

# NOTE TO USERS

This reproduction is the best copy available.

**UMI**<sup>®</sup>



**A HYBRID RATE ADAPTATION  
FRAMEWORK FOR MPEG-4 FGS VIDEO  
STREAMING OVER IP**

Colin Xialin Huang

B. Eng, Reliability Engineering  
Beijing University of Aeronautics and Astronautics  
Beijing, 1990

A thesis presented to Ryerson University  
in partial fulfillment of the requirements for the degree of  
Master of Applied Science  
in the program of  
Electrical and Computer Engineering

Toronto, Ontario, Canada  
May 2004

© Colin Xialin Huang 2004

UMI Number: EC52973

### INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

**UMI**<sup>®</sup>

---

UMI Microform EC52973

Copyright 2008 by ProQuest LLC.

All rights reserved. This microform edition is protected against unauthorized copying under Title 17, United States Code.

ProQuest LLC  
789 E. Eisenhower Parkway  
PO Box 1346  
Ann Arbor, MI 48106-1346



## **ABSTRACT**

### **A hybrid rate adaptation framework for MPEG-4 FGS video streaming over IP**

© Colin Xialin Huang 2004

Master of Applied Science  
Department of Electrical and Computer Engineering  
Ryerson University

There are increasing demands for real-time streaming video applications over the Internet. However, the current generation Internet was not originally designed for real-time streaming applications and only provides best-effort services, so there are many challenges in the deployment of video streaming applications over the Internet.

This thesis investigates a hybrid end-to-end rate adaptation framework that provides application-level enhancements to achieve Quality of Service (QoS) for MPEG-4 FGS-Encoded video streaming over the Internet. The receivers detect the available bandwidth on the path and the terminal process capabilities based on the packet-loss ratio and then determine their subscribing rate of video streams. The sender adjusts the transmission rate based on the proportion of load status feedbacks from the receivers. The sender and the receivers act together to minimize the possibility of network congestion by adjusting the transmission rate to match the network conditions.

This framework achieves inter-receiver fairness in a heterogeneous multicast environment and improves QoS stability for MPEG-4 FGS video streaming over the Internet.

## TABLE OF CONTENTS

Table of Contents.....	vi
List of figures .....	ix
List of tables .....	xi
Acknowledgments .....	xii
1. Introduction .....	1
1.1 Motivation.....	1
1.2 Streaming vs. download.....	2
1.3 Characteristics of streaming applications .....	2
1.4 Related works .....	4
1.5 MPEG-4 standard .....	5
1.6 Internet Protocol Multicast .....	6
1.7 Contributions .....	7
1.8 Organization of the thesis .....	8
1.9 Publications.....	10
2. MPEG-4 overview.....	11
2.1 MPEG standards introduction.....	11
2.2 MPEG-4 Scene Features.....	13
2.3 MPEG-4 visual coding scheme.....	15
2.3.1 MPEG-4 frame types .....	15
2.3.2 Layered scalable coding .....	16

2.3.3 Content based functionality .....	18
2.3.4 Fine Granularity Scalability .....	19
2.4 MPEG-4 Transportation .....	24
2.4.1 Delivery Multimedia Integration Framework (DMIF) .....	24
2.4.2 MPEG-4 over IP .....	25
3. MPEG-4 FGS video transport over Internet .....	27
3.1 RTP/RTCP introduction .....	27
3.2 RTP/RTCP functionalities .....	29
3.3 Sender and Receiver Reports Analysis .....	32
4. Rate Adaptation Framework .....	37
4.1 Background .....	37
4.2 Rate adaptation mechanisms .....	38
4.2.1 Network-based vs. End-to-end .....	38
4.2.2 Sender-adaptive vs. Receiver-driven .....	39
4.2.2.1 Sender-adaptive approach .....	39
4.2.2.2 Receiver-driven approach .....	40
4.3 The Proposed Framework .....	40
4.3.1 MPEG-4 FGS BL and EL .....	43
4.3.2 Rate Controllers .....	44
4.3.2.1 Rate controllers initialization .....	46
4.3.2.2 Packet-Loss Ratio (PLR) Analysis .....	46
4.3.2.3 Receivers subscribe rate control and algorithm .....	48

4.3.2.4 Sender rate control and algorithm .....	51
4.3.3 Avoiding feedback implosion .....	54
5. Test bed .....	56
5.1 The JMF architecture .....	56
5.2 JMF RTP APIs .....	57
5.3 Implementation .....	59
5.3.1 Session manager .....	59
5.3.2 Receiving and presenting RTP streams .....	61
5.3.3 Transmitting RTP streams .....	62
6. EXPERIMENT evaluation .....	64
6.1 Video quality evaluation .....	64
6.1.1 Subjective video quality assessment .....	64
6.1.2 Objective video quality measurements .....	67
6.1.3 Perceptual objective video quality assessment .....	69
6.2 Experiments .....	69
6.2.1 Test configuration and setup .....	69
6.2.2 Rate controller parameters settings .....	71
6.2.3 PSNR comparisons .....	75
7. Conclusions .....	79
References .....	81
Glossary .....	89

## LIST OF FIGURES

Figure	Page
Figure 1 Multicast transmission sends a single multicast packet addressed to all intended recipients.....	7
Figure 2 An Example of an MPEG-4 Scene.....	15
Figure 3 A frame sequence example .....	16
Figure 4 Layered Scalable Coding .....	17
Figure 5 the VLBV Core and the Generic MPEG-4 Coder.....	19
Figure 6 FGS Bitplane coding .....	20
Figure 7 Archiving Rate Scalability.....	21
Figure 8 FGS Bitplane Coding Gain .....	21
Figure 9 Performance of FGS vs. layered scalability .....	23
Figure 10 The MPEG-4 System Layer Model .....	25
Figure 11 RTP packet to logical SL packet mapping .....	26
Figure 12 RTCP/RTP/UDP/IP Stack .....	28
Figure 13 RTP Header .....	30
Figure 14 Receiver report RTCP packet format.....	33
Figure 15 A scenario for end-to-end hybrid rate adaptation.....	42
Figure 16 FGS Coding.....	44
Figure 17 Video Receiver Architecture.....	48

Figure 18 The receiver states transition .....	50
Figure 19 Received MPEG-4 FGS streaming .....	51
Figure 20 The sender in the framework .....	53
Figure 21 MPEG-4 FGS streaming at the sender .....	54
Figure 22 High-level JMF Architecture .....	56
Figure 23 JMF Media Processing Model.....	57
Figure 24 High-level JMF RTP Architecture.....	59
Figure 25 JMF Session Manager Object.....	60
Figure 26 JMF Player Object.....	61
Figure 27 RTP reception using JMF.....	62
Figure 28 RTP Transmission using JMF .....	62
Figure 29 Example of DSCQE side-by-side presentation .....	65
Figure 30 Examples of Digital Video Impairments .....	66
Figure 31 Test bed layout.....	70
Figure 32 PSNR and $\alpha$ relationship for "Foreman" sequence at 128kbps.....	73
Figure 33 PLR, PSNR and View Quality relationship examples.....	74
Figure 34 Comparison with Foreman sequence at 128kbps .....	76
Figure 35 Comparison with Foreman sequence at 64kbps .....	77
Figure 36 Comparison with Suzie sequence at 128kbps.....	77
Figure 37 Comparison with Suzie sequence at 64kbps.....	78

## LIST OF TABLES

<b>Table</b>	<b>Page</b>
Table 1 Compressed video bitrates.....	3
Table 2 PSNR comparison between different filters using "Foreman" sequence at 128kbps.....	72
Table 3 PSNRs comparison scenario .....	76

## ACKNOWLEDGMENTS

I would like to express my most heartfelt gratitude and appreciation to my advisor, Professor Ling Guan, who has been a constant source of inspiration for me, professionally and personally. His technical savvy and excellent tastes for important research problems have helped me far more than guiding this research.

I would greatly appreciate my committee for the time and efforts in reviewing my work and providing very valuable advice.

I would like to thank my colleagues at Ryerson Multimedia Lab and Ryerson University for their help during my study and research.

I also would like to thank Canada Foundation for Innovation (CFI) and the Department of Electrical and Computer Engineering for providing an excellently equipped multimedia laboratory.

# Chapter One

## 1. INTRODUCTION

### 1.1 Motivation

There are increasing demands for real-time streaming video applications over the Internet. More and more people are expecting to experience video streaming applications such as watching live special events/news, telemedicine, video conferencing, video-on-demand, and distance learning. Streaming applications have the potential to change the way people communicate and access information.

However, the current generation Internet was not originally designed for real-time streaming applications and only provides best-effort services, so it makes the deployment of these applications more challenging than that of traditional data applications. The traditional data service architecture underlying the Internet is unable to fully meet the requirements of real-time streaming applications.

Although recent efforts have made progress in real-time video streaming over the Internet, the end users are still expecting dramatic improvements in streaming video quality.

The MPEG-4 standard has recently been enriched with a new video coding technique Fine Granularity Scalability (FGS), which is especially designed for video streaming. With FGS coding, the

enhancement layer video stream can be flexibly truncated at very fine granularity to adapt to the available network bandwidth and terminal decoding capability, so the intermediate network nodes can easily truncate the received streams to match the downstream subscribing rate without transcoding.

This thesis aims to explore a rate adaptation solution that provides application-level enhancements to achieve Quality of Service (QoS) for MPEG-4 FGS video streaming over the current Internet.

## **1.2 Streaming vs. download**

Traditionally, users download the entire video file and then play back the video file. In this way, the compressed video can be transmitted via reliable channels such as TCP/IP channels. However, full file transferring usually takes quite a long time, therefore it is unacceptable.

In contrast, in the streaming mode, the receiver stations can play the video while "data bits" are coming in. There is no need to wait for the complete video file to be transferred before the received video content is played out. Streaming offers a significant improvement over the download-and-then-play approach.

## **1.3 Characteristics of streaming applications**

Real-time streaming applications are quite different from traditional data applications such as Telnet, Email, FTP, etc. and

require services that cannot be delivered within the traditional data service architecture.

Compared to the traditional Internet data applications, video streaming applications typically:

- (1) Require high network throughput and consume significant bandwidth; Table 1 is a list of compressed video bitrates.

Table 1 Compressed Video bitrates

Format	Codec	Bitrate	Storage
QCIF	MPEG-4 Simple	50-300 kbps	400-2400 KB/min
CIF	MPEG-4 Simple	500-1000 kbps	4-8 MB/min
NTSC	MPEG-2 Main	2-6 Mbps	15-45 MB/min
HDTV	MPEG-2 High	15-30 Mbps	113-225 MB/min

- (2) Are easier to compensate for lost data than to compensate for large delays in receiving the data, i.e. are highly sensitive to delay;
- (3) Are often transmitted over UDP, placing responsibility for congestion control on the application layer;
- (4) Use multicast delivery techniques to save the network resource;
- (5) Have scalable quality and delivery rate to meet the heterogeneities of user terminals and network conditions.

Thus, there are many challenges have to be overcome for a streaming video/audio delivery solution that can achieve Quality of Service over the best-effort Internet [4].

#### **1.4 Related works**

In order to achieve QoS for such new types of traffic in the Internet, many research efforts have been devoted [4]. These solutions can be classified into two categories: network-level QoS control and application-level QoS control.

From network perspective, several approaches have been proposed to deliver network-level QoS for real-time streaming applications over the Internet, such as Integrated Services (IntServ) [34], ReSerVation Protocol (RSVP) [37], Differentiated Services (DiffServ) [3, 38], Multi-Protocol Label Switching (MPLS) [14], QoS routing [15, 16], Forward Error Correction (FEC), Content Delivery Network (CDN), etc. Network-based solutions usually change the network architecture or service model to meet the expected bandwidth and delay requirements of real-time streaming applications.

The application-level QoS control techniques include congestion control and error control. These techniques are typically deployed by the end systems and do not require any special support from the Internet. From source coding perspective, layered scalable and error-resilient video codecs have been proposed. A layered video codec deals with heterogeneity and time-varying nature of the Internet by adapting its bit rate to the

available bandwidth. An error-resilient codec attempts to cope with packet losses using error concealment techniques to improve visual quality at the expense of loss in coding efficiency. From congestion control perspective, rate adaptation is often proposed to avoid network congestion.

The framework proposed in this work take advantage of the achievements in MPEG-4 FGS encode techniques and IP multicasting, and provide a hybrid end-to-end rate-adaptive solution for MPEG-4 FGS video streaming over IP.

### **1.5 MPEG-4 Standard**

MPEG-4 [2] aims at easily deploying multimedia content for any and all platforms. MPEG-4 is a fundamental revolution in multimedia experience and functionality. It is foreseen that MPEG-4 video streaming over the Internet will be an important component of video streaming applications in the near future.

Few profiles of MPEG-4 are developed in response to the growing need on a video-coding standard for streaming video over the Internet. MPEG-4 provides the capability to distribute frame-based video over a wide range of bit rates (typically between 5 kbit/s and more than 1 Gbit/s) with high coding efficiency. It also provides Fine Granularity Scalability (FGS) [5, 46] to address a variety of challenging problems in delivering real-time video over the Internet.

## **1.6 Internet Protocol Multicast**

Internet Protocol (IP) multicast [63, 64] is a bandwidth-conserving technology that reduces traffic by simultaneously delivering a single stream of source information to multiple intended recipients. Multicast packets are replicated in the network by routers enabled with Protocol Independent Multicast (PIM) and other supporting multicast protocols, which makes the most efficient delivery of data to multiple receivers possible.

Video streaming applications require a large portion of the available network bandwidth for a single stream, so the best way to send to more than one receiver simultaneously is by using IP multicast.

Figure 1 demonstrates how data from one source is delivered to several interested recipients using IP multicast.

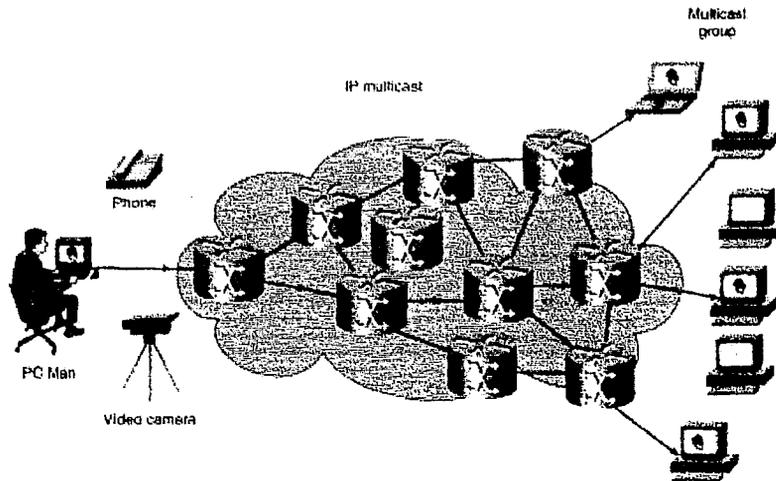


Figure 1 Multicast transmission sends a single multicast packet addressed to all intended recipients

## 1.7 Contributions

The rate adaptation framework proposed in this thesis can achieve application level QoS for MPEG-4 FGS video streaming over the Internet through end-to-end hybrid rate adaptation --- the receivers estimate the available bandwidth on the path based on the packet-loss ratio and then regulate the subscribing rate of video streams; the sender adjusts the transmission rate based on the proportion of load status feedbacks from the receivers.

The main contributions of this thesis are:

- (1) Combined the sender adaptive and the receiver driven rate control mechanisms into a hybrid rate adaptation framework that takes into account the rate-distortion characteristics of the source as well as the time-varying network

characteristics. Each receiver detects the available bandwidth on the path, and adjusts its receiving rate matching the available bandwidth. This leads to improve the inter-receiver fairness [27].

- (2) The proposed algorithm let the sender derive the rate adjustment decision from the proportion of receivers' load status, so that a receiver or a small proportion of receivers connected via a low bandwidth link cannot force the sender to provide low quality video to other receivers which have high bandwidth connections, and vice versa. In addition, this helps to reduce the overall Peak Signal to Noise Ratio (PSNR) standard deviations and therefore to avoid video quality oscillations and provide end users with smooth viewing experience.

This framework improves the inter-receiver fairness in a heterogeneous environment, reduces the standard deviations of PSNR at each receiver. The end-to-end approach can be employed by the end systems, so it is applicable under current Internet architecture.

## **1.8 Organization of the thesis**

The rest of the thesis is organized as follows:

Chapter 2 gave an overview of the MPEG-4 standard, and examined the Fine Granularity Scalability (FGS) features and functionalities that can benefit the transportation over Internet.

