USC–SIPI REPORT #324

Rate Control for Video Transmission Over Variable Rate Channels

by

Chi-Yuan Hsu

August 1998

Signal and Image Processing Institute
UNIVERSITY OF SOUTHERN CALIFORNIA
Department of Electrical Engineering-Systems
3740 McClintock Avenue, Room 400
Los Angeles, CA 90089-2564 U.S.A.

To my parents.

Acknowledgements

First of all, I would like to thank my advisor Professor Antonio Ortega for his continuous support and help in my research, I also like to thank him for his effort in creating an enjoyable research environment in our group. I thank all my colleagues in Prof Ortega's research group, for sharing our ideas – and pizza – in every seminar, also for the pleasant collaboration experience and all the fun we have together. I also like to thank all my friends in Signal and Image Processing and University of Southern California for I have a very pleasant stay in Los Angeles for the past years.

Special thanks goes to Dr. Amy Reibman of AT&T Bell Laboratories for her insightful suggestion in this research, and generous contribution to the publication of our work in the *IEEE Journal on Selected Areas in Communications*, for which Dr. Antonio Ortega, Dr. Amy Reibman and I were selected as the winners for the *IEEE Communications Society Leonard G. Abraham Prize Paper Award* of 1997.

I also like to thank Dr. Masoud Khansari of Hewlett-Packard Laboratories for the joint collaboration in this research, which made possible the publication of many papers and this dissertation.

I would like to thank Professor John Silvester, Professor C.-C. Jay Kuo, Professor Ulrich Neumann, Professor Keith Chugg for their time and precious suggestions as being my guidance committee of the dissertation. I also thank Prof. Robert Scholtz for giving me the opportunity to participate his research, which benefits me a lot in my research.

I like to thank my parents and my family for their love and endless support. Without their unselfish support, all of this work would not be possible.

Contents

List Of Tables

List Of Figures

Abstract

Video has become a component of modern communications and multimedia applications. Communication networks and channels with very different characteristics, such as circuit-switched networks, packet-switched networks, Internet, ATM-based B-ISDN, and wireless channels, have been explored as the platform for video transmission. However, the underlying networks and channels place different constraints on the transport of video data. To comply with these constraints, it is necessary to implement rate control at the video encoder in a video transmission applications.

In this research, the problem of video rate control is studied by considering the constraining factors in a video transmission system (including the applicable end-to-end delay, channel throughput, and possible transmission errors) and translating these constraints into the encoding rate. The encoding rate constraints have to be observed by the encoder in order to guarantee the successful decoding and displaying of the video data that are streaming into the decoder.

Based on this formulation of the rate control problem, a number of rate control algorithms, aim to maximize the video quality that are transmitted to the decoder, are proposed for video transmission over various types of transmission channels. The first communication channel under study for video transmission is a Variable Bit Rate (VBR) network with usage parameter control. A rate control approach is proposed to jointly select the source and channel rates in such VBR transmission

environment. Another transmission environment under study for video transmission is burst-error channels. An integrated rate control and error control scheme is proposed such that the source encoding rate can be adaptive to the current channel condition. Overall performance of the video transmission, in terms of reconstructed video quality and data loss rate, can be improved.

Chapter 1

Introduction

1.1 Digital Video and Video Transmission

Recent advances in digital video technology have fundamentally changed the way visual information is processed, stored and transfered. As other technological components such as semiconductor, microprocessors, digital communication and digital signal processing become mature and available, digital video is replacing the traditional analog video with the emergence of various applications such as High Definition TV (HDTV) [1], Video CD (VCD), Digital Versatile Disk (DVD), Digital Video Broadcasting (DVB) [2], and video conferencing. An advantageous property of digital video is that many digital signal processing techniques (e.g., compression, encoding, encryption and error correction) can be directly applied on the video signal which has been digitized. Another reason for the trend of such transition from analog to digital in video technology can be attributed to the wide availability of digital storage devices, computers, digital signal processors and digital communication systems, so that video information is more likely to be processed, stored, and transfered on those digital platforms.

However, video is also one of the most bandwidth-consuming types of information. Even as new technology causes communication channels with ever-increasing bandwidth, compression of the video data is still necessary to achieve more efficient bandwidth utilization. For example, in the terrestrial TV broadcasting, satellite TV and cable TV applications, more high resolution video channels can be delivered within the assigned bandwidth if each video signal is compressed for transmission. In the applications of video transmission over digital data network, compression of the video data is necessary to reduce the impact and burden on the network traffic. Therefore, the development of efficient video codecs has attracted substantial research interest. The demand for a common video compression scheme that makes it possible to exchange video information in various applications also triggered several standardization efforts by organizations such as International Standards Organization (ISO) and International Telecommunication Union–Telecommunication Standardization Sector (ITU-T) to develop video compression standards such as MPEG-1 [3], MPEG-2 [4], H.261 [5] and H.263 [6]. The MPEG-1 standard addresses the requirement of video data compression up to about 1.5MBit/sec, and targets the application of video storage on media such as CD-ROM, as well as transmission over narrowband communication channel such as Narrowband Integrated Service Digital Network (N-ISDN). The MPEG-2 standard offers more features and higher video quality than MPEG-1, and aims to provide a generic compression scheme for a vast variety of applications such as digital video broadcasting, DVD and video over broadband ISDN (B-ISDN). H.261 and H.263 standards are developed by ITU-T and are specifically designed for real-time low-bit rate video conferencing services over N-ISDN or Public Switched Telephone Network (PSTN) where bandwidth constraint and low delay latency are of primary concern.

While in some cases digital video is stored locally, in a number of video applications a communication channel is used to transfer and distribute the video information. This form of video communication applications includes video conferencing, video-phone, video on demand, and to some extent – Personal Communication System (PCS). A general structure for such video transmission system may consist of following components: input and output of video, a encoder and decoder to compress and decompress video data, buffers to store video data before transmission or after reception, and a transmission channel. Fig. 1.1 shows the generic structure of such video communication system.

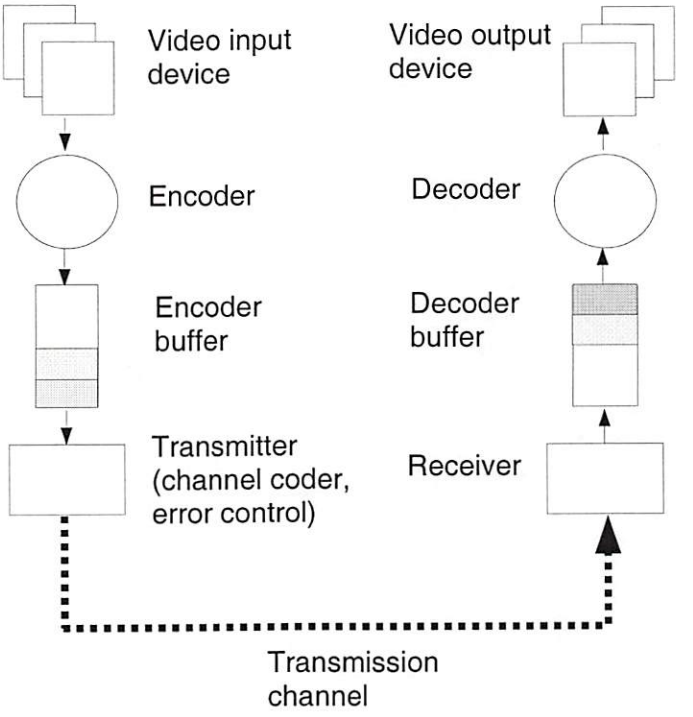


Figure 1.1: Structure of a generic video communication system.

Unlike other type of digital data transmission, video transmission is subject to certain time and delay constraints because each compressed video frame has to be transfered to the decoder before the time it is scheduled to be decoded. Due to such constraints, a much higher level Quality of Service (QoS) in terms of the guaranteed

transmission bandwidth, bounded delay and delay jitter, is required to support the video transmission.

