rate-distortion optimized frame type selection for MPEG encoding," IEEE Trans. on Circuits and Systems for Video Technology, vol. 7, pp. 501-510, June 1997.
- [81] A. Vetro, H. Sun, Y. Wang, "MPEG-4 rate control for multiple video objects," IEEE Trans. on Circuits and Systems for Video Technology, vol. 9, no. 1, pp. 186-199, Feb. 1999.
- [82] T. Weigand, M. Lightstone, D. Mukherjee, T. G. Campbell, S. K. Mitra, "Rate-distortion optimized mode selection for very low bit-rate video coding and the emerging H.263 standard," IEEE Trans. on Circuits and Systems for Video Technology, vol. 6, pp. 182-190, April 1996.
- [83] "Coding of Audio-Visual Objects, Part-2 Visual, Amendment 4: Streaming Video Profile", ISO/IEC 14496-2/FPDAM4, July 2000
- [84] S. Li, F. Wu, Y.-Q. Zhang, "Study of a new approach to improve FGS video coding efficiency", ISO/IEC JTC1/SC29/WG11, MPEG99/M5583, Dec. 1999.
- [85] W. Li, "Bit-plane coding of DCT coefficients for fine granularity scalability," ISO/IEC JTC1/SC29/WG11, MPEG98/M3989, Oct. 1998.
- [86] W. Li, "Streaming video profile in MPEG-4," IEEE Trans. on Circuits and Systems for Video

---

Technology, vol. 11, no. 1, Feb. 2001.

[87] W. Li, "Overview of Fine granularity scalability in MPEG-4 video standard", IEEE Transactions Circuits and Systems of Video Technology, vol. 11, no.3. pp 301-317, March 2001

[88] H. Radha, M. van der Schaar, Y. Chen, "The MPEG-4 fine-grained scalable video coding method for multimedia streaming over IP", IEEE Transactions on Multimedia, vol.3, no.1, March 2001

89 B.-F. Hung and C.-L. Huang, "Content-Based FGS Coding Mode Determination for Video Streaming Over Wireless Networks", JSAC Recent Advances in Wireless Multimedia, December 2003, Volume 21, Number 10

[90] F. Wu, S. Li, Y.-Q. Zhang, "A framework for efficient progressive fine granularity scalable video coding," IEEE Trans. on Circuits and Systems for Video Technology, vol. 11, no. 1, Feb. 2001.

[91] ISO/IEC JTC 1/SC 29/WG 11, "Information technology: coding of audio-visual objects, part 1: systems, part 2: visual, part 3: audio," FCD 14496, Dec. 1998.

[92] D. Wu, Y. T. Hou, W. Zhu, H.-J. Lee, T. Chiang, Y.-Q. Zhang, H. J. Chao, "On end-to-end architecture for transporting MPEG-4 video over the Internet," IEEE Trans. on Circuits and Systems for Video Technology, vol. 10, no. 6, Sept. 2000.

[93] T. Chiang, Y.-Q. Zhang, "A new rate control scheme using quadratic rate distortion model", IEEE Trans. on Circuits and Systems for Video Technology, vol. 7, no. 1, pp. 246-250, Feb. 1997.

[94] C. Y. Hsu, A. Ortega, A. R. Reibman, "Joint selection of source and channel rate for VBR transmission under ATM policing constraints," IEEE Journal on Selected Areas in Communications, vol. 15, pp. 1016-1028, Aug. 1997.

[95] J. Lee and B.W. Dickenson, "Rate-distortion optimized frame type selection for MPEG encoding," IEEE Trans. on Circuits and Systems for Video Technology, vol. 7, pp. 501-510, June 1997.

[96] H. Sun, W. Kwok, M. Chien, and C. H. J. Ju, "MPEG coding performance improvement by jointly optimizing coding mode decision and rate control," IEEE Trans. on Circuits and Systems for Video Technology, vol. 7, pp. 449-458, June 1997.

[97] T. Weigand, M. Lightstone, D. Mukherjee, T. G. Campbell, S. K. Mitra, "Rate-distortion optimized mode selection for very low bit-rate video coding and the emerging H.263 standard", IEEE Trans. on Circuits and Systems for Video Technology, vol. 6, pp. 182-190, April 1996.