Chapter 3 MPEG-4 FGS Video Transport Over Internet. This chapter reviewed the transport protocols for real-time contents over IP. Analyzed the functionalities that can be deployed for end-to-end rate adaptation. Focus was put on the analysis of sender and receiver reports, which are basis for end-to-end feedback control.

Chapter 4 Rate Adaptation Framework. After examined the features and functionalities of MPEG-4 FGS video and RTP/RTCP and reviewed the related research works, an end-to-end hybrid rate adaptation framework for MPEG-4 FGS video streaming over Internet was proposed. The architecture, algorithms, detail features and functionalities of the framework were discussed.

Chapter 5 described a test bed application for effectiveness verification of the proposed framework. The application took advantage of the JMF RTP APIs. The goal is to show that the framework can improve the application-layer QoS for MPEG-4 FGS video streaming over IP.

Chapter 6 Experiments and Evaluation. This chapter reviewed the latest research in video quality assessment, chose suitable video quality metrics for the framework effectiveness measurement. Described the test bed configuration and setup. Experimental results analysis verified that the rate adaptation framework improved QoS for MPEG-4 FGS streaming over IP.

Chapter 7 Conclusion summarized the major work of this paper and ongoing and future working direction.

## **1.9 Publications**

The publication related to the research work of this thesis:

- Colin Huang, Ling Guan, "A New Rate Adaptation Approach for MPEG-4 Video Streaming over IP", IEEE Canadian Conference on Electrical and Computer Engineering, 2004.

# Chapter Two

## 2. MPEG-4 OVERVIEW

### 2.1 MPEG Standards Introduction

MPEG-4 is an ISO/IEC standard developed by **Moving Picture Experts Group (MPEG)**.

MPEG is a working group of ISO/IEC in charge of the development of international standards for compression, decompression, processing, and coded representation of moving pictures, audio, and their combination, in order to satisfy a wide variety of applications [30, 31]. The group has produced:

- (1) MPEG-1, the standard for such products as Video CD and MP3.
- (2) MPEG-2, the standard on which such products as Digital Television set top boxes and DVD are based.
- (3) MPEG-4, the standard for multimedia for the fixed and mobile web.
- (4) MPEG-7, formally named "Multimedia Content Description Interface".
- (5) Mpeg-21, the "Multimedia framework" standard.

MPEG-1 and MPEG-2 have been successful standards that have given rise to widely adopted commercial products, such as CD-interactive, digital audio broadcasting, and digital television.

However these standards are deeply limited in terms of the functionalities provided by the data representation models used.

MPEG-4 extends to many more application areas through its features like its extended bit-rate range, its scalability, its error resilience, its seamless integration of different types of 'objects' in the same scene, its interfaces to digital rights management systems and its powerful ways to build interactivity into content.

MPEG-4, builds on the proven success of three fields:

- (1) Digital television;
- (2) Interactive graphics applications (synthetic content);
- (3) Interactive multimedia (World Wide Web, distribution of and access to content)

MPEG-4 provides the standardized technological elements enabling the integration of the production, distribution and content access paradigms of the three fields.

MPEG-4 opens new frontiers in the way users will play with, create, re-use, access and consume audiovisual content.

MPEG-7 specifies how to describe content. MPEG-4 defines how to represent content. MPEG-7 complements MPEG-4 and helps manage the growing abundance of content.

MPEG-21 [31] aims to provide a truly interoperable multimedia framework. MPEG-21's goal is to describe a 'big picture' of how different elements to build an infrastructure for the delivery and

consumption of multimedia content – existing or under development – relate to each other.

The essence of all MPEG efforts is interoperability – interoperability for the consumers. Interoperability means that consumers can be sure to be able to use the content and not be bugged by incompatible formats, codecs, metadata, and so forth.

## **2.2 MPEG-4 Scene Features**

The MPEG-4 standard provides a large and rich set of technologies to satisfy the needs of authors, service providers and end users.

MPEG-4 achieves these goals by providing standardized ways to:

- (1) Represent units of aural, visual or audiovisual content, called “media objects”. These media objects can be of natural or synthetic origin.
- (2) Describe the composition of these objects to create compound media objects that form audiovisual scenes. The scene description builds on the Virtual Reality Modeling Language (VRML).
- (3) Multiplex and synchronize the data associated with media objects, so that they can be transported over network channels providing QoS appropriate for the nature of the specific media objects.

(4) Interact with the audiovisual scene generated at the receiver's end.

An MPEG-4 scene consists of a structured collection of independently coded interactive media objects (e.g. video, audio, graphics, and text), which are organized in a hierarchical fashion.

A media object in its coded form consists of descriptive elements that allow handling the object in an audiovisual scene as well as of associated streaming data, if needed. In its coded form, each media object can be represented independent of its surroundings or background.

Being described as objects an MPEG-4 scene can be adapted on the fly by adding, removing, or replacing any objects in the scene and the structure allows varying levels of granularity. The object-oriented nature of MPEG-4 makes it possible for end users to manipulate the media objects and create a multimedia presentation tailored to their specific needs, end device and connection limitations.

The MPEG-4 object-based representation approach where a scene is modeled as a composition of objects, both natural and synthetic, with which the user may interact, is at the heart of the MPEG-4 technology.

Figure 2 shows an example of an MPEG-4 scene composed of individual media objects.

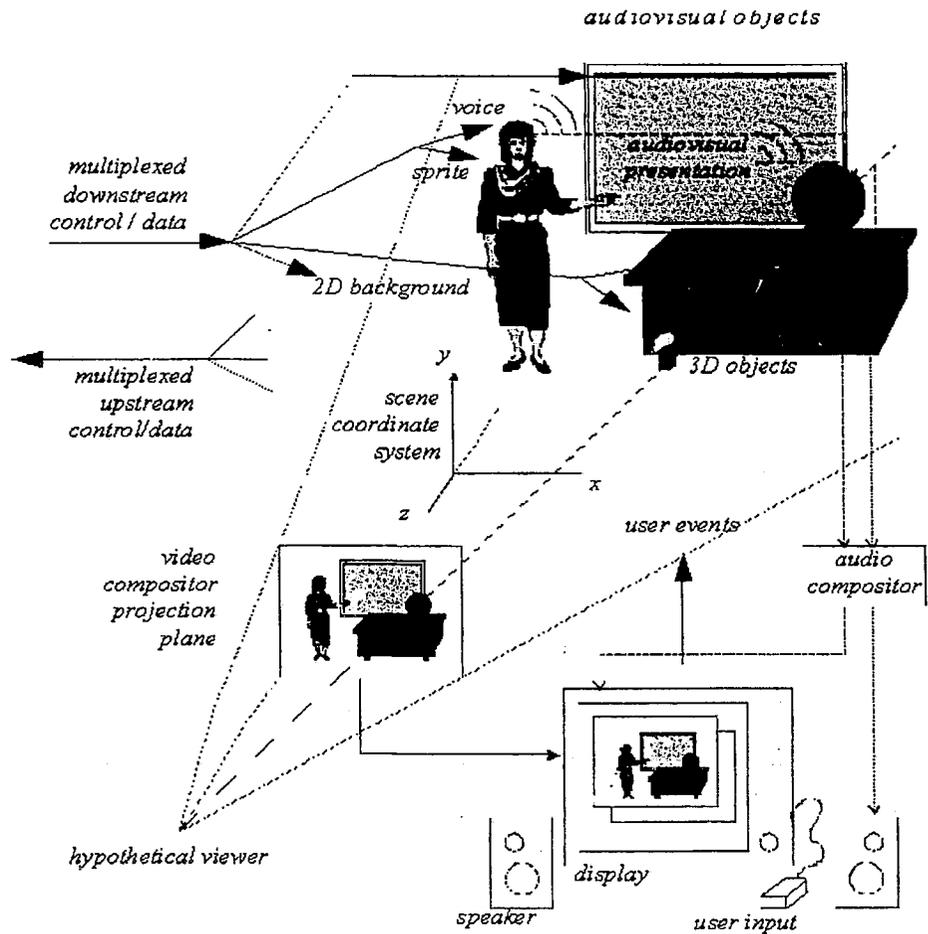


Figure 2 An Example of an MPEG-4 Scene

## 2.3 MPEG-4 Visual Coding Scheme

### 2.3.1 MPEG-4 frame types

MPEG-4 video is a sequence of three kinds of frames:

The **I-frames** are intra coded, i.e. they can be reconstructed without any reference to other frames. The **P-frames** are forward predicted from the last **I-frame** or **P-frame**, i.e. it is impossible to reconstruct them without the data of another frame (I or P). The **B-frames** are both forward predicted and backward predicted from the last/next I-frame or **P-frame**, i.e. there are two other frames necessary to reconstruct them. **P-frames** and **B-frames** are referred to as inter coded frames. Figure 3 shows a frame sequence example.

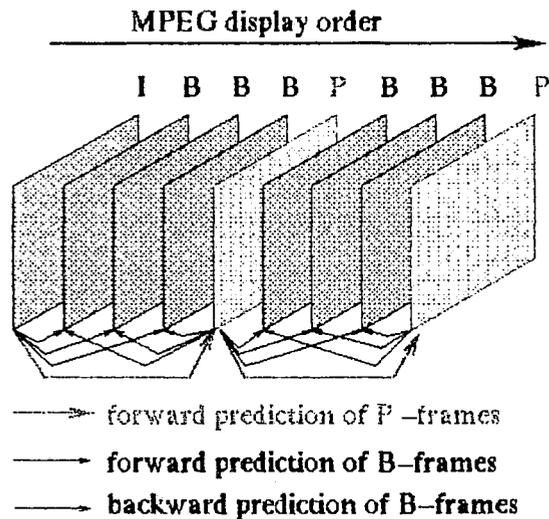


Figure 3 A frame sequence example

### 2.3.2 Layered scalable coding

Layered scalable coding as shown in Figure 4 is often proposed as a solution for rate control in video multicast application over the Internet.

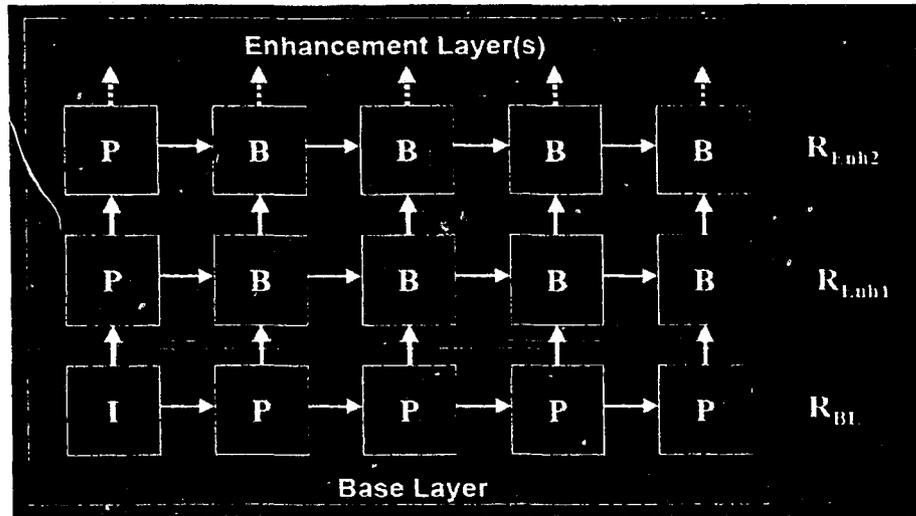


Figure 4 Layered Scalable Coding

A layered scalable video encoder compresses a raw video sequence into multiple sub-streams. One of the compressed sub-streams is the base layer stream, which can be independently decoded and provides coarse visual quality. Other compressed sub-streams are enhancement layer streams, which can only be decoded together with the base layer stream to provide better perceptual quality. Decoding the complete bit-streams provides the highest quality. Decoding the base layer stream and multiple enhancement layer streams produces video with degraded quality.

Video encoding scalability refers to the ability to decode only a part of a bit stream and reconstruct images or image sequences with:

- reduced decoder complexity and thus reduced quality

- reduced spatial resolution
- reduced temporal resolution
- with equal temporal and spatial resolution but with reduced quality

This functionality is desired for progressive coding of images and video sent over heterogeneous networks, as well as for applications where the receiver is not capable of displaying the full resolution or full quality images or video sequences. This could for instance happen when processing power or display resolution is limited.

### **2.3.3 Content Based Functionality**

The MPEG-4 image and video coding algorithms give an efficient representation of visual objects of arbitrary shape, also supporting so-called content-based functionalities.

Content-based functionality supports the separate encoding and decoding of content (i.e. physical objects in a scene, VOs). This MPEG-4 feature provides the most elementary mechanism for interactivity, flexible representation and manipulation with/of VO content of images or video in the compressed domain, without the need for further segmentation or transcoding at the receiver.

There are several scalable coding schemes in MPEG-4 visual: spatial scalability, temporal scalability, object-based spatial scalability, and fine granularity scalability.

Object-based spatial scalability extends the “conventional” types of scalability towards arbitrary shape objects, so that it can be used in conjunction with other object-oriented capabilities. Thus, a very flexible content-based scaling of video information can be achieved. This makes it possible to enhance SNR, spatial resolution, shape accuracy, etc, only for objects of interest or for a particular region, which can be done dynamically at playtime. This concept is illustrated in Figure 5.

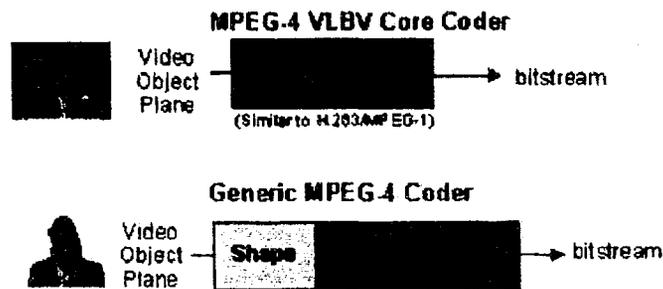


Figure 5 the VLBV Core and the Generic MPEG-4 Coder

### 2.3.4 Fine Granularity Scalability

For Internet streaming application, the channel capability is unpredictable and varies over a wide range. So it requires finer granularity than the layered representation.

A new scalable coding mechanism, called fine granularity scalability (FGS), was proposed to MPEG-4 [8, 9, 46]. An FGS encoder compresses a raw video sequence into two sub-streams: a base layer bit-stream and an enhancement bit-stream.

Similar to conventional scalable video coding, the base layer must be received completely in order to decode and display a basic quality video. In contrast to conventional scalable video coding, which requires the reception of complete enhancement layers to improve upon the basic video quality, with FGS coding the enhancement layer stream can be cut anywhere before transmission.

An FGS encoder uses bit-plane coding to represent the enhancement stream as shown in Figure 6. With bit-plane coding, an FGS encoder is capable of achieving continuous rate control for the enhancement stream (See Figure 7). This is because the enhancement bit-stream can be truncated anywhere to achieve the target bit-rate.