A distinguishing property of video transmission is the high data volume and Variable Bit Rate (VBR) nature of the video data stream. The bursty nature of the video data stream may cause severe degradation on the entire network service if sudden increases in the video data traffic occur. Hence, certain bit-rate constraints on the video data are required. Rate control is also necessary in the Constant Bit Rate (CBR) transmission, e.g., TV broadcast or cable TV, where the variable bit rate data stream of the video information has to be regulated in order to comply with the channel bandwidth and buffer constraints.

The VBR nature of video data stream can be attributed to the fact that data compression techniques such as quantization and variable length coding are commonly used in most video codecs. The number of bits that is required for encoding each video frame (which will be referred as "encoding rate" in our later discussion) is not constant throughout a video sequence. However, most video codecs have built-in adjustable compression parameters to increase or decrease the encoding rates in order to accommodate the different transmission bandwidths of the various types of channels. Given this flexibility in selecting the operating encoding rates, the rate control can also be used to regulate the bit stream of the video data from the source encoder in a way such that the external constraints imposed by the transmission channel and application can be complied with.

In this chapter, we will briefly introduce some commonly used video compression standards. We focus our study on video transmission over VBR channels, therefore in Section 1.4 we will investigate the properties of some VBR channels that will be used in our later study of video transmission.

1.2 Video Compression

The standardization of digital video compression is still evolving. In addition to MPEG-1, MPEG-2, H.261 and H.263 standards discussed above, other video compression standards such as MPEG-4 or the extension of H.263 [7] will be finalized in the near future. Without precluding the use of other compression schemes, the compression standards mentioned above (i.e., MPEG-1, MPEG-2, H.261 and H.263) are transform-based coding systems which use 2-D Discrete Cosine Transform (DCT) [8] to transform spatial image data into a frequency domain representation, in which most energy of the signal is compacted in the low frequency coefficients.

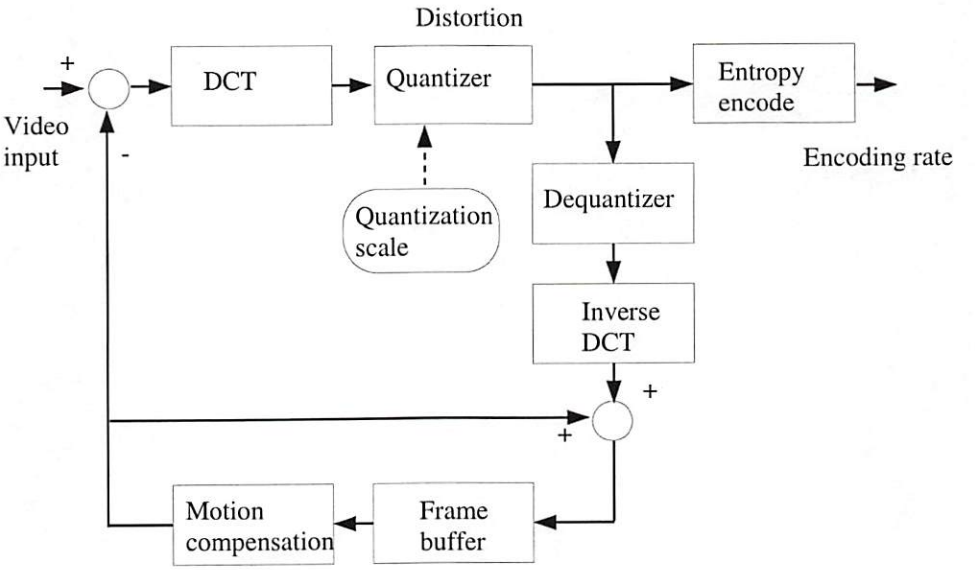


Figure 1.2: Block diagram of a general DCT-based encoder.

Those DCT-based coding systems can be divided into the following steps: DCT transform of the image, quantization of the resulting coefficients, and entropy coding of the quantized coefficients [9, 10]. Fig. 1.2 depicts a general DCT-based encoder. DCT coefficients are unequally quantized by a quantization matrix which is specially designed according to the spectral response of the human visual system.

A Further degree of compression on video signal can be achieved by reducing the temporal redundancy between consecutive video frames, which is often achieved by Motion Compensation (MC) [11, 12, 13]. In video compression with MC, video frames are predictively encoded by using motion vectors to indicate the translation of each sub-divided region in a predicted frame from other frame(s) [14]. The process to search for such motion vectors is called Motion Estimation (ME). In the MPEG standards, for example, multiple prediction methods (i.e., predictive coding from previous frames and bidirectional coding from previous and future frames) are used to effectively reduce the prediction error in ME. After MC, only the resulting prediction errors and motion vectors have to be encoded.

Video frames are classified into three picture types depending on which prediction methods are used: **Intra frame (I-frame)** which is independently encoded without MC; **Predicted frame (P-frame)** which is unidirectionally predicted from previous frame; and **Interpolated frame (B-frame)** which is bidirectionally predicted from previous and future frames. Fig. 1.3 depicts an example that video frames are are encoded in different in picture types in a predictive video coding scheme.

1.2.1 Variable quantization

The video codec discussed above is a "lossy" compression approach because the quantization of the DCT coefficients induces noise in the reconstructed video. While the introduced distortion in the reconstructed video may be imperceptible or tolerable by the users, video encoding by lossy compression schemes can achieve much higher compression ratio than that of the lossless compression schemes. One property of the lossy compression approaches is that the video encoding rates usually are scalable in those approaches. In the DCT-based compression scheme discussed above, for

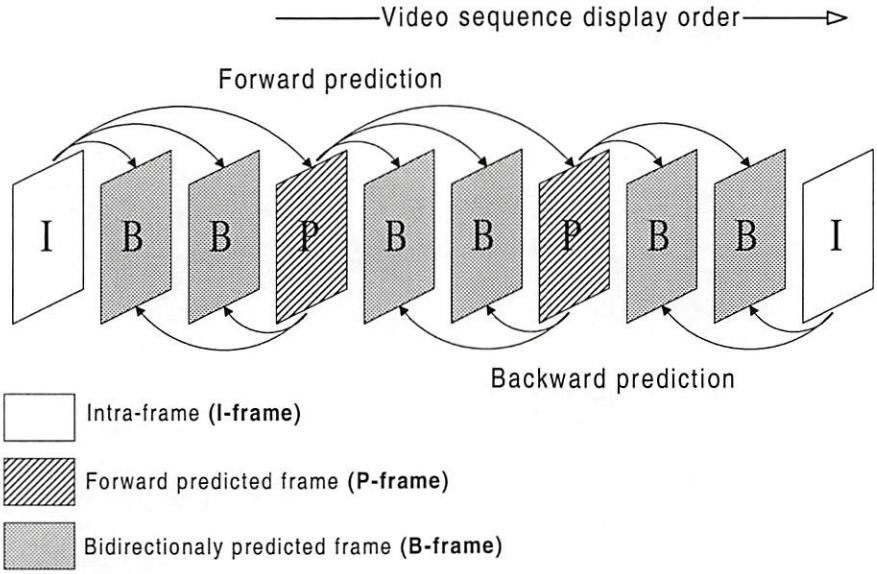


Figure 1.3: Video frames that are encoded in different picture types in a predictive video coding scheme.

example, the resulting encoding rate and the associated distortion of a video frame are dependent on how coarsely the DCT coefficients are quantized. The coarseness of the quantization can be universally scaled by adjusting the quantization step size (e.g., the MQUANT parameter in MPEG). Therefore, varying the encoding rate can be achieved by adjusting the quantization scale at the encoder.

Nevertheless, other encoding parameters, such as the selection of the picture-type and macroblock-type, also affect the resulting encoding rate and distortion, and rate control also can be achieved by mode selection [15]. In general, the distortion increases as the encoding rate decreases, and vice versa. Our goal will be selecting the encoding rate for each frame so as to minimize the overall distortion of the whole video sequence, given that the encoding rates are restricted by external system parameters, such as delay and channel constraints.