[98] Q. Zhang, W. Zu, Y-Q Zhang, "QoS Adaptive Multimedia Streaming over 3G Wireless Channels", Second International Symposium on Mobile Multimedia Systems and Applications (MMSA) 2000, Nov., 2000, Delft, The Netherlands

[99] P. Cheng, J. Li, C.-C.J. Kuo. "Rate control for an embedded wavelet video coder". IEEE Transactions on Circuits and Systems for Video Technology, vol. 7 no. 4, 1997.

[100] David Taubman and Avidesh Zakhor. "A Common framework for rate and distortion based scaling of highly scalable compressed video". IEEE Transactions on Circuits and Systems for Video Technology, 6(4), August 1996.

[101] J. Tham, S. Ranganath and A. Kassim. "Highly scalable wavelet-based video codec for very

- 
- low bit-rate environment". IEEE Journal of Selected Areas of Communications, Special Issue: Very Low Bit Rate Coding, 1996.
- [102] C. I. Podilchuk, N. S. Jayant, and N. Farvardin. "Three-dimensional sub band coding of video". IEEE Transactions on Image Processing, vol. 4, no. 2, Feb. 1995.
- [103] G. Morrison, "Video Transcoders with low delay", IEEE Transactions on Communications, Vol. E80-B, No. 6, June 1997
- [104] E. Amir, S. McCanne, R. Katz, "An active service framework and its application to real-time multimedia transcoding ", In Proceedings of ACM SIGCOMM '98, Vancouver, BC, Canada, Sep 1998.
- [105] G. Cheung, T. Yoshimura, "Streaming Agent: A Network Proxy for Media Streaming in 3G Wireless Networks", IEEE Packet Video Workshop 2002
- [106] I. Kouvelas, V. Hardman and J. Crowcroft, "Network adaptive continuous-media applications through self organized transcoding ", in Proceedings of NossDav'98, 8-10 July 1998, Cambridge, UK.
- [107] G. Davis, J. Danskin, "Joint source and channel coding for Internet image transmission," in Proc. SPIE Conference on Wavelet Applications of Digital Image Processing XIX, Denver, Aug. 1996.
- [108] M. Podolsky, K.Yano, S. McCanne, "A RTCP-based retransmission protocol for unicast RTP streaming multimedia," IETF Internet draft, Oct. 1999  
<http://www.cs.columbia.edu/~hgs/rtp/drafts/draft-podolsky-avt-rtprx-01.txt>
- 109 V. M. Stankovic, R. Hamzaoui, Y. Charfi, and Z. Xiong, "Real-time Unequal Error Protection Algorithms for Progressive Image Transmission", JSAC recent Advances in Wireless Multimedia, December 2003, Volume 21, Number 10
- [110] A. Albanese, J. Blomer, J. Edmonds, M. Luby, M. Sudan, "Priority encoding transmission," IEEE Trans. on Information Theory, vol. 42, no. 6, pp. 1737-1744, Nov. 1996.
- [111] W. Tan, A. Zakhor, "Multicast transmission of scalable video using receiver-driven hierarchical FEC", in Proc. Packet Video Workshop 1999, New York, April 1999.
- 112 Y. S. Chan and J. W. Modestino, "A Joint Source Coding-Power Control Approach for Video Transmission Over CDMA Networks", JSAC Recent Advances in Wireless Multimedia, December 2003, Volume 21, Number 10
- [113] P. A. Chou, "Joint source/channel coding: a position paper," in Proc. NSF Workshop on Source-Channel Coding, San Diego, CA, USA, Oct. 1999.
- [114] M. Podolsky, M.Vetterli, S. McCanne, "Limited retransmission of real-time layered multimedia", in Proc. IEEE Workshop on Multimedia Signal Processing, pp. 591-596, Dec. 1998.
- [115] M. Podolsky, K.Yano, S. McCanne, "A RTCP-based retransmission protocol for unicast RTP streaming multimedia" IETF Internet draft, draft-podolsky-avt-rtprx-00.txt, Oct. 1999.
- [116] ISO/IEC JTC 1/SC 29/WG 11, "Information technology: coding of audio-visual objects, part 1: systems, part 2: visual, part 3: audio," FCD 14496, Dec. 1998.
- [117] R. Talluri, "Error-resilience video coding in the ISO MPEG-4 standard," IEEE Communications Magazine, pp. 112-119, June 1998.