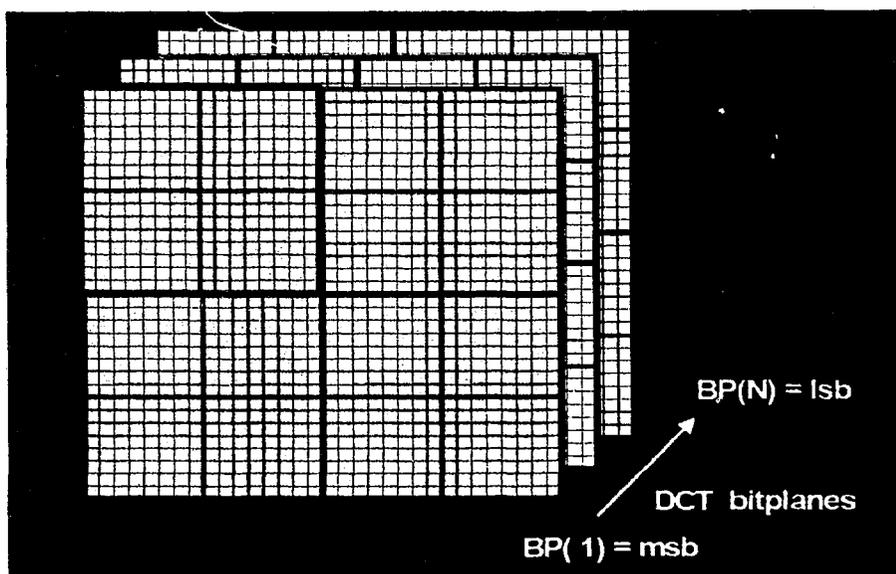


Figure 6 FGS Bitplane coding

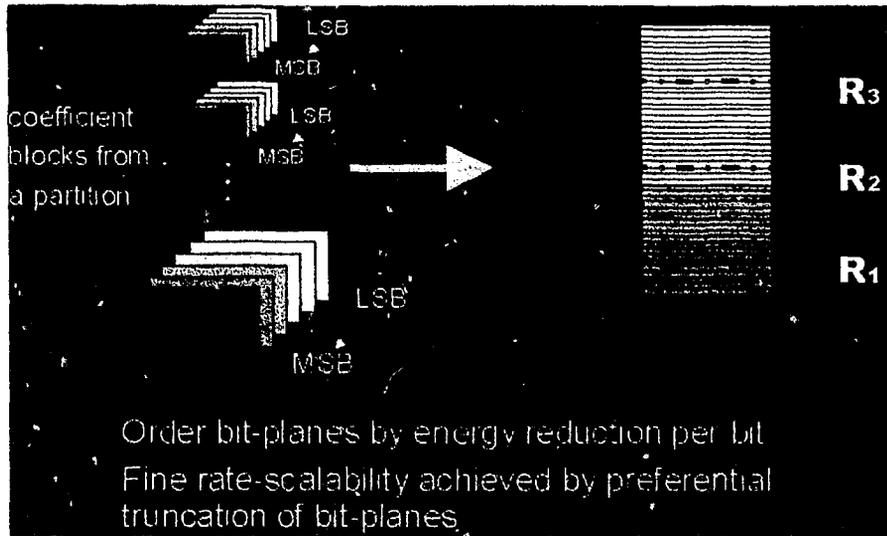


Figure 7 Archiving Rate Scalability

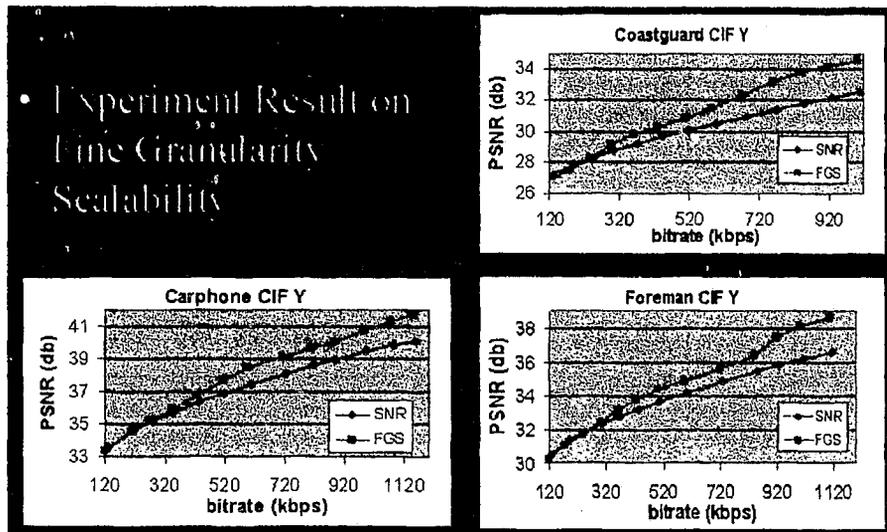


Figure 8 FGS Bitplane Coding Gain

Figure 8 illustrated the FGS Bitplane coding PSNR gain compared with SNR scalable coding.

The received part of the FGS enhancement layer stream can be successfully decoded and improves upon the basic video quality. Similar to conventional scalable encoding, the FGS enhancement layer is hierarchical in that “higher” bits require the “lower” bits for successful decoding. This means that when cutting the enhancement layer bit stream before transmission, the lower part of the bit stream (below the cut) needs to be transmitted and the higher part (above the cut) can be dropped. The FGS enhancement layer can be cut at the granularity of bits. This fine granular flexibility was the key design objective of FGS coding, along with good rate–distortion coding performance. With the fine granularity property, FGS–encoded videos can flexibly adapt to changes in the available bandwidth in wired and wireless networks. This flexibility can be exploited by video servers to adapt the streamed video to the available bandwidth in real–time (without requiring any computationally demanding re–encoding). In addition, the fine granularity property can be exploited by intermediate network nodes (including base stations in wireless networks) to adapt the video stream to the currently available downstream bandwidth.

A variation of FGS is progressive fine granularity scalability (PFGS). PFGS shares the good features of FGS, such as fine granularity bit–rate scalability and error resilience. 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.

The fine grain scalability (FGS) of MPEG-4 is one big step towards the scalable video solution where the base layer targets at providing the basic visual quality to meet the minimal user bandwidth, while the scalable enhancement layer can be arbitrarily truncated to meet heterogeneous network conditions (See Figure 9).

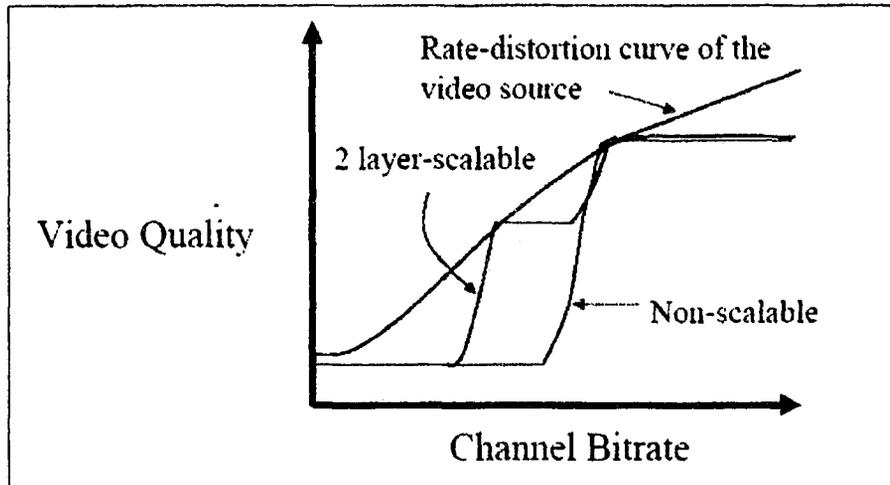


Figure 9 Performance of FGS vs. layered scalability

With the help of MPEG-4 FGS, video streaming is much simplified. However, scalable coding alone is not sufficient to achieve high inter-receiver fairness and high video quality. To improve the inter-receiver fairness in heterogeneous networks and

improve the quality of video obtained by each of the receivers, a flexible delivery technique is critical.

## **2.4 MPEG-4 Transportation**

MPEG-4 is transport-agnostic, and designed that way on purpose. This means that MPEG-4 content can be carried over many different transport layers, and move from one transport to the other. There are some cases, however, where MPEG did some work relating to the transport of MPEG-4 content, but adaptation to a specific transport layer is needed.

### **2.4.1 Delivery Multimedia Integration Framework (DMIF)**

DMIF [2] is a session protocol for the management of multimedia streaming over generic delivery technologies. The functionality provided by DMIF is expressed by an interface called DMIF-Application Interface (DAI), and translated into protocol messages.

The DMIF architecture is such that applications that rely on DMIF for communication do not have to be concerned with the underlying communication method. The implementation of DMIF takes care of the delivery technology details presenting a simple interface to the application.

The synchronized delivery of streaming information from source to destination, exploiting different QoS as available from the network, is specified in terms of the synchronization layer and a

delivery layer containing a two-layer multiplexer, as depicted in Figure 10.

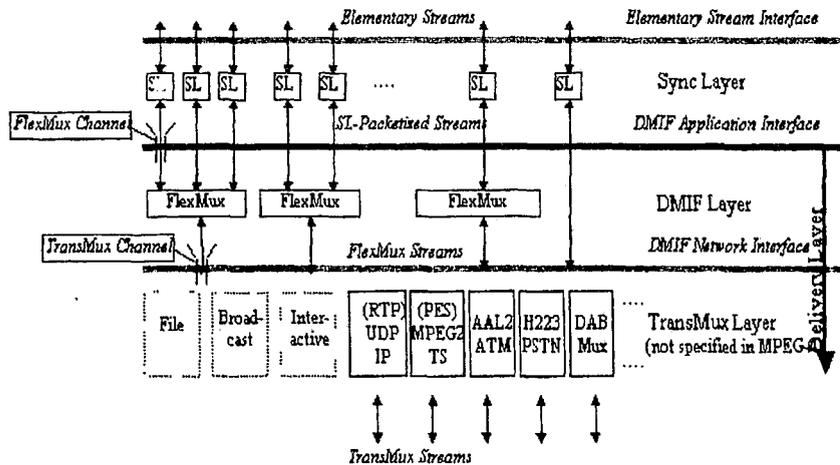


Figure 10 The MPEG-4 System Layer Model

## 2.4.2 MPEG-4 over IP

MPEG-4 provides a framework for the carriage of MPEG-4 contents over IP networks and guidelines for designing payload format specifications for the detailed mapping of MPEG-4 content into several IP-based protocols.

Real-Time Transport Protocol (RTP) is the most common protocol for transporting real-time streaming data over the Internet. To assure compatibility between different RTP payload formats, the framework defines a conformance point as illustrated in the Figure 11. To conform this framework all the payload formats shall provide normative mapping functions to reconstruct logical MPEG-4 SL

packets. The framework also defines the standard MIME types associated with MPEG-4 contents.

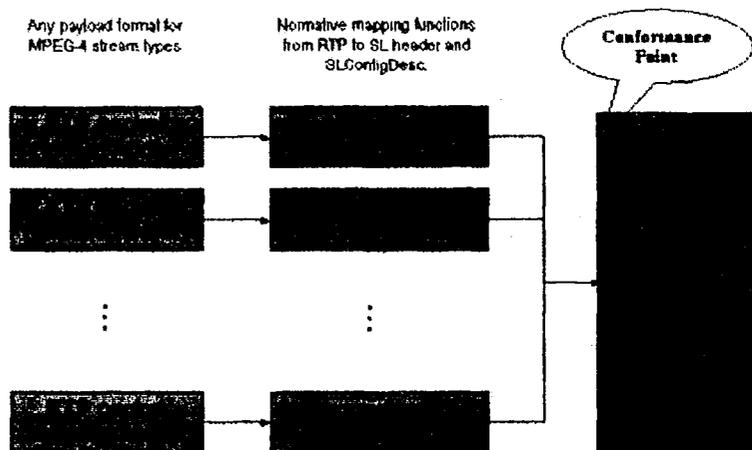


Figure 11 RTP packet to logical SL packet mapping

Several RTP payload formats are developed under this framework including generic payload format and FlexMux payload format [7]. Generic RTP payload format specify a homogeneous carriage of various MPEG-4 streams. It defines a simple but efficient mapping between logical MPEG-4 SL packets and RTP packets. It also supports concatenation of multiple SL packets into one RTP packets to minimize overheads. FlexMux payload format specifies a carriage of FlexMux packetized streams via RTP packets. It includes a payload formats to convey FlexMux descriptors to dynamically signal the configuration of FlexMux.

## **Chapter Three**

### **3. MPEG-4 FGS VIDEO TRANSPORT OVER INTERNET**

#### **3.1 RTP/RTCP Introduction**

The Internet standard for transporting real-time data such as audio and video is the Real-Time Transport Protocol (RTP).

RTP provides end-to-end network transport functions suitable for applications transmitting data with real-time characteristics over multicast or unicast network services. However, RTP does not address resource reservation and does not guarantee quality-of-service for real-time services. It also neither guarantees delivery nor prevents out-of-order delivery. It does not assume that the underlying network is reliable and delivers packets in sequence. It is up to the receiver to reconstruct the sender's packet sequence and detect lost packets using the information provided in the packet header.

It is augmented by a RTP control protocol (RTCP) to allow monitoring of the data delivery in a manner scalable to large multicast networks, and to provide minimal control and identification functionality. RTCP is a companion protocol of RTP:

- (1) RTP is for carrying data that has real-time properties.

(2) RTCP is for monitoring the quality of service and conveying information about the participants in an on-going session.

If quality of service is essential for a particular application, RTP can also be used over a resource reservation protocol such as RSVP.

RTP and RTCP are designed to be independent of the underlying transport and network layers. However, TCP is a transport-layer protocol designed for reliable data communications, and its overhead of guaranteeing reliable data transfer may slow the overall transmission rate and introduce delays. For this reason, streaming media applications typically run RTP on top of UDP to make use of its multiplexing and checksum services; both protocols contribute parts of the transport protocol functionality. Figure 12 shows the RTP/RTCP/UDP/IP stack for MPEG-4 streaming over Internet.

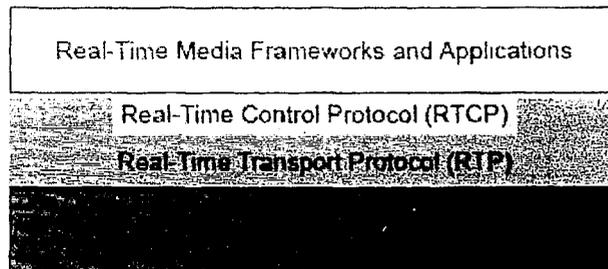


Figure 12 RTCP/RTP/UDP/IP Stack

### 3.2 RTP/RTCP functionalities

RTP supports data transfer to multiple destinations using multicast distribution if provided by the underlying network. RTP provides the following functions for streaming applications:

**Time-stamping.** RTP provides time stamping for applications to synchronize different media streams.

**Sequence numbering.** Since packets arriving at the receiver may be out of sequence (UDP does not deliver packets in sequence), RTP employs sequence numbering for application to determine what order the packets of data should be presented. The sequence number is also used for packet loss detection.

**Payload type identification.** The type of the payload contained in an RTP packet is indicated by an RTP-header field called payload type identifier. The receiver interprets the content of the packet based on the payload type identifier.

**Source identification.** The source of each RTP packet is identified by an RTP header field called Synchronization Source identifier (SSRC), which provides a mean for the receiver to distinguish different sources.

All these real-time services are implemented through the RTP header. Each chunk of payload data is preceded by an RTP header. The RTP header contains timing information and a sequence number that allow the receivers to reconstruct the timing produced by the source (See Figure 13).

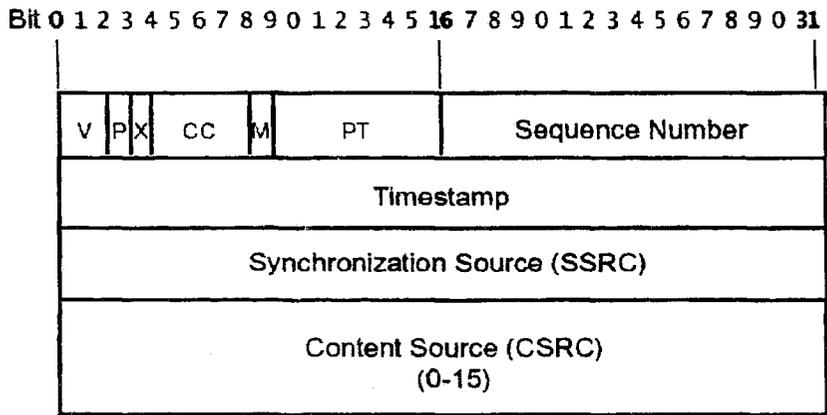


Figure 13 RTP Header

In an RTP session, participants periodically send RTCP packets to convey feedback on quality of data delivery and information of membership.

RTCP provides the following services:

**QoS feedback.** This is the primary function of RTCP. RTCP provides feedback to an application regarding the quality of data distribution. The feedback is performed in the form of sender reports (sent by the source) and receiver reports (sent by the receiver). The reports can contain information on the quality of reception such as (1) fraction of the lost RTP packets since the last report, (2) cumulative number of lost packets since the beginning of reception, (3) packet inter-arrival jitter, and (4) delay since receiving the last sender's report.

The feedbacks are useful for control of adaptive encodings as well as diagnose of the distribution faults. Based on the feedback, the sender can adjust its transmission rate; the receivers can determine their subscribing rates; network managers can evaluate the network performance for multicast distribution.

**Participant identification.** RTCP carries a persistent transport-level identifier for an RTP source called the canonical name or CNAME. The SSRC field in the RTP header can identify a source. However, the SSRC identifier may change if a conflict is discovered or a program is restarted, receivers require the CNAME to keep track of each participant. Receivers may also require the CNAME to associate multiple data streams from a given participant in a set of related RTP sessions, for example to synchronize audio and video. Inter-media synchronization also requires the NTP and RTP timestamps included in RTCP packets by data senders.