1.2.2 Other encoder functionalities for video transmission

In designing video codecs for video transmission applications, other factors related to the communication channel, such as the transmission delay and errors, need to be taken into account. MPEG-2 has potential use in video transmission applications, hence several types of scalability such as SNR, temporal and spatial temporal scalability are defined in the MPEG-2 standard [4, 16, 17]. In the recent development of the ITU Recommendation H.263 version 2 [7], which is the extension of H.263 and usually referred as "H.263+", several coding modes have been defined to provide more functionalities for real-time transmission: such as the "Reference Picture Selection mode" which uses a feedback acknowledgement from the receiver to inform the encoder if a picture has been degraded by transmission errors so other unaffected pictures can be used as the reference frame to predict following frames; the "Temporal, SNR and Spatial Scalability mode" in which the quality of the transmitted video can be adjusted according to the limitations of channel throughput and other conditions by selectively transmitting the enhancement data on top of the baseline-quality video data; and the "Independently Segmented Decoding mode" to confine the error propagation. Other modes such as "Slice Structured mode" defines the slice structure containing variable numbers of macroblocks. The use of slice structure allows more flexibility in sub-dividing the picture, which can be more suitable for the underlying packet transport, more efficient in error resilience and result in lower encoding delay.

1.3 Rate Control for Video Transmission

In the video communication system shown in Fig. 1.1, buffers are used in both the encoder and decoder to temporarily store the coded video data before the transmission or after the reception. In a CBR transmission environment, buffer between the video codecs and transmission channel can smooth out the variable source rate generate a constant output rate to the channel [18], whereas in a VBR transmission, buffer can function as "traffic shapper", which smoothes out the bursty video bitstream and alleviates the potential traffic burden of the transmission channel or network. In most video applications, the physical buffer sizes are limited, and a rate control mechanism is required to regulate the source encoding rate in order to prevent buffer overflow and underflow. In MPEG, a hypothetical Video Buffering Verifier (VBV) decoding model (MPEG-2 Annex C [4]) has been defined to constrain the source encoding rate. The rate control at the source coder constrains the video encoding rate in order to prevent overflowing or underflowing the hypothetical decoder buffer, which otherwise could be problematic for the decoder to receive and decode video data. It is usually desirable to keep a more consistent video quality for the encoded video sequence. Therefore in addition to prevent buffer overflow or underflow, another goal that can be achieve by rate control is to properly allocate bit among video frame or within a frame for consistent or better reconstructed video quality. The rate control algorithm Test Model 5 (TM5) [19] in MPEG uses the feedback from the buffer occupancy to modify the source encoding rate, and allocates bits to different frame types, namely I-, P-, and B-frame, proportionally such that the reconstructed video has more consistent video quality over frames with different frame types.

As discussed in previous section, in most video application a rate control mechanism is often required to constrain and regulate the encoding rate of the video, such that the bit rate of the compress video data stream can comply with external constraints that are imposed by the application and hardware. The importance of rate control is more evident in applications involving transferring video data in real time to the decoder through a communication channel, where video transmission is subject to the channel bandwidth and the *delay* constraints. While the major focus of rate control is on constraining the bit rate and avoiding the buffer overflow and underflow, the delay constraining factor is often overlooked by most rate control schemes. The rate constraints that are related to the delay constraints were formally formulated in [20]. An optimal rate control approach, in the sense of maximizing the video quality, for video transmission over CBR transmission environment with such delay constraint has been proposed in [21]. Extending this previous research, in this dissertation we study the rate control problems for video transmission over various VBR channels [22, 23, 24, 25]. Specifically we will focus our discussion on the video transmission over two types of VBR channels, namely broadband public access network such as ATM networks, and burst-error channel such as wireless links. First we briefly introduce the characteristics of the VBR channels which are considered in our study.

1.4 Transmission Channels

Due to the high acceptance and wide popularity of visual communication, many communication networks and channels have been investigated and studied as the platforms of video transmission. Traditionally CBR channels, such as circuit-switched

channels, dedicated connections with fixed transmission capacity, or a reserved bandwidth built on top of a packet-switching network [26], have been used for video transmission. The reason for using CBR channels for video transmission is that the guaranteed transmission bandwidth and bounded transmission delay, which are essential for real-time video transmission, can be more easily achieved. However, as the emerging broadband public access networks become more popular and accessible, these broadband networks with VBR transmission capability are regarded as a major platform for various video transmission application in the future. VBR transmission capabilities have the potential of bringing about substantial benefits for video transmission: constant or consistent video quality can be achieved through VBR video coding and thus high quality real-time applications with bounded transmission delay may become feasible with VBR transmission [22, 20, 27, 28]. VBR video transmission in a network environment can result in potential gains in network utilization, the so-called Statistical Multiplexing Gain (SMG), when multiple VBR video sources are multiplexed [29, 30]. Shorter transmission delay can be achieved by using VBR transmission instead of CBR transmission, because in a CBR system extra data buffering, which may cause extra delay, is required when the fixed channel capacity cannot accommodate the variations of video source bit rate [31].

In our research of video rate control, we focus on the following two types of transmission channels: (i) broadband networks with usage parameter control, for example transmission of Video over ATM, and (ii) unreliable channels with burst transmission errors, for example Internet or wireless channels. The characteristics of these two types of channels are described briefly in the following section. A more complete study of the rate control problem over these channels will be presented in Chapter 3 and Chapter 5.

1.4.1 Video transmission over broadband networks

Future public access networks are expected to provide transmission services for various forms of information, such as data, audio, and video. Asynchronous Transfer Mode (ATM) networks are an example of a network architecture designed for broadband applications in public networks. ATM networks have been increasingly used for real-time multimedia transmission [32, 33, 34] as their design is inherently applicable to provide fast and reliable transmission with predictable bandwidth provision and delay bounds.

The ATM Adaptation Layer (AAL) specification [35, 36] defines a set of connection-oriented VBR transmission services to be provided by the ATM networks, which are suitable for real-time video applications. In addition to the VBR transmission capability, real-time video transmission also requires guarantees on bounded transmission delay and delay jitter, the so-called Quality of Service (QoS). The QoS guarantees ensure that real time display of video at the decoding end is possible.

The QoS of the network can be maintained if the traffic flow in each connection is monitored and regulated by a "policing function" or "usage parameter control". Policing functions enforce traffic flow control and ensure that the source traffic flow from each user complies with the negotiated usage parameters. In order to efficiently monitor the arriving traffic and respond to any violation of usage parameters, most of the policing functions will be simple and very easy to implement [37, 38]. Examples of simple policing mechanisms include leaky bucket, double leaky bucket, jumping window, sliding window, etc., and will be briefly discussed in Chapter 3.

We will focus on the study of video transmission over ATM networks with policing functions that are monitored at the user-network interface and are known to the video encoder. Video encoding rate thus can be controlled by the encoder with

a desirable goal being to avoid the violation of any applicable policing function constraints.

1.4.2 Video transmission over burst-error channels

Research on video transmission over wireless channels has also become popular, because wireless connection can provide convenient tetherless data access and mobility to users. Video communication over wireless is challenging as the hardware and channel resources are very constrained. Given the bandwidth limitation of the wireless links, an efficient video compression is often required for wireless video applications. In addition, compared to other wired transmission channels, wireless links suffer from limited bandwidth and are more likely to see their performance degrade. The compressed video data stream is vulnerable to transmission errors due to the predictive coding and entropy coding schemes that are often used in video codecs. Therefore, reliable transmission of highly compressed video signals over error-prone wireless channels is the more significant issue of the wireless video transmission.

A special characteristic of transmission errors in the wireless channels is that errors tend to occur in bursts. Therefore error control techniques such as Forward Error Correction (FEC) would require large overhead in terms of redundancy to effectively correct all the clustered errors. However, such overhead is a waste of transmission bandwidth during the period when the channel is in good condition. Recent research has considered ways of improving the transmission reliability by making use of the feedback channel for "closed-loop" error control, including various forms of retransmission [39, 40, 41]. Using the feedback channel for error control has been accepted by ITU-T as part of the mobile extension of H.263 Recommendation [7]. Additional delay may be introduced in the feedback and retransmission

process, which may be problematic in real-time applications with critical delay constraints. Rate control in this delay-constrained video transmission over error-prone channels thus focuses on regulating the bit rate of video data stream to cope with the dynamically change channel conditions and meet the delay constraints. We will present a rate control scheme for robust video transmission over burst-error channel in Chapter 5.