- 
- [118] Y. Wang, Q.-F. Zhu, "Error control and concealment for video communication: A review", Proceedings of the IEEE, vol. 86, no. 5, pp. 974-997, May 1998.
- [119] ITU-T Recommendation P.910, "Subjective video quality assessment methods for multimedia applications"
- [120] EBU B/AIM 060 "Method for the Subjective Assessment of Intermediate Audio Quality"
- [121] A. J. Mason, "The MUSHRA audio subjective test method", BBC R&D White Paper, WHP 038, September 2002
- [122] ISO/IEC JTC1/SC29/WG11 N3075, "Report on the MPEG-4 Audio Version 2 Verification Test".
- [123] ITU-T Recommendation P.800, "Methods for subjective determination of transmission quality"
- [124] ITU-T Recommendation J.140, "Subjective picture quality assessment for digital cable television systems"
- [125] Christian J. van den Branden Lambrecht and Olivier Verscheure, "Perceptual Quality Measure using a Spatio-Temporal Model of the Human Visual System", Proceedings of SPIE 96, San Jose, CA
- [126] Video Quality Experts Group (VQEG), <http://www.its.bldrdoc.gov/vqeg/>
- [127] ITU-T Recommendation J.143, "User requirements for objective perceptual video quality measurements in digital cable television"
- [128] ITU-T Recommendation J.144, "Objective perceptual video quality measurement techniques for digital cable television in the presence of a full reference"
- [129] A. Pessoa, A. Falcão, R. Nishihara, A. Silva, R. Lotufo, "Video quality assessment using objective parameters based on image segmentation", SMPTE Journal, December 1999
- [130] J.Lubin, "A visual discrimination model for imaging system design and evaluation", book chapter pp.245 –283 in "Vision Models for Target Detection and Recognition" by E. Peli, World Scientific Publishing, 1995. ISBN 981-02-2149-5
- [131] J.Lubin, D.Fibush, "Sarnoff JND vision model", T1A1.5 Working Group Document #97-612,T1 Standards Committee, 1997.
- [132] C.J.van den Branden Lambrecht, "Color moving pictures quality metric." in Proc. ICIP, vol.1, pp.885 –888, Lausanne, Switzerland, 1996.
- [133] C.J.van den Branden Lambrecht et al, "Quality assessment of motion rendition in video coding", IEEE Transactions on Circuits and Systems Video Technology vol. 9, no. 5: pp766 –782, 1999.
- [134] S.Winkler, "A perceptual distortion metric for digital color video." In Proc. SPIE, vol. 3644, pp.175 –184, San Jose, CA, 1999.
- [135] P.Lindh, C.J.van den Branden Lambrecht: "Efficient spatio-temporal decomposition for perceptual processing of video sequences." in Proc. ICIP, vol.3, pp.331 –334, Lausanne, Switzerland, 1996.
- [136] S.Winkler, "Quality metric design: A closer look." in Proc. SPIE, vol. 3959, pp.37–44, San

---

Jose, CA, 2000.

[137] A. B. Watson et al., "Design and performance of a digital video quality metric", in Human Vision, Visual Processing and Digital Display IX, 1999, San Jose, CA: SPIE, Bellingham, WA.

[138] A.B. Watson, J. Hu, J.F. McGowan, "DVQ: A digital video quality metric based on human vision", In Journal of Electronic Imaging, 2000.

[139] A.P. Hekstra, J.G. Beerends, et al. "PVQM - A perceptual video quality measure ", Signal Processing: Image Communication, Vol. 17, no. 10, 2002, pp. 781-798, 2002 Elsevier Science B.V

[140] ITU-T Recommendation P.861, "Objective quality measurement of telephone-band (300-3400Hz) speech codecs", August 1996

[141] S. Wolf, M. Pinson, "Spatial-temporal distortion metrics for in-service quality monitoring on any digital video system," SPIE International Symposium on Voice, Video, and Data Communications, Boston, MA, September 11-22, 1999.