Furthermore, the SSRC identifier is not convenient for human users. RTCP provides a human-friendly mechanism for source identification. Specifically, RTCP SDES (source description) packets contain textual information called canonical names as globally unique identifiers of the session participants. It may include a user's name, telephone number, email address and other information.

**Control packets scaling.** The rate for control packets transmission must be controlled for RTP to scale up to a large number of participants. A control mechanism is designed as

follows. The control mechanism keeps the total control packets to 5% of the total session bandwidth. Among the control packets, 25% are allocated to the sender reports and 75% to the receiver reports.

**Inter-media synchronization.** RTCP sender reports contain an indication of real time and the corresponding RTP timestamp. This can be used in inter-media synchronization like lip synchronization in video.

**Minimal session control information.** This optional functionality can be used for transporting session information such as names of the participants. This is most likely to be useful in "loosely controlled" sessions where participants enter and leave without membership control or parameter negotiation. To support all the control communication requirements of an application, a higher-level session control protocol may be needed.

### **3.3 Sender and Receiver Reports Analysis**

RTP receivers provide reception quality feedback using RTCP report packets. There are two forms of reports: the Sender Report (SR) and the Receiver Report (RR). The SR is issued if a site has sent any data packets during the interval since issuing the last report or the previous one, otherwise the RR is issued. The only difference between the sender report and receiver report forms, besides the packet type code, is that the sender report includes a 20-byte sender information section for use by active senders. The RR RTCP packet format is shown in Figure 14.

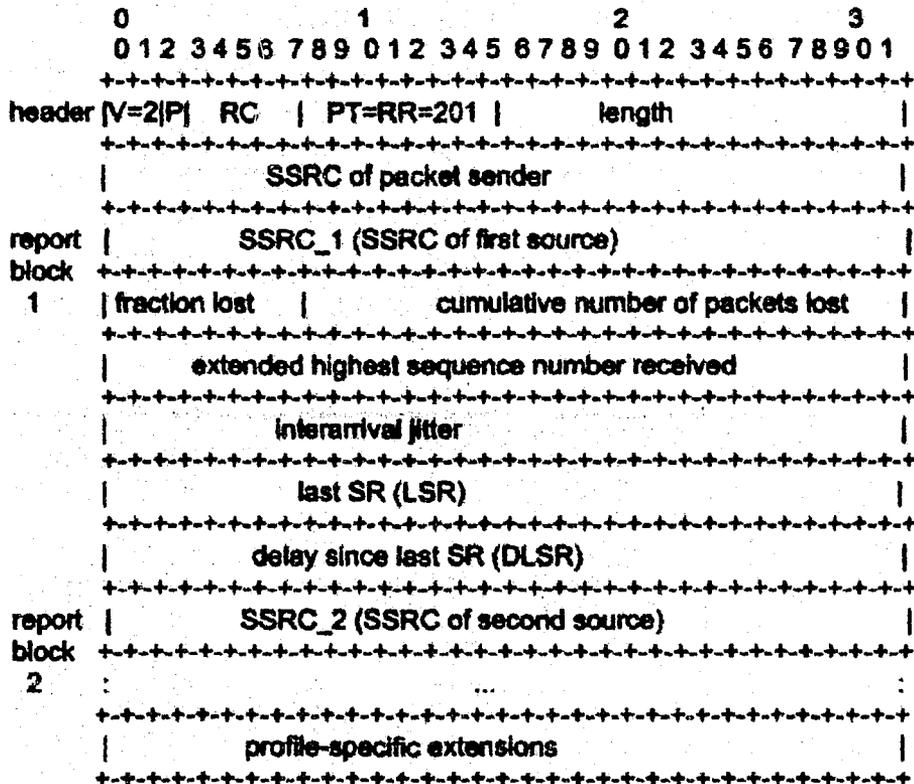


Figure 14 Receiver report RTCP packet format

Based on the feedbacks, the sender may modify its transmissions; the receivers can determine their subscribing rates. Network managers may use profile-independent monitors that receive only the RTCP packets and not the corresponding RTP data packets to evaluate the performance of their networks for multicast distribution.

Cumulative counts are used in both the sender report and receiver report blocks so that differences may be calculated between any two reports to make measurements over both short and long time periods, and to provide resilience against the loss of a report. The difference between the last two reports received can be used to estimate the recent quality of the distribution. The NTP timestamp is included so that rates may be calculated from these differences over the interval between two reports. Since that timestamp is independent of the clock rate for the data encoding, it is possible to implement encoding- and profile-independent quality monitors.

An example calculation is the packet loss rate over the interval between two receiver reports. The difference in the cumulative number of packets lost gives the number of packets lost during that interval. The difference in the extended last sequence numbers received gives the number of packets expected during the interval. The ratio of these two is the packet loss fraction over the interval. This ratio should equal the fraction lost field if the two reports are consecutive, but otherwise it may not. The loss rate per second can be obtained by dividing the loss fraction by the difference in NTP timestamps, expressed in seconds. The number of packets received is the number of packets expected minus the number lost. The number of packets expected may also be used to judge the statistical validity of any loss estimates.

In addition to the cumulative counts that allow long-term packet loss measurements using differences between reports, the fraction lost field provides a short-term measurement from a single report.

This becomes more important as the size of a session scales up enough that reception state information might not be kept for all receivers or the interval between reports becomes long enough that only one report might have been received from a particular receiver.

The inter-arrival jitter field provides a second short-term measure of network congestion. Packet loss tracks persistent congestion while the jitter measure tracks transient congestion. The jitter measure may indicate congestion before it leads to packet loss. The inter-arrival jitter field is only a snapshot of the jitter at the time of a report and is not intended to be taken quantitatively.

Rather, it is intended for comparison across a number of reports from one receiver over time or from multiple receivers, e.g., within a single network, at the same time. To allow comparison across receivers, it is important the jitter be calculated according to the same formula by all receivers.

Because the jitter calculation is based on the RTP timestamp which represents the instant when the first data in the packet was sampled, any variation in the delay between that sampling instant and the time the packet is transmitted will affect the resulting jitter that is calculated. Such a variation in delay would occur for audio packets of varying duration. It will also occur for video encodings because the timestamp is the same for all the packets of one frame but those packets are not all transmitted at the same time. The variation in delay until transmission does reduce the accuracy of the jitter calculation as a measure of the behavior of the network by

itself, but it is appropriate to include considering that the receiver buffer must accommodate it. When the jitter calculation is used as a comparative measure, the (constant) component due to variation in delay until transmission subtracts out so that a change in the network jitter component can then be observed unless it is relatively small. If the change is small, then it is likely to be inconsequential.

# Chapter Four

## 4. RATE ADAPTATION FRAMEWORK

### 4.1 Background

Packets loss and delay have devastating effect on video perceptual quality. So, real-time media streaming applications should be able to adjust the transmission rate to match the capacity of the receiver or to adapt to network congestion. In multicast environment, different receivers are connected to the source via paths with varying bandwidth, delays, and packets loss characteristics. For example, some clients use 56k modems that have only 36Kbps payload, ISDN clients usually have 128Kbps, DSL or cable modem clients may have 256Kbps to 1Mbps bandwidth that depends on the distance of the client location. Excessively high bit rate transmission would result in congestions to the users with lower bandwidth, while conservatively low bit rates would waste most users' bandwidth. In either case, the receivers suffer from performance degradation and experience inter-receiver unfairness [27].

In this thesis, the inter-receiver fairness is defined as each receiver can adaptively get close to the best affordable video quality that matches the available bandwidth on its path and the process capabilities of its terminal. The proposed rate adaptation framework attempts to achieve the inter-receiver fairness.

## **4.2 Rate Adaptation Mechanisms**

Rate adaptation is typically used for video streaming applications for congestion control. Rate adaptation attempts to minimize the possibility of network congestion by matching the transmission rate to the network conditions and the terminal process capabilities, and thus to prevent applications from overloading the network and avoid excessive packets loss and delay.

### **4.2.1 Network-based vs. End-to-end**

The rate adaptation schemes can be classified according to how the feedback information is generated and delivered. If it is generated by the end receivers, the scheme is called end-to-end. If it is generated by the switching units or routers on the network, the scheme is named as network-based scheme.

Network-based rate adaptation requires that routers and switches monitor their available bandwidth and perform congestion control. This kind of approach usually requires changing the network service model and cannot be implemented in every environment.

The end-to-end rate adaptations are performed solely at the end systems (sender and receivers). They can be employed by the end systems and do not require special architectural change for QoS support from the network. The end-to-end approach transfers the congestion control responsibilities to the end systems.

Compared with the network-based approach, the end-to-end approaches are easy to deploy in the Internet. However, there is a difficulty in catching up with fast variations of congestion status in the network, so the rate adaptive reaction is slow and imprecise.

#### **4.2.2 Sender-adaptive vs. Receiver-driven**

Rate adaptation to network congestion may be sender-adaptive or receiver-driven.

##### **4.2.2.1 Sender-adaptive approach**

In a sender-adaptive algorithm, the source adapts the transmission rate in response to congestion feedback from the network or the receivers.

Vickers et al proposed two Source-Adaptive Multi-layered Multicast (SAMM) algorithms, which use congestion feedback to adjust the transmission rates of multiple layers of data [13].

One is a network-based SAMM algorithm, in which it is assumed that network switches are capable of executing complex flow and congestion control algorithms. However, in most existing networks, where datagram routing and forwarding are often the only universally shared operations, the existence of special congestion control functions cannot be assumed.

Another is an end-to-end SAMM algorithm. In this algorithm, video receivers generate congestion feedback to the sender by monitoring the arrival rate of video traffic, and feedback packets are merged by an overlaid virtual network of feedback merging

servers. The prerequisites for its implementation include router-based priority packet discarding and flow isolation.

#### **4.2.2.2 Receiver-driven approach**

S. McCanne, V. Jacobson, and M. Vetterli proposed a receiver-driven adaptation algorithm for the multicast of layered video [17]. In this algorithm, known as Receiver-driven Layered Multicast (RLM), the video source generates a fixed number of layers, each at a fixed rate, and the receivers "subscribe" to as many layers as they have the bandwidth to receive. Congestion is monitored at the receivers by observing packet losses. This approach has the advantage that it uses video layering to address heterogeneous bandwidth constraints. However, it limits the receivers to choosing among the layers the source is willing to provide, and in many cases the provided selection may not be adequate to improve network utilization and video quality. Furthermore, RLM is relatively slow to adapt to changes in the network's available bandwidth due to the latency of the control mechanism named "*joint-experiments*". If the background traffic is particularly bursty, the receivers may not be able to adapt appropriately, resulting in degraded utilization and video quality.

### **4.3 The Proposed Framework**

The stability of the Internet to date has in large part been due to the congestion control and avoidance algorithms implemented in its dominant transport protocol TCP [39, 40, 41, 42].

However, the TCP congestion control algorithms such as additive-increase/multiplicative-decrease (AIMD) are not well suited for video streaming applications. The user's perceived quality would drop drastically when the throughput decreases multiplicatively.

A flexible rate adaptation framework plays a key role in real-time video streaming over the Internet. After examined the features and functionalities of MPEG-4 FGS video and RTP/RTCP, reviewed the related research works, an end-to-end hybrid rate adaptation framework was proposed in this paper. Figure 15 illustrates a scenario of end-to-end hybrid rate adaptation in heterogeneous streaming over IP environment. The framework has taken advantage of MPEG-4 FGS and sender-adaptive/receiver-driven hybrid rate adaptation to improve the overall QoS for MPEG-4 FGS video streaming over the Internet. It can deal with both unicast and multicast.

For multicasting, it is assumed that intermediate network nodes filter and drop packets at out going interfaces so that downstream receivers will receive packet at their subscribing rates. Because the scalable enhancement layer of FGS video stream can be arbitrarily truncated, there is no need for transcoding in intermediate network nodes in order to switch the received high rate stream to a lower out going rate stream when network congestion occurs.

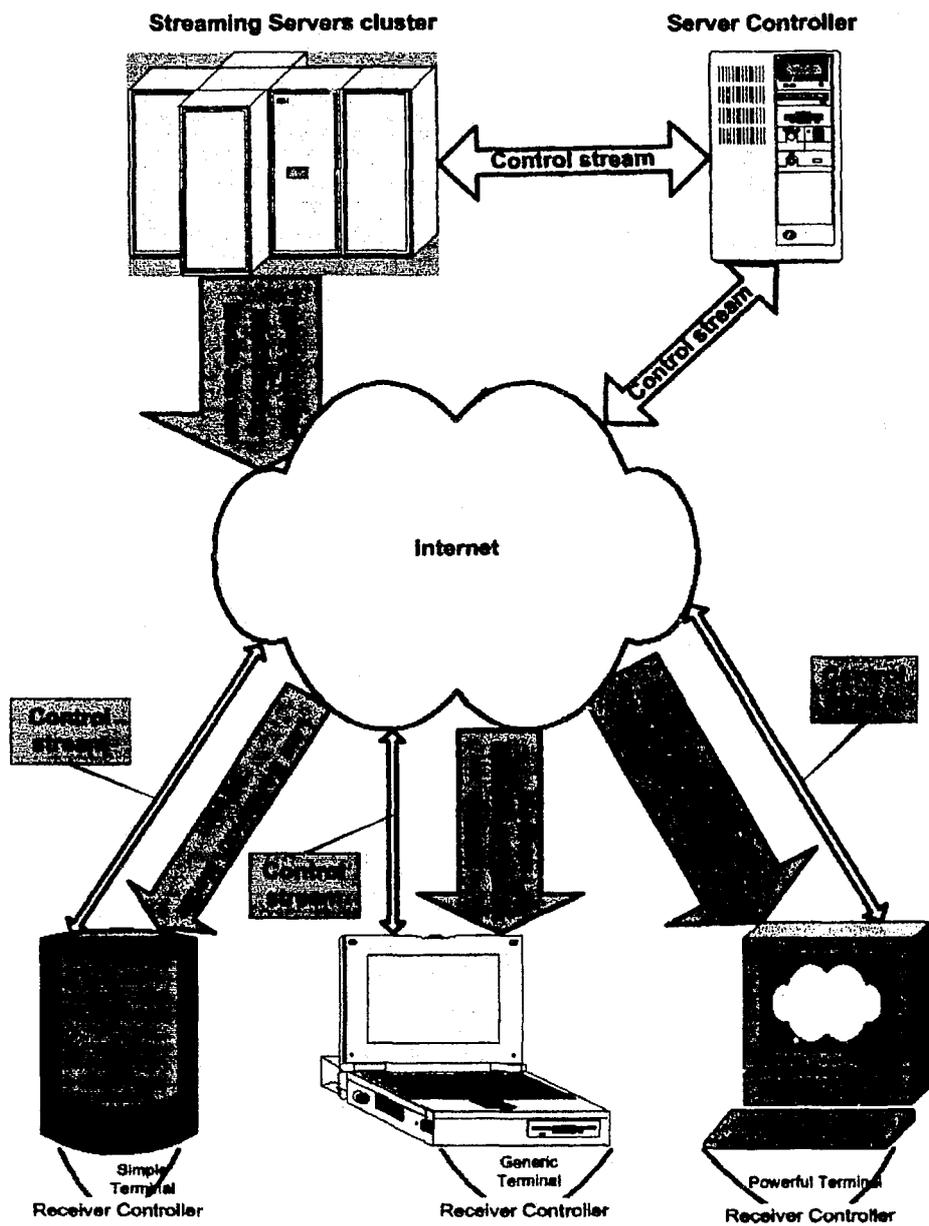


Figure 15 A scenario for end-to-end hybrid rate adaptation

### **4.3.1 MPEG-4 FGS BL and EL**

The MPEG-4 FGS video consists of one non-scalable coded base layer (BL) and bitplane-encoded scalable enhanced layer (EL) as shown in Figure 16. The base layer targets at providing the basic perceptual quality to meet the minimal user bandwidth, and encoded by constant-bit-rate (CBR) rate control. The enhancement layer can be arbitrarily truncated to meet heterogeneous network conditions. MPEG-4 FGS originally assumes guaranteed delivery of the BL and leaving EL to the mercy of the best-effort Internet.

Because relatively small bandwidth (as low as 5kbps) is required for the BL and the BL packets are very important, the BL stream should be guaranteed. The base layer of MPEG4 FGS is fully compatible with the MPEG4 non-scalable coding scheme. A lot of error resilience tools are being standardized in MPEG4 standard. Typically, packet-loss concealment for BL is copy data from previous frame.

Packet-loss within EL does not propagate. When packet-loss occurs in EL, the receiver can just discard remaining less significant bit-planes. No error concealment in FGS enhancement layer.