1.5 Overview and Contribution

In Chapter 2 we formulate the problem of rate control in a video communications system by taking into account all the available resources and constraints. We introduce the concept of *effective buffer size* as the single constraint on encoding rate combines the system delay constraint and the channel bandwidth constraint. This effective buffer size can be seen as a convenient representation of the constraints the encoding rate has to meet in order to comply with the delay constraint and channel constraint, and will be the basis of our rate-control algorithms for video transmission.

In Chapter 3 we focus on video transmission over ATM networks. In this environment both source bit stream and transmission data flow are VBR, and the source encoding rate and channel transmission rate can be controlled by the rate control mechanism. By combining the delay and channel constraints as in Chapter 2, an algorithm based on dynamic programming is proposed to jointly select the video encoding rate and transmission rate which can comply with these imposed constraints and achieve minimum video distortion. This joint selection algorithm can also be used as a benchmark tool to provide a quantitative comparison of the video quality enhancement gains when using a VBR channel instead of a CBR one.

In Chapter 4 we consider a real-time rate control scheme for the applications in which video is captured and encoded just before transmission, a scenario that arises in applications such as video conferencing or videophone. Encoding rate constraints in this real-time system are formulated with the goal of delivering video information in time for decoding. Algorithms based on dynamic programming and Lagrangian optimization for optimal bit allocation are proposed. This real-time rate control approach is also the basic structure for the robust video transmission over unreliable channels introduced in Chapter 5.

In Chapter 5 we look at the problem of video communication over a burst-error channels. A new scheme which combines the rate control and the error control elements in a communication system is proposed to achieve robust video transmission over an unreliable channel. We focus on the "two-way" wireless channel where a feedback channel exists, and an error control scheme like Automatic Repeat reQuest (ARQ) can be used for data re-transmission. It can be shown that the overall robustness of video transmission can be improved through the source rate control assisted by an appropriate channel model and real-time feedback of the channel condition.

Conclusions of our research are summarized in Chapter 6.

Chapter 2

Video Transmission

2.1 Overview of Digital Video Applications

Nowadays, digital video is used in a wide range of multimedia and communication applications to disseminate information. Due to the diversity in the nature of those applications, the underlying system and hardware requirements may be quite different, even though most of the video transmission application share a similar structure as shown in Fig. 1.1. For example, high video quality is of prime importance in the system design for digital video storage applications (VCD, DVD) or HDTV applications, whereas low delay latency (to make interactive visual communication possible) is the key concern in video-conferencing applications. Therefore, in addition to the core video codec component, other system-level components [34], such as rate control, channel error control and networking interface, have to be adjusted to the specific needs of different applications. In order to get more insight of various constraining factors in a video transmission system, which are required in our research to develop the corresponding rate control approaches, we first classify video applications into the following three types (see Fig. 2.1) according to where

the source video data is initially located and how video data is transferred to the decoder:

I. **Pre-compressed video is available on a local storage device:**

In this type of applications, video data is retrieved from a local storage device. The bandwidth of the internal data bus is considered to be sufficiently large to handle the most demanding video data transfer rates. Examples of such applications include Video CD, Digital Versatile Disc (DVD) movies, or any video applications in which the data of the whole video sequence has been transferred to the local storage device before displaying the video sequence.

II. **Real-time transmission of pre-compressed video data:**

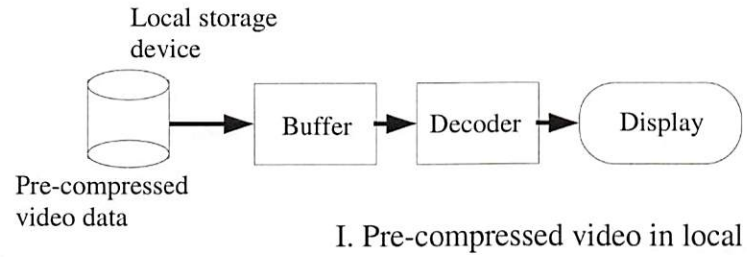
Video is pre-compressed and stored in a remote site. When the video displaying session begins, video data is transferred through a communication channel to the decoder followed by the decoding and display processes immediately or with a small decoder buffer delay. One important requirement for this real-time transmission is that the video data of each frame has to be received at the decoder before the time for decoding. Therefore there is a delay constraint in transferring the video data, and the applicable video encoding rate is also constrained by the communication channel bandwidth. In some applications a longer delay may be tolerable. In these applications the overall delay latency is only noticeable at the beginning of the video transmission session, i.e., the time it takes to display the first video frame. Examples of such applications are video on demand or broadcast of pre-recorded program.

III. **Real-time encoding and transmission**

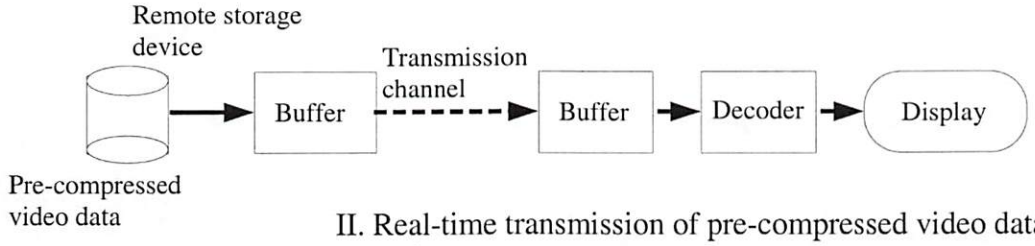
In this type of application, video data is captured, encoded, and delivered to

the encoder in real-time just before transmitted to the destination. The overall delay latency thus begins from the time that a video frame is captured and encoded till the time that video frame is decoded and displayed. In the particular case of interactive applications, such as video conferencing and videophone, a short end-to-end delay is required. This is the case where delay constraint is much more restricted than that in the two previous cases. However in some examples such live TV or video broadcast, the delay constraints is not so restricted although video data is also encoded and transmitted in real-time. In these cases the delay requirement is more like of the type II application mentioned above, and the delay latency is perceived as the initial setup before the video session.

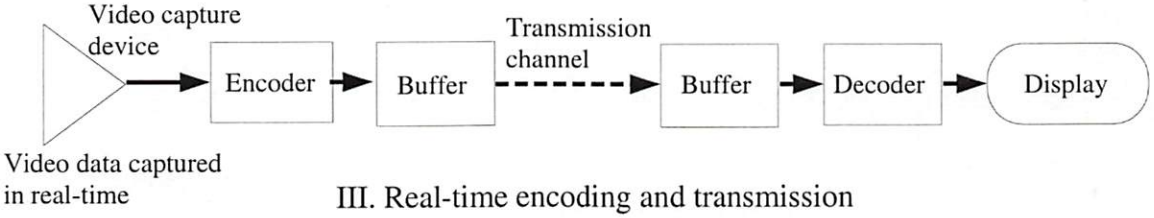
In type I applications, video data is retrieved from the local storage device. There is no bottleneck in the data transfer between the video source and the decoder. The only possible constraining factor is the capacity of the storage device. Therefore the main purpose of the rate control in this type of applications is to meet the total bit budget, and to distribute bits evenly for encoding each video frame. In type II and III applications video data is streamed into the decoder through a communication channel in real time and is immediately decoded and displayed. We will refer to this type of data transfer as "real-time video transmission" in the later discussion. In these cases a large local storage device, as that in type I applications, is no longer needed. Instead only a small decoder buffer is required to temporarily store the arriving video data. However, other problems arise such as whether the channel bandwidth is large enough to support video transmission, or whether video data can be delivered in time for decoding and displaying. Since a communication channel is used to transmit the video data, the reliability and transmission delay of the channel



I. Pre-compressed video in local storage device



II. Real-time transmission of pre-compressed video data



III. Real-time encoding and transmission

Figure 2.1: Different types of video applications

also affects the effectiveness of real-time video transmission. Therefore, rate control in these types of applications should aim to regulate the bit-rate of video data stream to comply with those additional constraints that are imposed by the communication channel and the real-time requirement of the applications.