[142] Olivier Verscheure, Pascal Frossard, Maher Hamdi, "MPEG-2 Video Services over Packet Networks: Joint Effect of Encoding Rate and Data Loss on User-Oriented QoS", Proc. NOSSDAV 98, Cambridge, England, July 1998.

[143] Christian J. van den Branden Lambrecht and Olivier Verscheure, "Perceptual Quality Measure using a Spatio-Temporal Model of the Human Visual System", Proceedings of SPIE 96, San Jose, CA

[144] R. L. De Valois and K. K. De Valois. Spatial Vision. Oxford University Press, 1988.

[145] ITU-T J.143, User requirements for objective perceptual video quality measurements in digital cable television.

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

[147] ITU-T P.910, Subjective video quality assessment methods for multimedia applications.

[148] G. Ghinea, J.P. Thomas, "QoS Impact on User Perception and Understanding of Multimedia Video Clips", ACM Multimedia 1998, Bristol

[149] ITU-T Recommendation P.910, "Subjective video quality assessment methods for multimedia applications"

[150] A. Watson, M.A. Sasse, "Measuring Perceived Quality of Speech and Video in Multimedia Conferencing Applications"

[151] A. Watson, M.A. Sasse, "Evaluating audio and video quality in low-cost multimedia conferencing systems", Interacting with Computers, 1996, 8(3), 255-275

[152] H. de Ridder, R. Hamberg, "Continuous assessment of image quality", SMPTE Journal, February 1997, 123-128

[153] VQEG, <http://www.vqeg.org/>

[154] Test Sequences from CCIR: <ftp://ftp.crc.ca/>

[155] ITU-T Recommendation P.910, "Subjective video quality assessment methods for multimedia applications"

[156] StreamCrest, <http://www.Streamcrest.com>

- 
- [157] Nick Efford, Digital Image Processing Using Java, Addison Wesley; ISBN: 0201596237.
- [158] M. W. Levine, Fundamentals of Sensation and Perception (3rd Ed.), Oxford University Press, 2000.
- [159] G. Ghinea, J.P. Thomas, "QoS Impact on User Perception and Understanding of Multimedia Video Clips", ACM Multimedia 1998
- [160] J.T. McClave, F. H. Dietrich, T. Sincich, "Statistics", 7<sup>th</sup> Edition Prentice-Hall, ISBN 0-13-471542-X
- [161] S. Wolf and M. Pinson, "In-service performance metrics for MPEG-2 video systems," Made to Measure 98 - Measurement Techniques of the Digital Age Technical Seminar," technical conference jointly sponsored by the International Academy of Broadcasting (IAB), the ITU, and the Technical University of Braunschweig (TUB), Montreux, Switzerland, November 12-13, 1998.
- [162] Stephen Wolf and Margaret H. Pinson, "Spatial-temporal distortion metrics for in-service quality monitoring on any digital video system," SPIE International Symposium on Voice, Video, and Data Communications, Boston, MA, September 11-22, 1999.
- [163] S. Wolf, M. Pinson, "NTIA Report 02-392: Video Quality Measurement Techniques", Institute for Telecommunication Sciences, <http://its.blrdoc.gov/index.php>
- [164] R. S. Fish, G Ghinea and J P Thomas, "Mapping Quality of Perception to Quality of Service for a Runtime Adaptable Communication System", Proc. SPIE/ACM Multimedia Computing and Networking MMNC '99, San Jose
- [165] A. Watson, M.A. Sasse, "Measuring Perceived Quality of Speech and Video in Multimedia Conferencing Applications"
- [166] A. Watson, M.A. Sasse, "Evaluating audio and video quality in low-cost multimedia conferencing systems", Interacting with Computers, 1996, 8(3), 255-275
- [167] H. de Ridder, R. Hamberg, "Continuous assessment of image quality", SMPTE Journal, February 1997, 123-128
- [168] ITU-R Recommendation BT.500-7: Methodology for the subjective assessment of the quality of television pictures.
- [169] T. Alpert, J.P. Evain, European Broadcasting Union Technical Report, "Subjective quality evaluation – The SSCQE and DSCQE methodologies"  
[http://www.ebu.ch/trev\\_271-evain.pdf](http://www.ebu.ch/trev_271-evain.pdf)
- [170] Helix Streaming Server, <https://www.helixcommunity.org/>
- [171] "Darwin Streaming Server", <http://developer.apple.com/darwin/projects/streaming/>
- [172] "QuickTime Streaming",  
<http://developer.apple.com/documentation/QuickTime/RM/Streaming/StreamingClient/StreamingClient.pdf>
- [173] "Quicktime Streaming Server Modules",  
<http://developer.apple.com/documentation/QuickTime/QTSS/QTSS.pdf>
- [174] "Broadcasting"  
<http://developer.apple.com/documentation/QuickTime/QTBroadcasting/BroadcastQT5.pdf>