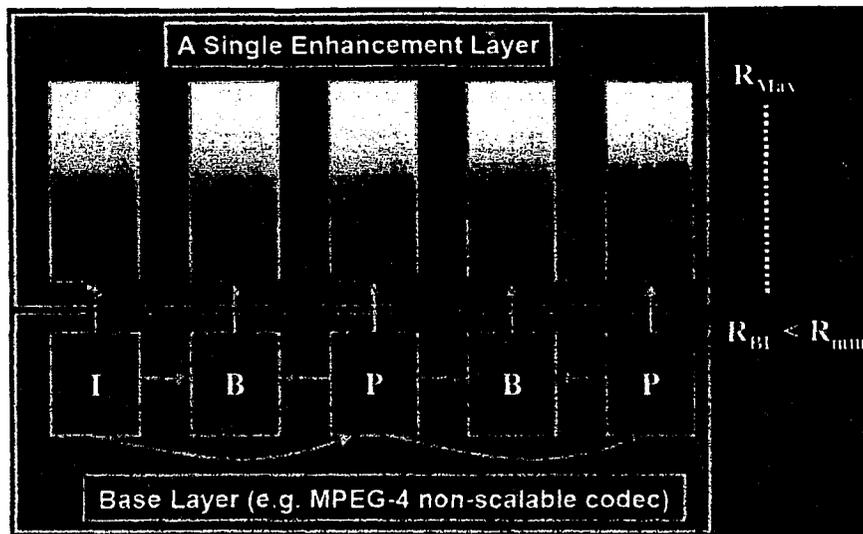


Figure 16 FGS Coding

#### 4.3.2 Rate Controllers

TCP is dominant in today's Internet and its end-to-end congestion control mechanisms [41, 42, 43, 54, 55, 56] are crucial to the robustness of the Internet. Since a TCP flow reduces its sending rate on detection of congestion, flows without appropriate congestion control mechanism can obtain larger share of bandwidth on congested links and will possibly lead to 'congestion collapse' in the network. TCP friendliness is a fairness criterion to guide behaviors of non-TCP based best effort traffics and to prevent them from starving TCP flows. A flow said to be TCP friendly if its long-term throughput does not exceeds the throughput of a conformant TCP connection under the same circumstances.

To be safe for the deployment of video streaming applications in the Internet, the rate controllers must implement "TCP-compatible" congestion control algorithms, which interact well with TCP and maintain the stability of the Internet [42, 43]. The idea is to ensure that the TCP connections get their fair allocation of bandwidth in the presence of these non-TCP protocols and vice versa.

The hybrid rate adaptation framework is based on RTCP reports. It takes the packet loss ratio measured at the receivers as the control indicator. The rate controllers work at both the receivers' side and the sender's side. The rate controllers can be configured with different rate control schemes via changing the controller parameters. These parameters should be adjusted for the specific applications and video sources.

Rate controllers perform:

- **RTCP analysis.** The receiver reports of all receivers are analyzed and statistics of packet loss, packet delay jitter and roundtrip-time are computed.
- **Network state estimation.** The actual network congestion state seen by every receiver is determined as unloaded, hold, or congested. This is used to decide whether to increase, hold or decrease both the subscribing rates of the receivers and the transmission rate of the sender.
- **Bandwidth adjustment.** The bandwidth of the multimedia application is adjusted according to the decision of the network state analysis. The end systems can set the rate parameters for adjusting bandwidth.

#### 4.3.2.1 Rate controllers initialization

At the beginning of a multicast session, the sender announces the rate range  $R_{min}$  and  $R_{max}$  suitable for the source. The values can be estimated from the average rate-distortion curve of the source.  $R_{min}$  should be at least a little bit greater than the rate of the BL  $R_{BL}$ .  $R_{max}$  is the rate above which there is no significant improvement of the video quality.

When a receiver join the multicasting session, it will set an initial Receive Rate ( $RR_{init}$ ) which roughly match its network condition and within the range of  $R_{min}$  and  $R_{max}$ . Then the receiver controller will monitor the transmission state.

By default, the initial sending Rate ( $R_{init}$ ) is set to the larger value between  $R_{min}$  and 80% of  $R_{max}$ , i.e.  $R_{init} = \max\{R_{min}, 0.8 R_{max}\}$ . However it can be configured according to the source and session features. The sender rate maybe adjust according the status feedback from the receivers during the session.

The increase step rate ( $R_{is}$ ) and decrease step rate ( $R_{ds}$ ) used in the control action also needs to be set during the session initialization. For example,  $R_{is} = 0.1 R_{min}$ ;  $R_{ds} = 1.5 R_{is}$

#### 4.3.2.2 Packet-Loss Ratio (PLR) Analysis

RTCP provides an algorithm to calculate the packet-loss ratio at the receiver. This property can be retrieved from RTCP receiver report (RR).

In order to avoid QoS oscillations, a low-pass filter is adopted to compute the average PLR, i.e.  $PLR = (1 - \alpha) PLR_{pre} + \alpha PLR_C$ , where  $PLR_C$  is the current packet loss ratio,  $PLR_{pre}$  is the previous average packet loss ratio, and  $1 \geq \alpha \geq 0$ . Increasing  $\alpha$  means increasing the influence of the current packet loss ratio.

In this framework, two packet loss ratio thresholds are defined. One is the tolerable packet loss ratio, called  $PLR_0$ . At the  $PLR_0$  packet loss ratio level, the perceptual quality at the application level will not be degraded.

The reason for this definition is to improve network bandwidth utilization. As described in RFC 1757, a reasonable estimate of Ethernet bandwidth utilization is the ratio of the total number of octets of data (including those in bad packets) received on the network to the time interval [65].  $PLR < PLR_0$  indicates there is room for improving network bandwidth utilization.

Another is the unacceptable packet loss ratio called  $PLR_1$ . At the level of packet loss ratio  $PLR_1$ , the perceptual quality is significantly degraded and it is unacceptable to end-users.

So, the desired packet loss ratio for receivers is in the range between  $PLR_0$  and  $PLR_1$  ( $PLR_1 > PLR > PLR_0$ ).

Both  $PLR_0$  and  $PLR_1$  can be chosen empirical data and justified by experimental results.

#### 4.3.2.3 Receivers subscribe rate control and algorithm

The receivers generally include sub-modules such as FGS decoder, stream buffer, packet buffer, video display, retransmission monitor, and client controller. Figure 17 illustrates the architecture of the video receivers.

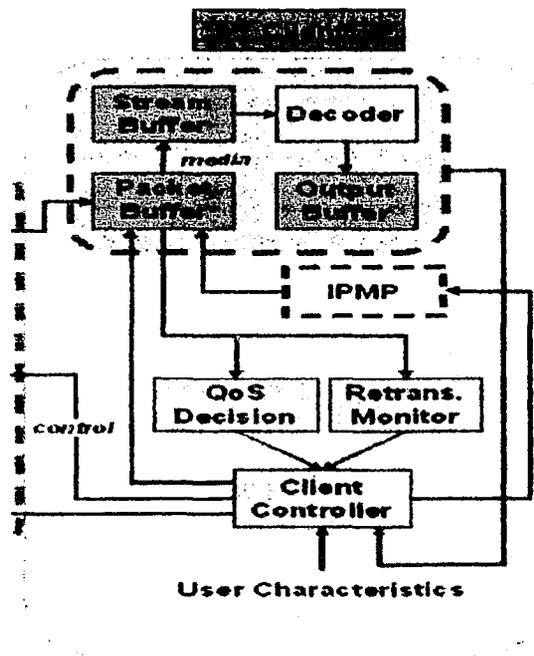


Figure 17 Video Receiver Architecture

The subscribing rate control mechanism is implemented in client controller. The client controller adjusts the subscribing rate RR for each receiver according to the following algorithm:

```
IF ( PLR < PLR0 )
  RR = MIN ( RR + Ris, Rmax );
ELSE IF ( PLR > PLRr )
  RR = MAX ( RR - Rds, Rmin );
ELSE
  RR remains no change.
```

There are three states for each receiver: HOLD, UNLOADED, and CONGESTED.

In "HOLD" state, the receiver rate does not change. In "UNLOADED" state, the receiver rate increases step by step, at the pace of  $R_{is}$ , until the receiver's load status reaches "HOLD". In "CONGESTED" state, the receiver drops rate step by step, at the pace of  $R_{ds}$ , to avoid congestion and reach the "HOLD" load state. The states transition diagram shows as Figure 18. The receivers send the load state to the sender via receive report RTCP packet.

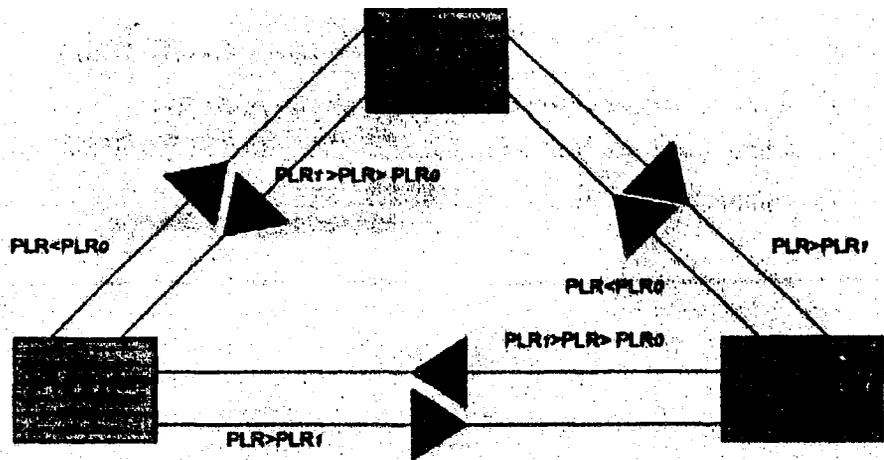


Figure 18 The receiver states transition

With the rate adaptation, the receiver can adjust its subscribing rate match the available bandwidth on its path, and this leads to the inter-receiver fairness in a multicast session. Otherwise, if the subscribing rate is higher than the available bandwidth on its path, network congestion will occur. If the subscribing rate is lower than the available bandwidth on its path, the bandwidth is not fully utilized. In either case, the receiver suffers from QoS degradation.

A pattern of the received MPEG-4 FGS streaming with truncating the enhancement layer at the decoder is shown in Figure 19.

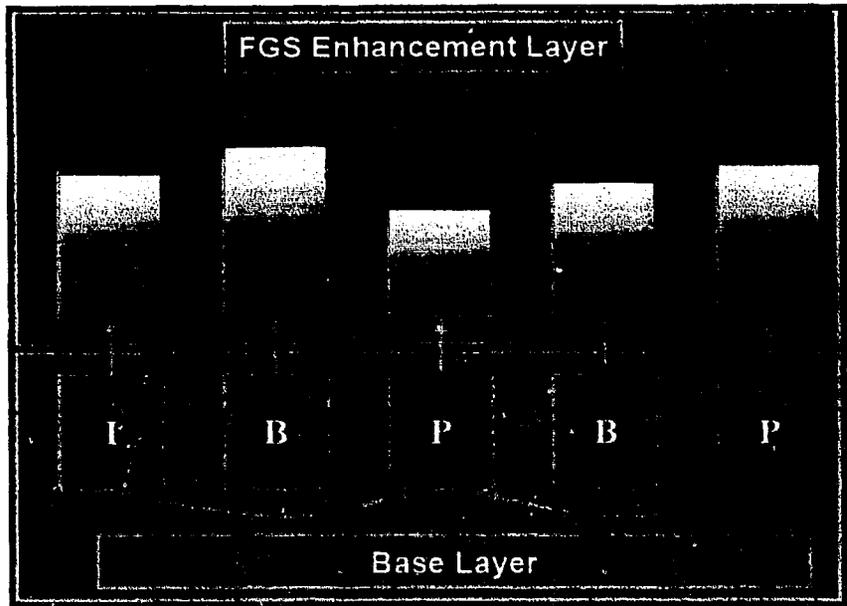


Figure 19 Received MPEG-4 FGS streaming

#### 4.3.2.4 Sender rate control and algorithm

The sender maintains the information of session members. The sender adjusts the sending rate  $R$  according to the proportion of load status feedback from the receivers. The sender updates the state of the receivers when it receives a RR RTCP packet from the receiver. Let  $N$  stands for the number of session members,  $N_H$ ,  $N_C$ , and  $N_U$  stand for the number of the members whose state are HOLD, CONGESTED, and UNLOADED respectively. All members have the same service priorities are assumed here. The algorithm is designed as: only if the ratio  $N_C/N$ , or  $N_U/N$  is greater or equals 50%, then the sender will adjust its sending rate  $R$  according to the following algorithm. In unicast case,  $N=1$ .

```
IF (  $N_u/N \geq 50\%$  )  
  R = MIN ( R+ $R_u$ ,  $R_{max}$  );  
  
ELSE IF (  $N_d/N \geq 50\%$  )  
  R = MAX ( R- $R_d$ ,  $R_{min}$  );  
  
ELSE  
  R remains no change.
```

This algorithm allows a certain fraction of the receivers to be in a congested state and avoids frequently adjusting the sender's transmission rate according to a small fraction of receivers. Since with MPEG-4 FGS encoding techniques, the intermediate network nodes can easily truncate the downstream rate matching the receivers' subscribing rate without transcoding, each receiver will estimate the available bandwidth on the path and adapt the subscribing rate promptly. In this case, a receiver or a small proportion of receivers connected via a low bandwidth link cannot force the sender to provide low quality video to other receivers which have high bandwidth connections, and vice versa. Hence, the inter-receiver fairness is achieved.

The architecture of the sender in the framework is shown as Figure 20.

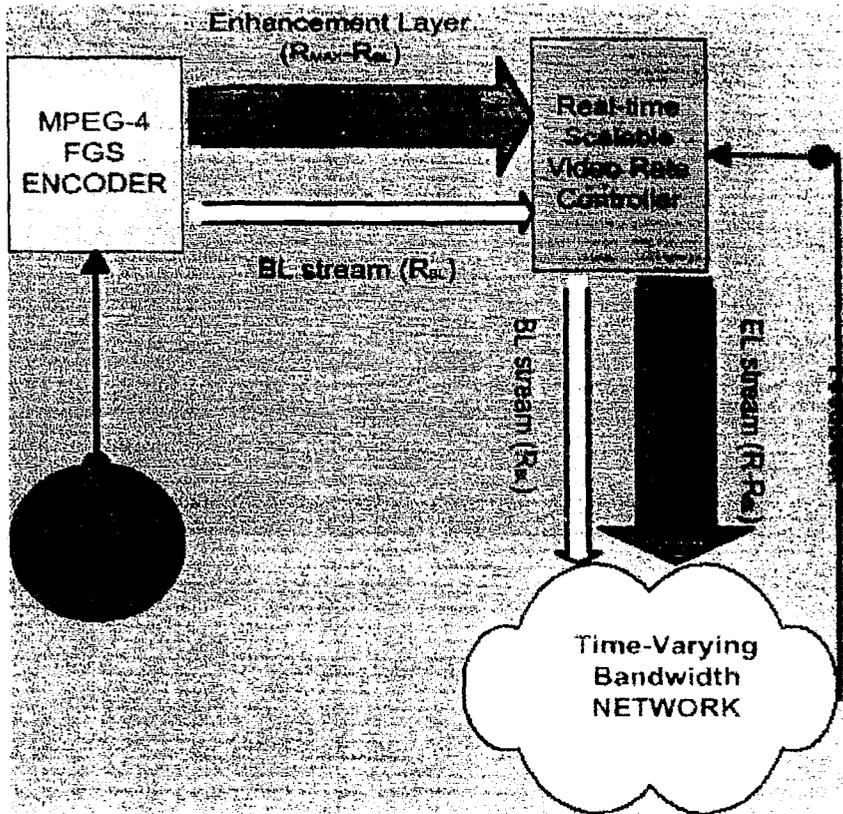


Figure 20 The sender in the framework

A pattern of the MPEG-4 FGS streaming at the sender is shown in Figure 21.