2.2 Delay Latency and Delay Constraint in Video Transmission

As discussed above, delay constraint for the transport of video data is one of the major reasons for requiring a rate control mechanism at the video encoder. To formulate the delay constraint, first we define the *end-to-end delay*, denoted as ΔT, as the time interval elapsed from the time a video frame is captured and encoded at the transmitter until the time when the video frame is decoded and displayed at the receiver. In a real-time video communication system where both encoder and decoder are attached to synchronous devices (camera and display, respectively), the end-to-end delay of a video frame traversing the system should be constant. Video data that arrives at the decoder too late to be decoded by its scheduled display time is useless and is considered lost. Clearly, frame skipping at the decoder results in quality loss, especially when motion compensated video coding is used, while skipping frames a the encoder can be done without as heavy a quality penalty. However in this latter case there will also be end-to-end delay constraints for those frames that are transmitted. Different video applications have different delay requirements. In interactive video communications (e.g., video conferencing) low delay is required, while in one-way video transmission (e.g., broadcast or video on demand) the end-to-end delay is only noticeable to the user as an initial latency, i.e., the time interval between the start of the video transmission session and the time the first video frame is displayed. Frame loss may result if some of the information corresponding to a video frame arrives at the decoder after the scheduled decoding time. Since the information received at the decoder is stored in the decoder buffer before actually being decoded, we will call this situation *decoder buffer underflow.*

For a generic video communication system, the end-to-end delay ΔT may consist of the following major components (refer to Fig. 2.2) as:

$$\begin{aligned} \Delta T &= \Delta T_e(\text{Encoder delay}) \\ &+ \Delta T_{eb}(\text{Encoder buffer delay}) \\ &+ \Delta T_c(\text{Channel transmission delay}) \\ &+ \Delta T_{db}(\text{Decoder buffer delay}). \\ &+ \Delta T_d(\text{Decoder delay}). \end{aligned} \tag{2.1}$$

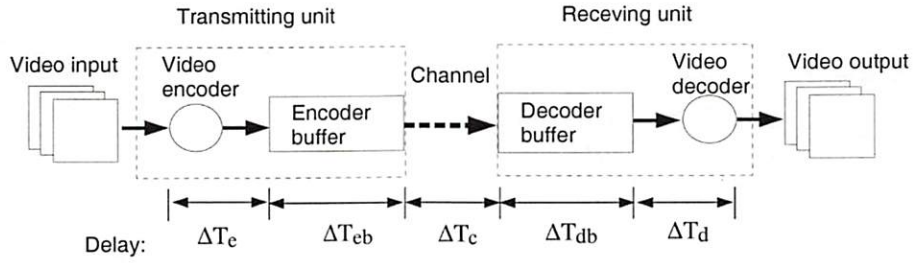


Figure 2.2: Delay components of a video communication system.

The end-to-end delay ΔT in (2.1) has to be constant for the transmission of every video frame in order to keep synchronized timing between encoder and decoder. In most video codecs, a bounded maximum encoder and decoder delay is more likely to be expected, because the design of the encoder and decoder has to meet the minimum performance requirement that every frame can be encoded or decoded within each frame interval. We thus assume a constant encoder encoder delay ΔT_e and the decoder delay ΔT_d in our formulation.

The channel delay ΔT_c, however, could be variable in some transmission environments. In the point-to-point communication channel with direct connection, the transmission delay ΔT_c can be treated as fixed because of the fixed delay latency

in the transmission link. However in public access networks, it may be difficult for the networks to provide transmission services with fixed transmission delay due to the possible network congestion caused by unpredictable increases in traffic load. Therefore, we over-dimension the variable transmission delay ΔT_c by its maximum expected value ΔT_c^{max}, given that the channel can provide data transmission service with bounded transmission delay.

Excluding the delay elements ΔT_e, ΔT_d, and ΔT_c^{max} from the constant end-to-end delay ΔT in (2.1), we can focus on the buffer constraint due to the total buffer delay (at encoder and decoder) as:

$$\Delta T_{eb} + \Delta T_{db} = \Delta T - \Delta T_c^{max} - \Delta T_e - \Delta T_d \tag{2.2}$$

Defining the time interval for one video frame as T_f, then the number of video frames that are stored in either encoder or decoder buffer is

$$\Delta N = \frac{\Delta T_{eb} + \Delta T_{db}}{T_f} = \frac{\Delta T - \Delta T_c^{max} - \Delta T_e - \Delta T_d}{T_f}. \tag{2.3}$$

What (2.3) states is that given a end-to-end delay constraint ΔT, there will be ΔN video frames stored in both the encoder and decoder buffer. ΔN can also be interpreted as the buffer delay in terms of the number of frames.

From (2.2) if ΔT_e, ΔT_d, and ΔT_c^{max} are constant, then ΔN will also be constant. However we over-dimension the channel delay ΔT_c by its maximum expected value ΔT_c^{max} in above formulations. If the actual channel delay ΔT_c is smaller than ΔT_c^{max}, video data will arrive at the decoder buffer earlier. In this scenario, more data (or video frames) will be stored in the decoder buffer than that in our formulations, but the compliance with the end-to-end delay constraints can be ensured by assuming

the worst cast transmission delay, if the decoder buffer size is also over-dimensioned to accommodate those data that arrive earlier.

Note that in what follows we consider that the encoder and decoder clocks are shifted by an amount equal to ΔT_c^{max}, the transmission delay. Thus, if the i-th frame interval starts at time t_i at the encoder, it will start at time $t_i + \Delta T_c$ at the decoder. Refer to Fig 2.3.

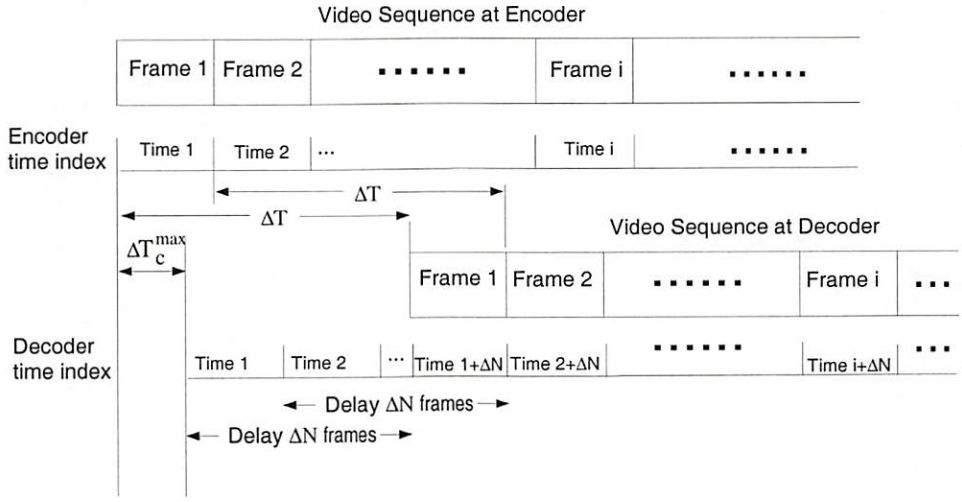


Figure 2.3: Timing diagram of encoder and decoder, note that the clocks at encoder and decoder are shifted by ΔT_c the transmission delay.