- 
- [175] 3GPP TS 26.244 V0.2.0, Tdoc S4-030417, "3GPP file format (3GP)", Release 6, February 2003
- [176] 3GPP TR 26.937 V1.3.0, "Transparent end-to-end packet switched streaming service (PSS) – RTP Usage model" Technical Report Release 5, January 2003
- [177] 3GPP TS 26.234 V0.2.1, "Transparent end-to-end packet switched streaming service (PSS) – Protocols and codecs" Technical Report Release 6, February 2003
- [178] Apple Computer, "QuickTime File Format",  
<http://developer.apple.com/techpubs/quicktime/qtdevdocs/PDF/QTFileFormat.pdf>
- [179] Quicktime 6.3 +3GPP
- [180] Tdoc S4 (03) 0181, Ericsson, "3GPP Server File Format" 3GPP February 2003
- [181] C. Krasic, J. Walpole, W-C Feng, "Quality Adaptive Media Streaming by Priority Drop", Proc. NOSSDAV'03, Monterey, CA, 1-4 June 2003.
- [182] G. Conklin, G. Greenbaum, K. Lillevold, A. Lippman, Y. Reznik, "Video Coding for Streaming Media delivery on the Internet", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 11, No. 3, March 2001
- [183] Dorgham Sisalem, Henning Schulzrinne "The Loss-Delay based adjustment algorithm: a TCP-friendly adaptation scheme" Workshop on Network and Operating System Support for Digital Audio and Video, 1998.
- [184] Dorgham Sisalem, Adam Wolisz "LDA+: A TCP-friendly adaptation scheme for multimedia communication", IEEE International Conference on Multimedia and Expo (III) 2000
- [185] Dorgham Sisalem, Adam Wolisz, "MLDA: A TCP-friendly congestion control framework for heterogeneous multicast environments," in Eighth International Workshop on Quality of Service (IWQoS 2000), Pittsburgh, PA, June 2000
- [186] V. Jacobson S. McCanne, M. Vetterli. "Receiver-driven layered multicast." In Proc. of ACM SIGCOMM'96, pages 117--130, Stanford, CA, August 1996.
- [187] N. Cranley, L. Murphy, P. Perry "User-Perceived Quality Aware Adaptive Delivery of MPEG-4 Content", Proc. NOSSDAV'03, Monterey, California, June, 2003
- [188] J. C. Bolot, "End-to-End packet delay and loss behaviour in the Internet", SigComm Symposium on Communications Architectures and Protocols, pp289-298, San Francisco, California, Sept 1993. ACM also in Computer Communication Review 23 (4), Oct. 1992
- [189] RFC 3611, "RTP Control Protocol Extended Reports (RTCP XR)", November 2003
- [190] Nick Feamster, Hari Balakrishnan "Packet loss recovery for streaming video", Proc. of 12th International Packet Video Workshop, April 2002
- [191] G. Seckin, R. Brooks, "Challenges in Wireless Media Streaming"
- [192] D. Chambers, M. Sloman, "A Survey of Quality of Service in Mobile Computing Environments", IEEE Communications Surveys, 2<sup>nd</sup> Quarter 1999
- [193] QuickTime ISMA Support,  
[http://developer.apple.com/documentation/QuickTime/QT6WhatsNew/Chap1/chapter\\_1\\_section\\_9.html](http://developer.apple.com/documentation/QuickTime/QT6WhatsNew/Chap1/chapter_1_section_9.html)

---

[194] StreamingMedia.com, “ISMA Unveils MPEG-4 Specs”,

<http://www.streamingmedia.com/article.asp?id=7926>

[195] MPEG4IP, <http://mpeg4ip.net/>