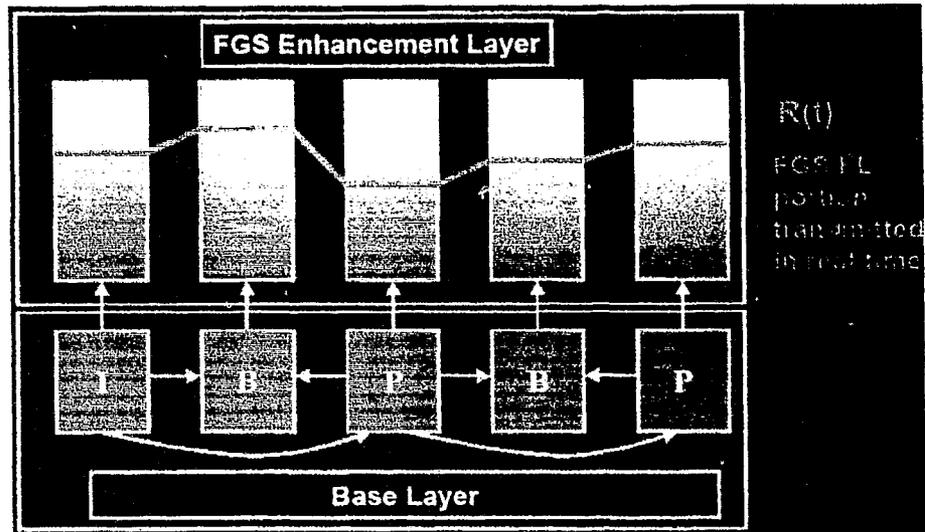


Figure 21 MPEG-4 FGS streaming at the sender

#### 4.3.3 Avoiding feedback implosion

Soliciting feedback from receivers in a multicast environment might cause a so-called implosion problem. A potential large amount of feedback information would consume the resource: session bandwidth.

The solution is limiting the feedback control traffic to a small and known fraction of the session bandwidth: small so that the primary function of the transport protocol to carry data is not impaired.

RTP specification recommends that the fraction of the session bandwidth added for RTCP be fixed at 5%. It also recommends that 1/4 of the RTCP bandwidth be dedicated to participants that are sending data so that in sessions with a large number of

receivers but a small number of senders, newly joining participants will more quickly receive the CNAME for the sending sites.

In the hybrid rate adaptation mechanism, the receivers adjust their individual receiving rate, the sender adjusts rate according to the proportion of the load status feedbacks of all session members, this benefits the framework to scale the RTCP transmission interval with the session sizes (members). To make the transmission rate of RTCP packets more adaptive to changes in the session group size, a "reverse reconsideration" algorithm proposed in [1] was deployed to computing the RTCP transmission interval.

# Chapter Five

## 5. TEST BED

### 5.1 The JMF architecture

Java Media Framework (JMF) provides a rich set of APIs to support the playing, streaming, and capturing of audio and video. JMF is designed to support most standard media content types including MPEG-4. By exploiting the advantages of the Java platform, JMF provides a common cross-platform Java API for accessing underlying media frameworks. The framework enables advanced developers and technology providers to perform custom processing of raw media data and seamlessly extend JMF to support additional content types and formats, optimize handling of supported formats, and create new presentation mechanisms. Figure 22 depicts the high level JMF architecture.

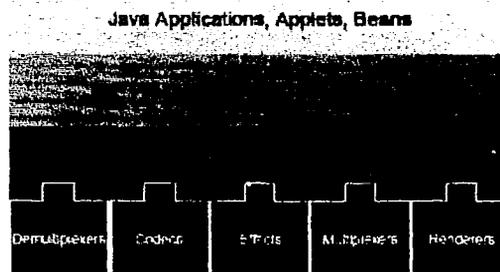


Figure 22 High-level JMF Architecture

The JMF component architecture is very flexible, and its components can generally be classified in three groups:

- (1) An **input** describes some sort of media that is used as an input to the rest of the process.
- (2) A **process** performs some sort of action on the input. A process has a distinct input and output. A large number of processes are available, and can be applied to an input or a group of inputs. These processes can be chained together so that the output from one process is used as an input to another process. In this manner multiple processes may be applied to an input.
- (3) An **output** describes some sort of destination for media.

Figure 23 shows the JMF media processing model.

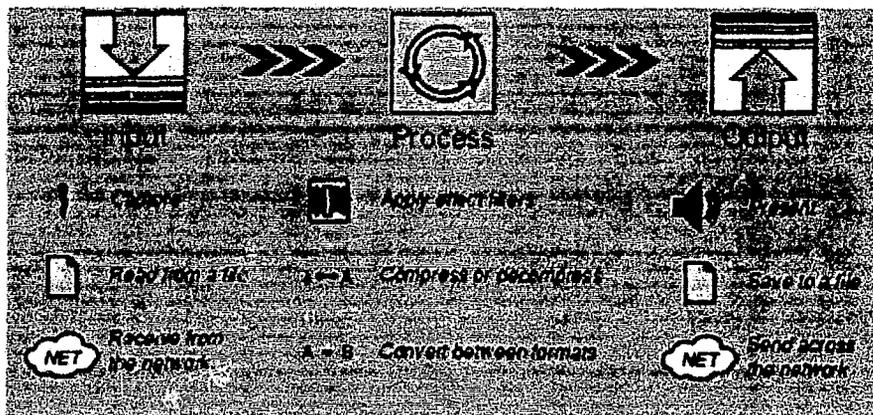


Figure 23 JMF Media Processing Model

## 5.2 JMF RTP APIs

JMF enables the playback and transmission of RTP streams through the APIs defined in the `javax.media.rtp`, `javax.media.rtp.event`, and

javax.media.rtp.rtcp packages. The JMF 2.1 FCS Reference Implementation supports:

- Presenting RTP streams using JMF players.
- Transmitting RTP streams using JMF processors and data sinks.
- Presenting and transmitting dynamic RTP payloads.
- Dynamically switching payload types during a session.
- Accessing session statistics and monitoring RTCP.
- Unicast, multicast, and broadcast sessions for both presentation and transmission.
- Network protocol independence for playback and transmission.

JMF can be extended to support additional RTP-specific formats and dynamic payloads through the standard JMF plug-in mechanism. Figure 24 illustrates the high level JMF RTP architecture.

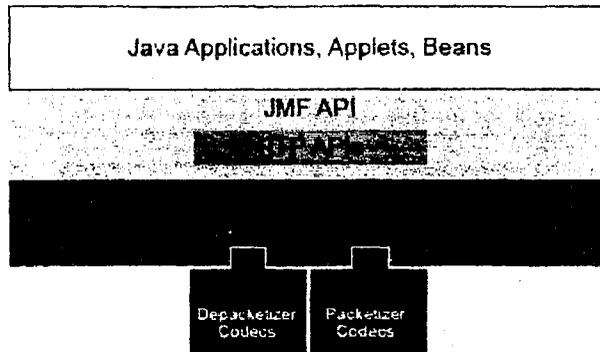


Figure 24 High-level JMF RTP Architecture

### 5.3 Implementation

The application implements the proposed framework for MPEG-4 FGS video streaming over RTP/UDP/IP with end-to-end rate adaptation.

#### 5.3.1 Session Manager

In JMF, an RTPSessionManager is implemented to coordinate an RTP session. The session manager keeps track of the session participants and the streams that are being transmitted.

The session manager maintains the state of the session as viewed from the local participant. In effect, a session manager is a local representation of a distributed entity, the RTP session. The session manager also handles the RTCP control channel, and supports RTCP for both senders and receivers.

The RTPSessionManager interface defines methods that enable an application to initialize and start participating in a session,

remove individual streams created by the application, and close the entire session. The Session Manager object (See Figure 25) is used for instantiating:

- (1) DataSources, which is used to deliver time-based multimedia data.
- (2) Players, which is used to control and render multimedia data.
- (3) Processors, which is used to process data and output the processed data.
- (4) DataSinks, which takes a DataSource as input and renders the output to a specified destination.

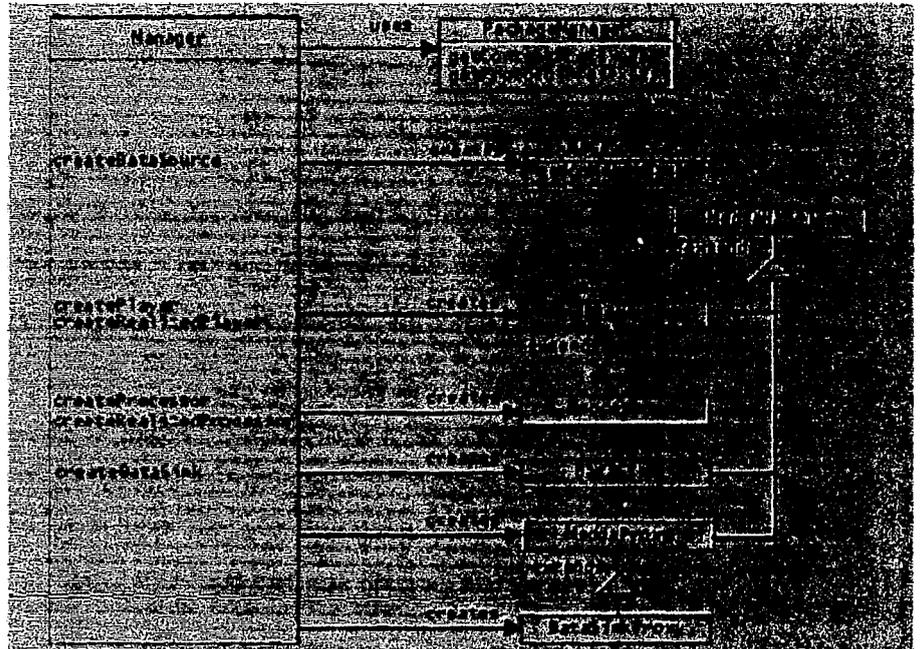


Figure 25 JMF Session Manager Object

### 5.3.2 Receiving and presenting RTP streams

The presentation of an incoming RTP stream is handled by a player. Figure 26 describes the JMF player object model. To receive and present a single stream from an RTP session, using a MediaLocator that describes the session to construct a player. A media locator for a RTP session is of the form: `rtp://address:port[:ssrc]/content-type/[ttl]`.

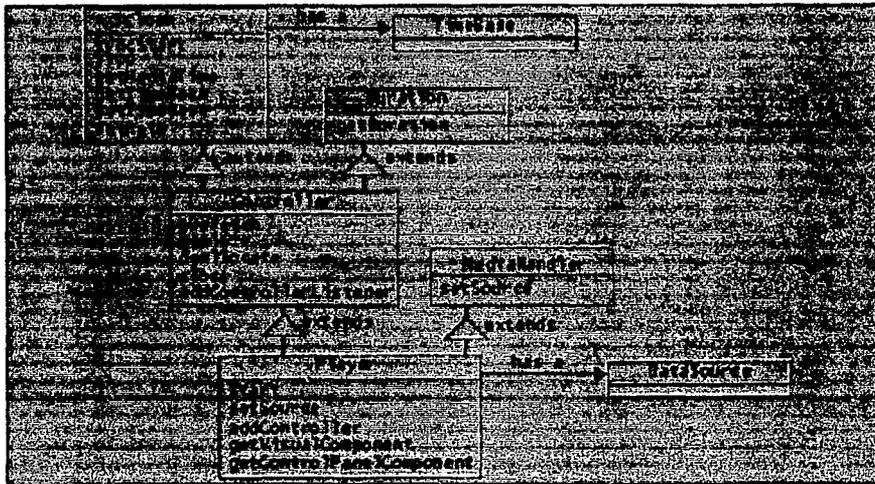


Figure 26 JMF Player Object

The player is constructed and connected to the first stream in the session. If there are multiple streams in the session, a session manager is needed. The session manager will notify whenever a stream is added to the session and construct a player for each new stream. The session manager also monitors and controls the session. Figure 27 shows the RTP reception process using JMF.

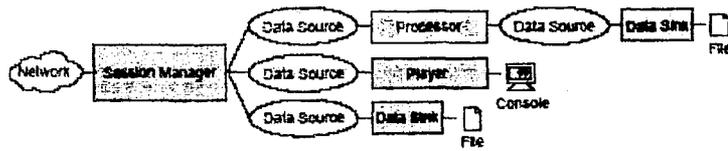


Figure 27 RTP reception using JMF

### 5.3.3 Transmitting RTP streams

A session manager can also be used to initialize and control a session so that you can stream data across the network. The data to be streamed is acquired from a Processor. Figure 28 illustrates the RTP transmission using JMF.

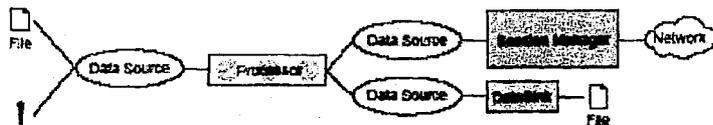


Figure 28 RTP Transmission using JMF

Steps to create a send stream to transmit data from a live capture source:

- (1) Create, initialize, and start an RTPSessionManager for the session.
- (2) Construct a Processor using the appropriate capture DataSource.
- (3) Set the output format of the Processor to an RTP-specific format. An appropriate RTP packetizer codec must be available for the data format to be transmitted.

- (4) Retrieve the output DataSource from the Processor.
- (5) Call createSendStream on the session manager and pass in the DataSource.

The transmission is controlled through the SendStream start and stop methods.

When it is first started, the RTPSessionManager behaves as a receiver (sends out RTCP receiver reports). As soon as a SendStream is created, it begins to send out RTCP sender reports and behaves as a sender host as long as one or more send streams exist. If all SendStreams are closed (not just stopped), the session manager reverts to being a passive receiver.

# Chapter Six

## 6. EXPERIMENT EVALUATION

### 6.1 Video quality evaluation

Over the past few years, video quality evaluation and measurement schemes have been developed [25, 26, 28, 29].

Video quality assessment technologies can be classified into: subjective, objective, and perceptual objective video quality assessment schemes.

#### 6.1.1 Subjective video quality assessment

Subjective video quality assessment is the most reliable video quality measurement method. A group of viewers is selected and gathered in a room, the environment of which is specified by the ITU-T Recommendation P.910 [28].

The source video and the processed video are presented in pairs to the viewers who are expected to grade the video quality. This is called Double Stimulus Continuous Quality Evaluation (DSCQE) as described in ITU-R BT.500-11. Reference is coded CIF without network side by side with coded CIF after networks. DSCQE method allows subjects to continuously monitor the quality of the material under test with the reference material side by side. This allows presenting two side-by-side CIF windows on the same screen (see Figure 29).

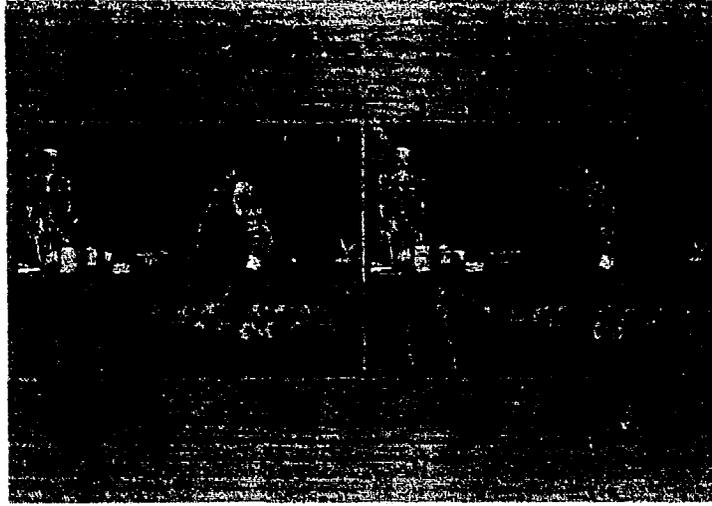


Figure 29 Example of DSCQE side-by-side presentation

Subjective video quality measurement has several disadvantages. It requires a special viewing room and equipment; it needs a large group of people to view the video; it requires a large amount of post processing on the video data. In conclusion, subjective video quality measurement cannot provide real-time in-service quality monitoring for real-time video applications. Figure 30 shows some examples of digital video impairments.



Original Image



Blurring



Tiling



Edge Noise

Figure 30 Examples of Digital Video Impairments

### 6.1.2 Objective video quality measurements

Objective video quality measurements, although not as accurate, can be conducted in the background without intruding on the end user. Objective video quality estimation software processes the video signals and produces the quality evaluation results.

One traditional objective video quality measurement, Peak Signal to Noise Ratio (PSNR), has been widely used in many applications to assess video quality. The standard deviation of PSNR is also a common used measure of the stability of video quality. A smaller standard deviation of PSNR means more stable video quality.

The advantage of PSNR is that it is very easy to compute. Assume a given source image  $f(i,j)$  that contains  $N$  by  $N$  pixels and a reconstructed image  $F(i,j)$  where  $F$  is reconstructed by decoding the encoded version of  $f(i,j)$ . Error metrics are computed on the luminance signal only so the pixel values  $f(i,j)$  range between black (0) and white (255).