2.3 Encoding Rate Constraints Related to Delay Constraint

We will now discuss how the video encoding rates are constrained by the end-to-end delay, channel transmission rate, and the sizes of the buffers at the encoder and decoder. Denote $B^e(i)$ and $B^d(i)$, respectively, the encoder buffer and decoder buffer occupancies at time i. Let $C(i)$ and $R(i)$ be, respectively, the channel rate at

time i (i.e., during the i-th frame interval) and the source rate used for i-th frame. The formulation does not change if one considers a time unit smaller than the frame interval. For example, as will be seen in our experiments, we can use the Group of Blocks (GOB) as the basic unit so that $C(i)$ and $R(i)$ become, respectively, the channel and source rates for the i-th GOB. All of these variables have units of bits. The encoder and decoder buffer occupancies can be obtained as

$$B^e(i) = \sum_{j=1}^{i} R(j) - \sum_{j=1}^{i} C(j) \tag{2.4}$$

$$B^d(i) = \begin{cases} \sum_{j=1}^{i} C(j) - \sum_{j=1}^{i-\Delta N} R(j), & \text{when } i \geq \Delta N \\ \sum_{j=1}^{i} C(j), & \text{when } i < \Delta N. \end{cases} \tag{2.5}$$

Note that, because of the end-to-end delay, the decoder waits ΔN frame intervals before starting to decode the video frames available in its buffer. Here we are assuming sufficiently large physical buffers at encoder and decoder and thus we focus on the effect of buffer underflow. For most cases of interest, given that our goal is to minimize distortion, the set of quantizers will be such that encoder buffer underflow will not occur and we thus do not take this into account. However *decoder buffer underflow*, i.e., the situation where not all the bits corresponding to a given frame are present at the decoder in time to be decoded, may very well occur. This problem is important since it results in frame losses.

Combining the encoder buffer occupancy (2.4) at time i and decoder buffer occupancy (2.5) at time $i + \Delta N$, we have that:

$$\begin{aligned} B^d(i+\Delta N) &= \sum_{j=1}^{i+\Delta N} C(j) - \sum_{j=1}^{i} R(j) = \sum_{j=i+1}^{i+\Delta N} C(j) - (\sum_{j=1}^{i} R(j) - \sum_{j=1}^{i} C(j)) \\ &= \sum_{j=i+1}^{i+\Delta N} C(j) - B^e(i), \end{aligned} \tag{2.6}$$

In order to prevent the decoder buffer from underflowing, we have to keep the right hand side of (2.6) always greater than zero, i.e.,

$$\sum_{j=i+1}^{i+\Delta N} C(j) - B^e(i) \geq 0, \quad \forall i \tag{2.7}$$

We introduce the concept of *effective buffer size*, $B_{eff}(i)$, which we define as the maximum admissible level of buffer occupancy that the encoder can reach at time i such that the channel rates are adequate to transport all the bits without violating the end-to-end delay constraint (i.e., without producing decoder underflow.) From (2.6) the maximum level of encoder buffer occupancy is $\sum_{j=i+1}^{i+\Delta N} C(j)$, therefore we have

$$B_{eff}(i) = \sum_{j=i+1}^{i+\Delta N} C(j) \tag{2.8}$$

so that the effective buffer size is equal to the sum of the future channel rates during next ΔN frame intervals. We can guarantee that if the encoder buffer fullness $B^e(i)$ is always smaller than $B_{eff}(i)$, then the decoder buffer will not underflow. In the special case of a CBR channel, the channel rate is constant, i.e., $C(i) = C$, $\forall i$. Then the effective buffer size is

$$B_{eff}(i) = \Delta N \cdot C \tag{2.9}$$

which is also constant. Note that the buffer sizes at encoder and decoder should be the same in this case.

Although the physical size of the buffer can be very large, the actual buffer size that the encoder can use is constrained by the end-to-end delay; thus, the *effective* buffer size can be potentially smaller than the physical buffer size. This concept is also useful in situations where the encoder has no control over the (variable) channel

rates but seeks to avoid decoder underflow [42], as will be seen in Chapter 4 and Chapter 5.

2.4 Physical Buffer Constraint

Let B^e_{max} and B^d_{max}, respectively, be the physical encoder and decoder buffer sizes, and assume that the end-to-end delay ΔN is a design parameter. In this section we study how the sizes of physical buffers at encoder and decoder affect other elements of the encoding system.

2.4.1 Constraints on the encoder buffer state

If the physical buffer size is smaller than $B_{eff}(i)$, then the constraints on the encoder buffer fullness become

$$0 \leq B^e(i) \leq B^e_{max}, \tag{2.10}$$

and, similarly, from (2.6), the conditions to prevent decoder underflow/overflow are

$$0 \leq \textstyle\sum_{j=i+1}^{i+\Delta N} C(i) - B^e(i) \leq B^d_{max} \tag{2.11}$$

$$\text{or} \quad \textstyle\sum_{j=i+1}^{i+\Delta N} C(j) - B^d_{max} \leq B^e(i) \leq \sum_{j=i+1}^{i+\Delta N} C(j) \tag{2.12}$$

These two constraints can be combined into a single one:

$$\max(\sum_{j=i+1}^{i+\Delta N} C(j) - B^d_{max}, 0) \leq B^e(i) \leq \min(\sum_{j=i+1}^{i+\Delta N} C(j), B^e_{max}) \tag{2.13}$$

From (2.13), if either the physical encoder or decoder buffer sizes are smaller than the effective buffer size, then the applicable buffer occupancy is not only upper bounded by the physical buffer size, but also lower bounded by a minimum buffer occupancy which may not be zero.

2.4.2 Constraints on the allowable channel rate

An intuitive interpretation of (2.13) is to say that the channel rates have to be sufficiently low so that information does not arrive too fast to the (small) decoder buffer; this explains the lower bound on buffer occupancy at the encoder. From (2.11) we have

$$B^e(i) \leq \sum_{j=i+1}^{i+\Delta N} C(j) \leq B^d_{max} + B^e(i) \tag{2.14}$$

which states that limited buffer sizes at encoder and decoder can actually impose a constraint on the range of channel rates that can be used. Thus, even if network policing imposes no restrictions on the admissible channel rates, arbitrary rates may not be possible for a given choice of buffer sizes at encoder and decoder.

2.4.3 Physical buffer size selection

In the next chapter, we will discuss the problem of rate control for video transmission over an ATM networks with policing function constraint. Obviously in the ATM network with policing function constraint, a good choice of buffer sizes at encoder/decoder for given policing function parameters would be one such that the buffers are sufficiently large that they do not introduce any additional constraints on the channel rates. The physical buffer sizes constraint both the channel rates as

in (2.14), and the encoder buffer fullness as in (2.13). Assume the end-to-end delay and policing function parameters are given and define

$$C_{max} = \max \sum_{j=i+1}^{i+\Delta N} C(j) \quad \forall i \tag{2.15}$$

as the maximum aggregate channel rate that can be allowed by the given channel policing function over a ΔN-frame interval. In the Leaky Bucket (LB) policing function [37, 38], for example, user traffic is monitored by an imaginary counter that fills up at the transmission rate $C(i)$ and empties at the constant output rate $\bar{C}$. A transmission rate $C(i)$ which makes the counter greater than the pre-specified maximum value LB_{max} is thus inadmissible. The highest aggregate channel rate over any ΔN-frame period is $LB_{max} + \Delta N \cdot \bar{C}$. If the physical encoder and decoder buffer sizes are larger than C_{max}, i.e.

$$B^e_{max} \geq C_{max}, \quad B^d_{max} \geq C_{max} \tag{2.16}$$

then the condition $\sum_{j=i+1}^{i+\Delta N} C(j) \leq B^d_{max} + B^e(i)$ in (2.14) will always hold, and the following relations will always be true:

$$\sum_{j=i+1}^{i+\Delta N} C(j) \leq B^d_{max}, \quad \sum_{j=i+1}^{i+\Delta N} C(j) \leq B^e_{max} \tag{2.17}$$

and therefore the encoder buffer is only constrained by the effective buffer size, and there are no additional constraints due to the encoder/decoder physical buffer sizes.

2.5 Summary

In this chapter, we have discussed the delay elements in a generic video transmission system and the delay constraints for video transmission. The constraints on encoding rate are formed by taking into account the applicable end-to-end delay and channel transmission rate. In addition to the physical buffer constraints, we introduced the concept of "effective buffer size" to represent the constraints imposed by the delay and the channel transmission rate. In the next chapter, we will show an example of introducing this rate constraint in the context of video transmission over an ATM network with policing function constraints.

Chapter 3

Joint Selection of Source and Channel Rate for VBR Video Transmission

3.1 Introduction

The VBR nature of encoded video data has been cited [43, 28] for a number of years as a motivation for establishing networks that allow video transmission at variable rate. Because the real-time nature of video transmission places a special requirement on the delay constraint, the networks that support VBR video transmission will have to provide transmission service with specific bounds on the end-to-end delay and delay jitter: the Quality of Service (QoS) guarantees. Asynchronous Transfer Mode (ATM) networks are an example of the network architecture which would allow this type of VBR transmission with QoS guarantees, as they incorporate flexible mechanisms for resource allocation [44, 45].