First, the mean squared error (MSE) of the reconstructed image is computed as follows

$$\text{MSE} = \frac{\sum [f(i,j) - F(i,j)]^2}{N^2}$$

The summation is over all pixels. The root mean squared error (RMSE) is the square root of MSE.

PSNR in decibels (dB) is computed by using

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

Typical PSNR values range between 20 and 40. They are usually reported to two decimal points (e.g., 25.47). The actual value is not meaningful, but the comparison between two values for different reconstructed images gives one measure of quality. The MPEG committee used an informal threshold of 0.5 dB PSNR to decide whether to incorporate a coding optimization because they believed that an improvement of that magnitude would be visible.

There is a function in MATLAB to compute PSNR: PSNR(A,B), where A and B are MATLAB Intensity Images.

The standard deviation of PSNR is a measure of PSNR spread. It is calculated as following:

$$\text{STDEV}_{\text{PSNR}} = \sqrt{\frac{\sum (\text{PSNR} - \text{PSNR}_{\text{mean}})^2}{N-1}}$$

Where  $\text{PSNR}_{\text{mean}}$  is the mean of the sample, N is the sample size,  $\text{STDEV}_{\text{PSNR}}$  is the standard deviation of PSNR.

### **6.1.3 Perceptual objective video quality assessment**

Perceptual objective video quality assessment technologies try to achieve a high correlation with subjective video quality assessment without losing the advantages of that objective quality assessment has to offer. Stefan Winkler presented his Vision Model in [25]. This model simulates the human visual system and achieves a high correlation with subjective video quality assessment, but it is not capable of in service measurements and requires offline processing. The ANSI objective video quality standards T1.801.03-1996 [29] as well as the metrics developed by Institute for Telecommunication Sciences (ITS) capture the relationship between the measurable video quality parameters and perceptual quality distortions (blurring, tiling, noise, etc.).

## **6.2 Experiments**

To validate the effectiveness of the proposed framework, a series of experiments are carried out with multiple video sources under different channel conditions to assess the video quality at the receiving ends. Several scenarios are also identified for optimizing the algorithm parameters.

### **6.2.1 Test configuration and setup**

The simulation experiments were performed in the Intranet environment. Although Intranet users can afford high bandwidth for high quality video, to better simulate Internet conditions, only two streaming bandwidth options 64kbps and 128kbps were simulated,

because Internet 56K modem clients have only 36Kbps payload and ISDN clients have 128Kbps.

The test bed layout is shown as Figure 31.

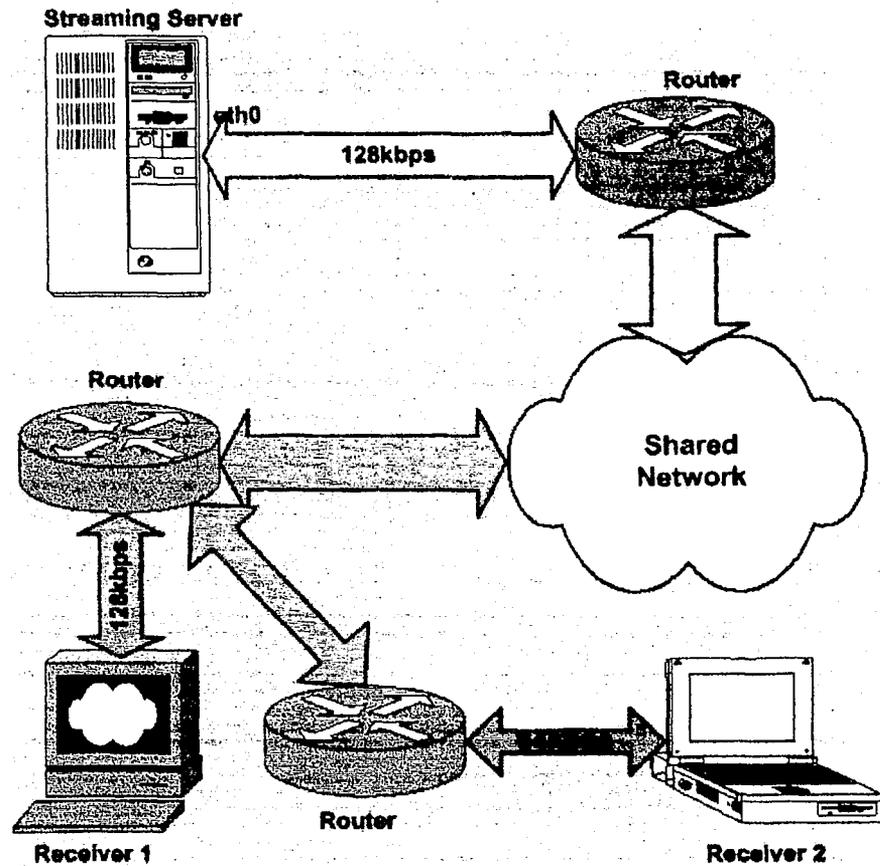


Figure 31 Test bed layout

Both the server and the receivers are running Red Hat Linux 8 on Intel-based box. The limitation of the outgoing bandwidth on the server is accomplished using the built-in QoS Class Based Queuing

(CBQ) functions in Linux Kernel. The outgoing rate of the server interface eth0 that connects the Intranet was configured as 128kbps.

The limitation of the incoming bandwidth is accomplished using a stand-alone kernel module called rshaper. Firstly, the rshaper.o module should be loaded using `# insmod /usr/lib/rshaper.o`. Then the incoming bandwidth limitation can be set using the rshaperctl utility. For example, `# rshaperctl 224.168.1.0/24 128000`, this shapes all incoming traffic on the local network to 128kbps. The reason to limit the incoming bandwidth is to simulate the heterogeneous Internet channels in a Intranet environment.

The server stream out using multicast defined IP addressed. IP addresses starting from 224.xxx.xxx.xxx on a network is for multicast. All users on the net can share and view the content by logging onto this address. The server will send only one stream out. Routers and network switches at different levels of the net will duplicate, filter and propagate the content to the downstream.

### 6.2.2 Rate controller parameters settings

A set of experiments was conducted to find the suitable controller parameters. For the filter parameter  $\alpha$ , if it is set too low, e.g.,  $\alpha = 0.1$ , then the current packet loss ratio (PLR<sub>c</sub>) value has not enough influence on the average packet loss ratio (PLR) but will significantly influence the next average packet loss ratio and the adjustment will be delayed. If it is set too high, e.g.,  $\alpha = 0.9$ , then

the current packet loss ratio value has high influence on the average packet loss ratio but this influence vanishes immediately with the advent of the next values if they indicate no or only few losses. Table 2 and Figure 32 shows PSNR comparison of the "Foreman" sequence at 128kbps with different filter parameter  $\alpha$ . It shows that when  $\alpha$  was set to 0.5, the PSNR standard deviation is 0.868. The PSNR standard deviations for  $\alpha=0.1$  and  $\alpha=0.9$  are 1.462 and 1.441 respectively. This means that when setting  $\alpha$  to 0.5, the framework produces the smallest PSNR standard deviation and hence provides the smoothest video quality. So the filter parameter  $\alpha$  is set to 0.5 in the simulation experiments.

Table 2 PSNR Comparison between different filters using "Foreman" sequence at 128kbps

PSNR(dB)	Foreman Sequence @128 kbps		
	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 0.9$
20	31.833	31.88	31.92
40	34.42	32.63	32.42
60	32.77	33.22	34.97
80	34.93	34.37	32.43
100	32.93	34.9	34.89
120	35.86	33.87	31.86
140	31.01	34.67	35.98
160	34.93	33.81	32.93
180	32.87	32.98	35.17
200	35.08	34.05	32.48
220	32.89	33.26	34.53
Mean	33.593	33.60364	33.59818
STDEV	1.46203	0.868167	1.441267



Figure 32 PSNR and  $\alpha$  relationship for "Foreman" sequence at 128kbps

The tolerable packet loss ratio  $PLR_0$  and the unacceptable packet loss ratio  $PLR_1$  are related with the rate-distortion characteristics of the source video and encoding techniques. They can be chosen from the video rate-distortion curve or empirical data and justified by experimental results. The two packet loss ratio thresholds was set as  $PLR_0 = 5\%$  and  $PLR_1 = 20\%$  in the experiments both for Foreman and Suzie sequences. Figure 33 shows an example relationship among the packet loss ratio, PSNR, and the perceptual video quality.

	
Original Image	PSNR=32.91; PLR=1%
	
PSNR=29.41; PLR=10%	PSNR=25.13; PLR=20%

Figure 33 PLR, PSNR and View Quality relationship examples

### 6.2.3 PSNR Comparisons

The evaluation tests use video sequences "Foreman" and "Suzie" in QCIF. The frame rate of MPEG-4 FGS coding is 10 frames per second. For comparison, MPEG-4 Annex L [51] rate control solution was chosen as the reference.

MPEG-4 Annex L is a scalable rate control scheme. It assumes the following model for the encoder rate R:

$$R = \frac{X_1 S}{Q} + \frac{X_2 S}{Q^2}$$

**Q** is the quantiser step size. **S** is the mean absolute difference of the residual frame after motion compensation. **X<sub>1</sub>** and **X<sub>2</sub>** are model parameters. **S** provides a measure of frame complexity.

MPEG-4 Annex L rate control consists of a set of steps that are carried out after motion compensation and before encoding of each frame.

Table 3 and Figure 34-37 show the PSNR comparison between the proposed rate adaptation framework and the MPEG-4 Annex L rate control solution at different channel bit rates.

Table 3 PSNRs comparison scenario

PSNR (dB)	Foreman Sequence @128 kbps		Foreman Sequence @64 kbps		Suzie Sequence @128 kbps		Suzie Sequence @64 kbps	
	MPEG-4 Annex L	Proposed RA	MPEG-4 Annex L	Proposed RA	MPEG-4 Annex L	Proposed RA	MPEG-4 Annex L	Proposed RA
20	36.23	32.63	34.53	25.12	33.23	30.28	31.54	27.53
40	33.42	33.23	23.34	28.37	30.42	31.33	28.42	28.32
60	32.77	33.52	31.54	30.28	32.27	32.42	31.27	29.42
80	34.43	34.37	27.13	32.22	32.53	32.37	27.53	28.34
100	32.93	34.9	26.78	30.34	30.23	31.91	28.29	28.95
120	35.86	33.87	33.97	31.08	33.99	32.17	28.43	29.23
140	31.21	34.67	26.95	28.93	31.91	32.67	30.91	28.56
160	34.93	33.81	35.64	30.42	32.93	31.87	27.04	27.87
180	32.87	32.98	26.67	28.74	32.87	32.98	30.87	27.67
200	35.08	34.05	31.86	30.04	30.68	32.05	27.68	28.07
AVG	33.973	33.803	29.841	29.554	32.106	32.005	29.198	28.396
STDEV	1.506539	0.693095	3.962601	1.83525	1.213097	0.720142	1.651295	0.612114

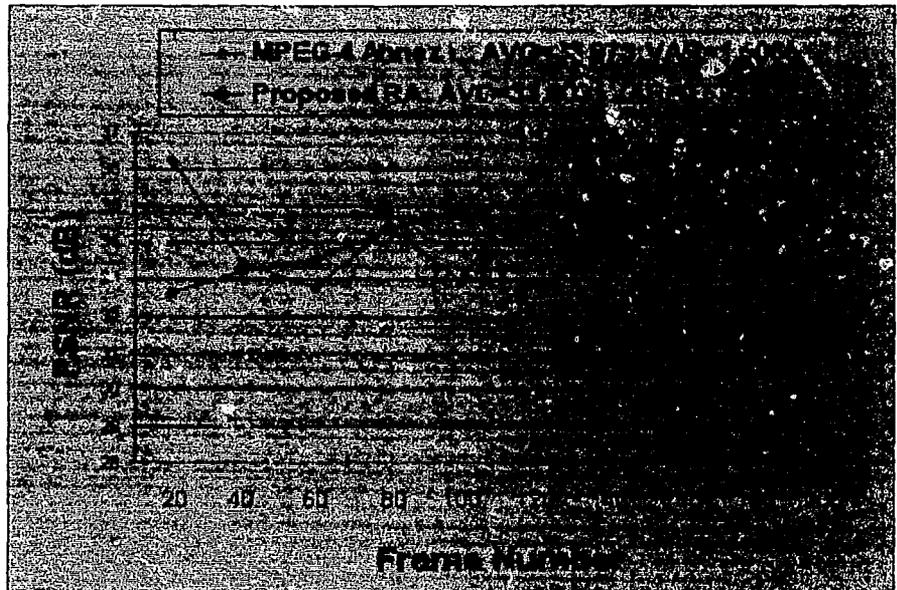


Figure 34 Comparison with Foreman sequence at 128kbps

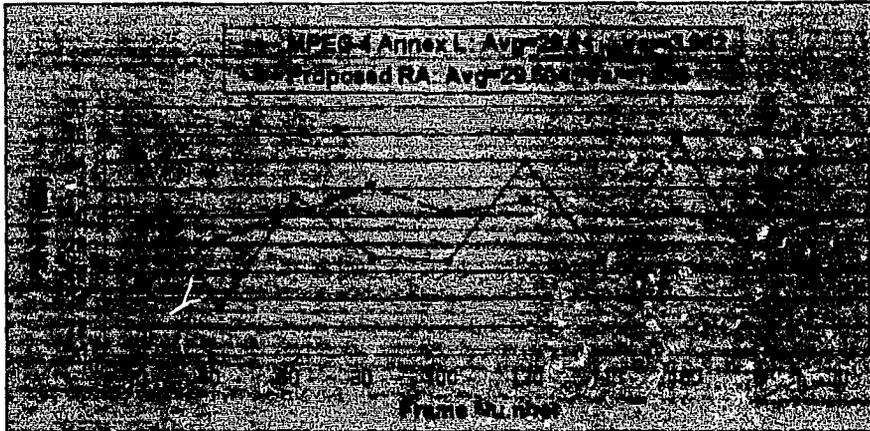


Figure 35 Comparison with Foreman sequence at 64kbps

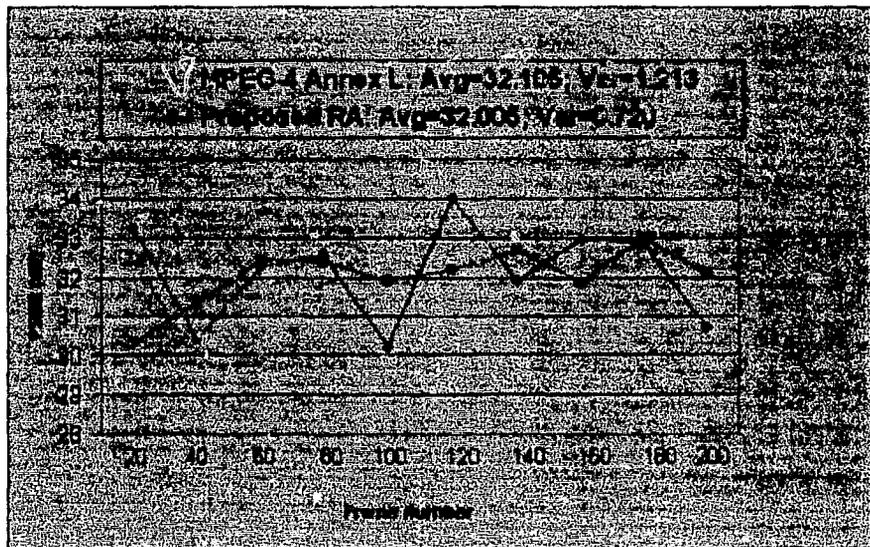


Figure 36 Comparison with Suzie sequence at 128kbps

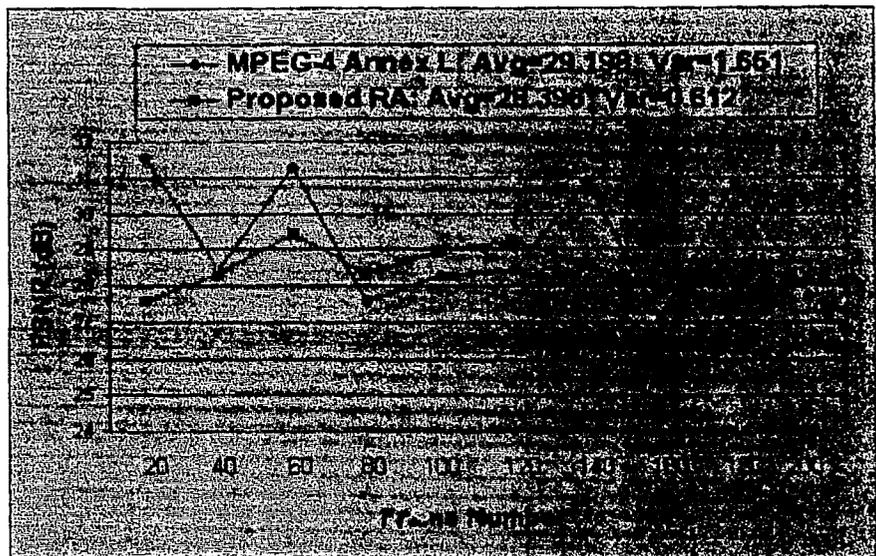


Figure 37 Comparison with Suzie sequence at 64kbps

The above table 3 and figures 34-37 show that compared with the MPEG-4 Annex L solution, the standard deviation of PSNR at each receiver of the proposed framework is much lower than that of MPEG-4 Annex L solution, while the average PSNR are at the same level. Thus, the proposed framework provides more stable video quality and the end users can get more smooth and pleasant viewing experience. This achievement is mainly due to the feature of the rate adaptation framework that each receiver detects the available bandwidth on the path and adjusts the subscribing rate promptly to match the network conditions.

# Chapter Seven

## 7. CONCLUSIONS

The Internet holds great promise as a medium for real-time streaming video because of its ubiquity and flexibility. However, today's Internet was originally designed for real-time streaming applications, there are still many challenges for real-time streaming over the Internet.

Congestion control is one of these challenges. In this thesis, an end-to-end hybrid rate adaptation framework for MPEG-4 FGS video streaming over IP was proposed to address the network congestion control challenge in both unicast and multicast environment.

The main contributions of this thesis are:

- (1) Combined the sender adaptive and the receiver driven rate control mechanisms into a hybrid rate adaptation framework that takes into account the rate-distortion characteristics of the source as well as the time-varying network conditions. Each receiver estimates the available bandwidth on the path, thus each receiver can realize its fair video quality matching the available bandwidth on its path and the process capability of its terminal.
- (2) Proposed algorithms for the sender derive the adjustment decision from the proportion of receivers' load status, so that

a receiver or a small proportion of receivers connected via a low bandwidth link cannot force the sender to provide low quality video to other receivers which have high bandwidth connections in a heterogeneous multicast environment, and vice versa. In addition, this helps to reduce the overall PSNR standard deviations and therefore to avoid QoS oscillations.

A set of designed experiments tuned the algorithms and controller's parameters for selected MPEG-4 FGS video sequences. Some of the parameters are related with the rate-distortion characteristics of the source video, so the rate controller should be adjusted accordingly.

The experiment results show that this framework archives the inter-receiver fairness in a heterogeneous environment, reduces the PSNR standard deviations, and thus improves the video quality stability. The end-to-end approach can be employed by the end systems, so it is effective and applicable in actual network environments.

Nevertheless, many open questions and challenges still remain; the framework presented a solution to make MPEG-4 FGS video streaming over the current Internet with end-to-end QoS a reality.

The future Internet [35] would add more services for real-time streaming applications. An important direction of ongoing and future work is to combine the end-to-end approaches with the next generation Internet architecture. We are also working to extend our test bed application.

## REFERENCES

- [1] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", IETF Draft, draft-ietf-avt-rtp-new-12.txt, September 2003
- [2] Rob Koenen "MPEG-4 Overview (V.21-Jeju Version)", ISO/IEC JTC1/SC29/WG11 N4668, March 2002.
- [3] A. Leon-Garcia, I. Widjaja, Communication Networks Fundamental Concepts and Key Architectures, McGraw Hill, 2000
- [4] D. Wu, T. Hou, W. Zhu, Y.-Q. Zhang, J. Peha, "Streaming Video over the Internet: Approaches and Directions," IEEE Transactions on Circuits and Systems for Video Technology, vol. 11, no. 3, pp. 282-300, March 2001
- [5] Weiping Li "Overview of fine granularity scalability in MPEG-4 video standard" IEEE Transactions on Circuits and Systems for Video Technology, vol. 11, no. 3, pp. 301-317, March 2001.
- [6] J-C. Bolot, T. Turetti, and I. Wakeman, "Scalable feedback control for multicast video distribution in the Internet" in Proc. the Conference on Communications Architectures, Protocols and Applications, pp. 58-67, Sept. 1994, London, UK
- [7] P. Gentric et al., "RTP Payload Format for MPEG-4 Streams", IETF Draft, draft-ietf-avt-mpeg4-multiSL-02.txt
- [8] S. Li, F. Wu, and Y.-Q. Zhang, "Study of a new approach to improve FGS video coding efficiency" ISO/IEC JTC1/SC29/WG11, MPEG99/M5583, Dec. 1999.

- [9] W. Li, "Bit-plane coding of DCT coefficients for fine granularity scalability", ISO/IEC JTC1/SC29/WG11, MPEG98/M3989, Oct. 1998.
- [10] W. Li: "Streaming video profile in MPEG-4", IEEE trans. on Circuits and Systems for Video Technology, Vol. 11, no. 3, pp. 301-317, Mar 2001.
- [11] F. Wu, S. Li, and 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, pp. 332-344, Mar. 2001.
- [12] F. Leannec, J. Vieron, X. Henocq, and C. Guillemont, "Hybrid sender and receiver driven rate control in multicast layered video transmission", in Proc. IEEE ICIP 2000, vol.3, pp.532-535, Sept. 2000.
- [13] J. Vickers, C. Albuquerque, and T. Suda, "Source daptive Multi-Layered Multicast Algorithms for Real-Time Video Distribution", IEEE/ACM Transactions on Networking, vol. 8, no. 6, pp. 720 – 733, Dec. 2000.
- [14] E.Rosen, A.Viswanathan, and R.Callon. "Multiprotocol Label Switching Architecture", IETF RFC 3031, January 2001
- [15] R. Guerin, A. Orda, and D. Williams. "QoS Routing Mechanisms and OSPF Extensions" IETF RFC 2676, August 1999.
- [16] D. Cavendish and M. Gerla. "Internet QoS Routing using the Bellman-Ford Algorithm", in Proc. IFIP Conference on high performance networking, pp. 627-646, Sept. 1998

- [17] S. McCanne, V. Jacobson , and M. Vetterli, "Receiver-Driven Layered Multicast", in Proc. ACM SIGCOMM, pp. 117—130, 1996
- [18] Li X., Paul S., and Ammar M. "Layered Video Multicast with Retransmissions (LVMR): Evaluation of Hierarchical Rate Control", in Proc. IEEE Infocom, pp. 1062-1072, June 1998
- [19] Vicisano L. and Crowcroft J. "TCP-like Congestion Control for Layered Multicast Data Transfer", in Proc. RMRG Meeting, pp. 996-1003, Dec.1998
- [20] H. Schulzrinne, A. Rao, R. Lanphier, M. Westerlund, "Real Time Streaming Protocol" IETF Draft, draft-ietf-mmusic-rfc2326bis-02.txt
- [21] L. Zhao, J. Shin, J. Kim, J. Kuo, "MPEG-4 FGS Video Streaming with Constant-Quality Rate Control and Differentiated Forwarding" in Proc. VCIP'02, pp. 230-241, Jan. 2002
- [22] J. van der Meer, D. Mackie, V. Swaminathan, D. Singer, P. Gentric, "Transport of MPEG-4 Elementary Streams", draft-ietf-avt-mpeg4-simple-04.txt, July 2002
- [23] Java, <http://java.sun.com/>
- [24] Java Media Framework, <http://java.sun.com/products/java-media/jmf/>
- [25] Stefan Winkler, "A perceptual distortion metric for digital color images", in Proc. IEEE International Conference on Image Processing, vol. 3, pp. 399-403 , Chicago, IL, October 4-7, 1998.
- [26] W. Ashmawi, R. Guerin, S. Wolf, and M. Pinson, "On the impact of policing and rate guarantees in Diff-Serv networks: A video streaming

application perspective", in Proc. of ACM SIGCOMM, pp. 83-95, San Diego, CA, 2001.

- [27] T. Jiang, M. H. Ammar, and E. W. Zegura. Inter-receiver fairness: A novel performance measure for multicast ABR sessions. Proc. of SIGMETRICS, pp. 202–211, June 1998.
- [28] ITU-T Recommendation P.910, "Subjective video quality assessment methods for multimedia applications," Recommendations of the ITU, Telecommunication Standardization Sector.
- [29] ANSI T1.801.03-1996, "American National Standard for telecommunications- digital transport of one-way video signals- parameters for objective performance assessment", American National Standards Institute.
- [30] Rob Koenen, "Object-based MPEG offers flexibility", EE Times, Nov. 2001, <http://www.eetimes.com/story/OEG20011112S0042>
- [31] Rob Koenen, "From MPEG-1 to MPEG-21: Creating an Interoperable Multimedia Infrastructure", ISO/IEC JTC1/SC29/WG11 N4518, Pattaya, Dec. 2001.
- [32] B. Krishnamurthy, C. Wills, Y. Zhang, " On the use and performance of Content Distribution Networks", in Proc. ACM SIGCOMM, pp. 169-182, 2001
- [33] J. Bormans, K. Hill, "MPEG-21 Overview V.5" ISO/IEC JTC1/SC29/WG11/N5231, Shanghai, Oct. 2002
- [34] D. Clark, S. Shenker, L. Zhang, "Supporting Real-time Applications in an Integrated Services Packet network: Architecture and Mechanism" in Proc. the ACM SIGCOMM, pp.14-16, 1992

- [35] R. Braden, D. Clark, S. Shenker, J. Wroclawski, "Developing a Next-Generation Internet Architecture", July, 2000. [www.ana.lcs.mit.edu/papers/PDF/NewArch\\_wp\\_v1.pdf](http://www.ana.lcs.mit.edu/papers/PDF/NewArch_wp_v1.pdf)
- [36] P. de Cuetos, M. Reisslein, K. Ross, "Evaluating the Streaming of FGS-Encoded Video with Rate-Distortion Traces", Technical Report RR-03-078, Institut Eurecom, June 2003
- [37] R. Braden et al. "Resource ReReservation Protocol (RSVP) Version 1 - Functional Specification", IETF RFC 2205, September 1997.
- [38] S. Blake et al. "An Architecture for Differentiated Services", IETF RFC 2475, December 1998.
- [39] D. Bansal, H. Balakrishnan, "Binomial Congestion Control Algorithms", in Proc. IEEE INFOCOM, pp631-640, April 2001
- [40] S. Floyd, K. Fall, "Promoting the use of end-to-end congestion control in the Internet", IEEE/ACM Transactions on Networking, vol. 7, no. 4, pp. 458-472, August 1999.
- [41] M. Allman, V. Paxson, "TCP Congestion Control", IETF RFC2581, April 1999
- [42] B. Braden, D. Clark, 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", Internet Engineering Task Force, April 1998, IETF RFC 2309.

- [43] D-M. Chiu, R. Jain, "Analysis of the Increase and Decrease Algorithms for Congestion Avoidance in Computer Networks", Computer Networks and ISDN Systems, vol.17, pp.1-14, 1989
- [44] J. Vieron, T. Turetli, X. Hjnocq, C. Guillemot, and K. Salamatian, "TCP-Compatible Rate Control for FGS Layered Multicast Video Transmission Based on a Clustering Algorithm". In Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS), pp. 453-456, 2002.
- [45] Microsoft MPEG-4 video codec reference software (Microsoft-FDAM1-2.3-001213)  
<http://research.microsoft.com/~swinder/mpeg4.aspx>
- [46] W. Li, F. Ling, and X. Chen, "Fine Granularity Scalability in MPEG-4 for Streaming Video", ISCAS 2000, vol.1, pp. 299-302, Geneva, Switzerland, May 2000.
- [47] Y.-S. Tung, J.-L. Wu, P.-K. Hsiao, and K.-L. Huang, "An Efficient Streaming and Decoding Architecture for Stored FGS Video", IEEE Transactions on Circuits and Systems for Video Technology, Volume: 12, Issue: 8, pp.730-735, August 2002.
- [48] J. Vieron, T. Turetli, X. Hjnocq, C. Guillemot, and K. Salamatian, "TCP-Compatible Rate Control for FGS Layered Multicast Video Transmission Based on a Clustering Algorithm", In Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS), pp. 453-456, 2002.
- [49] MPEG4IP project, <http://www.mpeg4ip.net>
- [50] IBM Toolkit for MPEG-4, <http://www.alphaworks.ibm.com/tech/tk4mpeg4>

- [51] ISO/IEC 14496-2: Information technology -- Coding of audio-visual objects -- Part 2: Visual.
- [52] I. Rhee, N. Balaguru and G.N. Rouskas, "MTCP: Scalable TCP-Like Congestion Control for Reliable Multicast", In Proceeding IEEE INFOCOM, New York, pp. 1265-1273, March 1999.
- [53] A. Legout and E.W. Biersack, "PLM: Fast Convergence for Cumulative Layered Multicast Transmission Schemes", In Proceeding ACM SIGMETRICS, pp. 13-22, Santa Clara CA, June 2000.
- [54] John W. Byers, Grain Horn, Michael Luby, Michael Mitzenmader and William Shaver, "FLID-DL: Congestion Control for Layered Multicast", IEEE Journal on Selected areas in Communication, Vol.20, No.8, 1558-1570, October 2000.
- [55] Sherlin Shi and W. Marcel, "A Rate-Based End-to-End Multicast Congestion Control Protocol", Computer and Communications-2000, in Proceeding ISCC-2000, pp. 678-686, 2000.
- [56] S.J. Golestani, "Fundamental Observations on Multicast Congestion Control in the Internet", In Proc. IEEE INFOCOM, pp. 990-1000, New York, March 1999.
- [57] B. Whetter and J. Conlan, "A Rate-based Congestion Control Scheme for Reliable Multicast", Technical White Paper, GlobalCast Communications 1998.
- [58] H. J. Chao and X. Guo, Quality of Service Control in High-Speed Networks, John Wiley & Sons. ISBN: 0-471-00397-2, Nov. 2001

- [59] Paul P. White, "RSVP and Integrated Services in the Internet: A Tutorial", IEEE Communications Magazine, pp.100-106, May 1997.
- [60] K. Almeroth, "The Evolution of Multicast: From the Mbone to Interdomain Multicast to Internet2 Deployment", IEEE Network, Vol. 14, pp.10-20, January/February 2000.
- [61] C. Diot et al, "Deployment Issues for the IP Multicast Service and Architecture", IEEE Network, Vol. 14, no. 1, pp. 78-88, Jan./Feb. 2000.
- [62] D. Wu, T. Hou, B. Li, W. Zhu, Y.-Q. Zhang, H. J. Chao, "An End-to-End Approach for Optimal Mode Selection in Internet Video Communication: Theory and Application," IEEE Journal on Selected Areas in Communications, vol. 18, no. 6, pp. 977-995, June 2000
- [63] "Internet Protocol Multicast" [http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito\\_doc/ipmulti.htm](http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/ipmulti.htm)
- [64] Bill Fenner, Mark Handley, Hugh Holbrook, Isidor Kouvelas, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", draft-ietf-pim-sm-v2-new-08.txt, Oct. 2003
- [65] S. Waldbusser "Remote Network Monitoring Management Information Base", IETF RFC 1757, February 1995

## GLOSSARY

MPEG - Moving Picture Experts Group

FGS - Fine Granularity Scalability

RTP - Real-time Transport Protocol.

VBLV - Very Low Bitrate Video

UDP - User Datagram Protocol

SNR - Signal to Noise Ratio

PSNR - Peak Signal to Noise Ratio

QoS - Quality of Service

DMIF - Delivery Multimedia Integration Framework

AU - Access Unit in an ES

BIFS - binary format for scenes

ES - elementary stream

SL - synchronization layer

CIF - Common Intermediate Format

QCIF - Quarter-CIF

DSCQE - Double Stimulus Continuous Quality Evaluation

PLR - Packet Loss Ratio

PLR<sub>1</sub> - Unacceptable Packet Loss Ratio

PLR<sub>0</sub> - Tolerable Packet Loss Ratio

ITU - International Telecommunication Union

AIMD - Additive Increase / Multiplicative Decrease

RTT - Round Trip Time

DCT - Discrete Cosine Transform

API - Application Programming Interface

VOP - Video Object Plane

VRML - Virtual Reality Modeling Language

PIM - Protocol Independent Multicast

BL - Base Layer

EL - Enhancement Layer

STDEV - Standard Deviation