However, the design of the network resource allocation scheme becomes very challenging when VBR video data are transmitted through the network [46, 47]. The video transmission is delay-sensitive, thus dedicated network resources haves to be reserved for the transmission of video in order to ensure the timely data

reception at the decoder. On the other hand, the unpredictable bursty nature of video data traffic usually has significant impact on the overall network service [20, 31]. Realistic VBR transmission environments will certainly impose constraints on the rate that each source can submit to the network in order to ensure that the service quality of the entire network is maintained. Therefore, before the start of each transmission session, the network will likely have to negotiate with the user a set of traffic parameters, such as average rate, peak rate, and the peak rate duration. Once the traffic parameters for the transmission session are set up, a policing mechanism is required to monitor the input data traffic and enforce the traffic parameters on the data transmission. While not precluding the use of other policing approaches, we concentrate here on policing functions *that are known to the video encoder and are monitored at the user-network interface* [48]. Policing functions such as leaky bucket (LB), jumping window, sliding window, etc, are commonly used due to their simplicity and efficiency [37, 38]. Note that our methods would also be directly applicable to shaping for ITU-T defined parameters such as Peak Cell Rate and Sustainable Cell Rate.

In this chapter, we focus on the discussion of VBR video transmission over ATM networks with usage parameter control. Video data is read at some rate from the source and has to be played back in real-time, or with some small buffering delay, at the destination. Previous studies [20, 22, 49] have shown that video transmission is subject to both end-to-end delay constraints and channel rate constraints. The explicit formulation of encoding rate constraints for delay-constrained video transmission has been established in previous chapter. (refer to Section 2.3 and 2.4 in Chapter 2). Various heuristic methods for selecting both source and channel rates have been considered and proposed [20, 50, 51, 52, 53] to achieve the best, or at

least good video quality under those constraints. However, little has been done on the problem of how to jointly allocate channel bit rate and encoder bit rate subject to the buffer constraint and channel constraint.

The novelty in our approach toward the optimal rate control is that we explicitly formulate the delay constraint and channel constraint in a VBR channel with policing function constraint, and propose a new algorithm to find the optimal selections of source and channel rates jointly for most policing functions of interest [49, 22]. Independently other researchers have studied a similar formulation and provided an alternative algorithm based on a "sliding window" approach [54]. Joint control of encoder and channel rates is also considered in [55] which focus on controlling the instantaneous and sustainable rate on the real-time basis. When using a sliding window, global optimality can no longer be guaranteed, but on-line encoding becomes a possibility.

Specifically speaking, we propose an algorithm that aims to maximize the video quality by jointly selecting the source rate (number of bits used for encoding a given frame) and the channel rate (number of bits transmitted during a given frame interval). These rate selections are subject to two sets of constraints: (i) the end-to-end delay for the real-time video transmission as discussed in Chapter 2, and and (ii) the transmission rate constraints that are imposed by the usage parameter control. Note that while other researchers have considered shaping techniques for bitstreams generated by video coders, here we go one step further by introducing the shaping requirements, in the form of policing constraints, *within the video encoder loop*. We are thus able to adjust the quality of the encoded video in a rate-distortion optimal manner while complying with required shape parameters. Further shaping

can be also performed at the output of the encoder buffer without affecting the video quality.

Our algorithm allows us to present a global picture of a VBR video transmission system by considering the trade-offs among the available resources, namely, end-to-end delay, policing function parameters and physical buffer sizes at encoder/decoder. We can thus compute the maximum achievable video quality for a given video sequence with a variety of system configurations. From our experimental results, we also show that, for the specific environment of an ATM network with LB constraints, the optimization problem is equivalent to the constant bit rate (CBR) channel with a single buffering constraint, whenever sufficient physical buffering is available. Our methods can be used for off-line encoding, for benchmarking and also to derive approximate allocation algorithms which can operate under real-time encoding.

3.2 Optimal Rate Control

In the previous introduction to digital video and video transmission in Chapter 1, we pointed out that rate control is a necessary element to ensure that the video transmission complies with applicable channel bandwidth and delay constraints. In addition, a proper rate control scheme can also function as a bit allocation scheme to distribute the available bit budget to each video frame in the sequence in a way to maximize the "video quality". However, the definition of the quality measure can be quite different in each application. The most direct measure, which is also the one we used in our research, is the total distortion, i.e., the sum of the distortion of every video frame in the video sequence. Other video quality measures may aim, for example, at taking into account the variation in distortion between consecutive

frames so that the total distortion may be weighted by a measure of distortion variations. This approach has been adopted for example in [56].

Following the notations that used in Chapter 2, denote $R(i)$ as the encoding rate (the number of bits used in encoding) for frame i. In addition we define $D(i)$ as the distortion of frame i when $R(i)$ bits are used in coding frame i. The goal of rate control is defined as to minimize the overall distortion of all the video frames in a video sequence. In other words, for a video sequence with N video frames which are indexed from 1 to N, we seek to allocate $R(i)$, $i \in \{1, \ldots, N\}$, so that the accumulated distortion $\sum_{i\in\{1,\ldots,N\}} D(i)$ is minimized, that is:

$$\min_{R(i),i\in\{1,\ldots,N\}} \sum_{i\in\{1,\ldots,N\}} D(i), \tag{3.1}$$

where $R(i)$, $i \in \{1, \ldots, N\}$, are subject to rate constraints.

3.3 Optimal Encoder and Channel Rate Allocations

Based on the rate constraints that are derived from the delay and buffer constraints as discussed in Chapter 2, we now introduce a technique that can find the optimal operating encoding rates on a given input video sequence. The proposed algorithm use the Dynamic Programming (DP) approach to search for a set of selections of encoding rates and transmission rate that maximize the video quality for a given set of delay and channel constraints. Conceptually we construct a trellis in which each branch represents a candidate selection of encoding rate and transmission rate for each video frame. Therefore the propose technique is complex and may not be suitable for real time implementation but can be used for benchmarking or off-line encoding.

The proposed approach is extended from the previous work [21], in which a similar DP approach is used to select the encoding rate with buffer constraint in a CBR transmission environment. Therefore, we start by briefly overview on the encoding rate allocation based on DP in the CBR transmission scenario in Section 3.3.1 followed by the detailed discussion for the joint rate allocation algorithm in Section 3.3.2.

3.3.1 Optimal rate control for CBR transmission

Given a discrete set of quantizers in the video coder and a buffer constraint, the DP technique is used to find the optimal allocation of encoding rates for video transmission over CBR a channel [21]. In this formulation, assuming a sufficiently long encoding delay, a trellis can be formed where each branch represents a choice of quantization for the frame and has associated a distortion, refer to Fig. 3.1. The total distortion of a given path can be found by adding up the distortion of each of the branches comprising the path. The trellis path with minimum total distortion can be found using the Viterbi algorithm (VA) [57], a form of deterministic dynamic programming similar to Dykstra's shortest path algorithm, when the length of a path is determined by the amount of distortion added if that path is taken. The trellis state represents the buffer occupancy and thus each path in the trellis represents a possible solution. The basic idea is to simplify the search by eliminating the suboptimal paths, namely, those paths that overflow the buffer or those paths that reach a given node of the trellis with a cost higher than that of the minimum cost path at that node. Thus, as shown in Fig. 3.1 if two paths result in the same encoder buffer fullness only the one with minimum cumulative distortion up to that stage is kept. This approach still yields the globally optimal solution since both

paths are equivalent as far as the rest of the stages are concerned and thus the one having higher distortion up to the intermediate stage is also sure to result in higher distortion overall.

Note that when growing these trellises the number of states (possible buffer fullness levels) can become quite large and the set of paths can be relatively sparse compared to the number of states (thus, logically, few paths would be pruned since the paths would be unlikely to meet). In [21] it was shown that in general the granularity of the buffer states can be made coarser without affecting the result of the optimization. Thus we can consider buffer states spaced in, say, 100 or 1000 bit intervals, rather than spaced by just one bit. In this scenario paths are made to converge to the nearest buffer state. Both the error in rate and the incurred sub-optimality are minimal. The same approach will be used in the next section to quantize the channel rate levels.

3.3.2 Optimal rate control for VBR transmission

The dynamic programming method above can be extended to find the optimal encoder and channel bit-allocation jointly in the VBR channel environment. Our goal is to choose the number of bits to use for each frame in the sequence *and* the number of bits that the channel should transmit for each time slot (i.e., each frame interval), such that (i) the total video distortion is minimized and (ii) any applicable policing constraints are met.

Suppose we are given M possible quantizers for each frame, and P possible channel rates for each frame interval. Define $\mathbf{x} = \{x(1), x(2), ...x(N)\}$ as the sequence of quantizer choices, where $x(i) \in \{1, \ldots, M\}$ is the quantizer index for frame i. The number of bits generated is $R_{x(i)}(i)$ and the associated distortion is

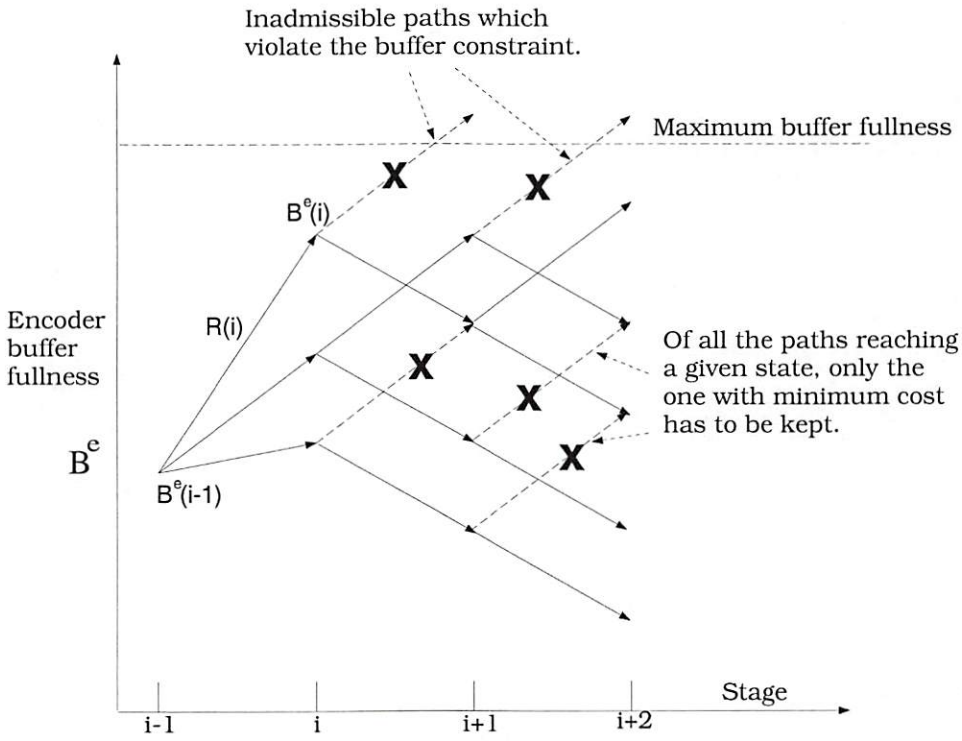


Figure 3.1: Buffer constrained optimization in the CBR channel case.

$D_{x(i)}(i)$. Also define $\mathbf{y} = \{y(1), y(2), ..., y(N + \Delta N)\}$ as the sequence of channel rate choices, where $y(i) \in \{1, \ldots, P\}$ is the index of channel rate, and the associated channel rate is $C_{y(i)}(i)$. Therefore, $\left\{R_{x(1)}(1), R_{x(2)}(2), ..., R_{x(N)}(N)\right\}$ and $\left\{D_{x(1)}(1), D_{x(2)}(2), ..., D_{x(N)}(N)\right\}$ are, respectively, the rate and distortion for each frame for a given choice of $\mathbf{x}$, $\left\{C_{y(1)}(1), C_{y(2)}(2), ..., C_{y(N+\Delta N)}(N + \Delta N)\right\}$ represents the channel rates for each frame interval for given $\mathbf{y}$.

Because there is an end-to-end delay ΔN between encoder and decoder, the decoder is actually decoding the $(i - \Delta N)$th frame at time i. We will have to select at any instant $C_{y(i)}(i)$ and $R_{x(i-\Delta N)}(i - \Delta N)$ to prevent decoder buffer underflow, i.e.,

$$B^d(i) = B^d(i-1) + C_{y(i)}(i) - R_{x(i-\Delta N)}(i - \Delta N) \geq 0 \tag{3.2}$$

In addition to the delay constraint, $C_{y(i)}(i)$ is also constrained by a policing function. Policing is implemented by keeping track of the bit rate transmitted through the channel by means of a monitor function and then imposing constraints on the allowable state of the monitor function. Define $L(i)$ as the state of the monitor function. For most cases of interest the change in the state of the monitor function will only depend on the previous state $L(i-1)$ and the choice of channel rate $C_{y(i)}(i)$, i.e.,

$$L(i) = \mathcal{F}\left(C_{y(i)}(i), L(i-1)\right) \tag{3.3}$$

The policing function decides whether to admit the data with bit rate $C_{y(i)}(i)$ into the network according to a criteria:

$$\text{if } L(i) = \mathcal{F}\left(C_{y(i)}(i), L(i-1)\right) \in \mathcal{L}, \text{ admit } C_{y(i)}(i) \text{otherwise, reject } C_{y(i)}(i) \tag{3.4}$$

where $\mathcal{L}$ represents the admissible region of the policing function $L(i)$. Refer to Fig. 3.2 for an example of such a system. We first introduce the general formulation and then show several examples of policing functions that can be seen to fit within this framework.

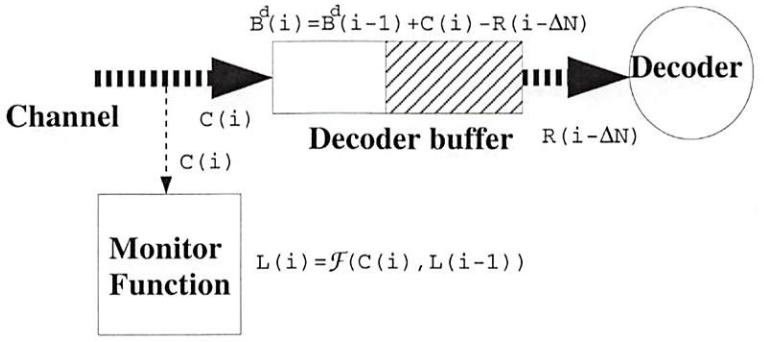


Figure 3.2: Decoder buffer in the receiver end.

In the general case, (3.2) and (3.4) jointly constrain the admissible buffer state transitions from $B^d(i-1)$ to $B^d(i)$, and the monitor function transition from $L(i-1)$ to $L(i)$. Therefore, the joint encoder and channel rate allocation can be formulated as:

Formulation 1 *Find mappings* $\mathbf{x} : (1, ...N) \rightarrow (1, ..., M)$ *and* $\mathbf{y} : (1, ..., N+\Delta N) \rightarrow (1, ..., P)$ *that solve:*

$$\min \sum_{i=1}^{N} D_{x(i)}(i) \tag{3.5}$$

subject to the constraints:

$$B^d(i) \geq 0 \tag{3.6}$$

$$L(i) \in \mathcal{L}, \quad \forall i = 1, ..., N + \Delta N. \tag{3.7}$$

Therefore the channel rates $C_{y(i)}(i)$ and the encoder bit rate $R_{x(i-\Delta N)}(i - \Delta N)$ should be allocated jointly to avoid decoder buffer underflow and meet (3.6) and (3.7).

We use the Viterbi algorithm (VA) [58, 57] to find out the admissible solutions $\mathbf{x}$ and $\mathbf{y}$, which have the minimum overall distortion. A trellis with $N + \Delta N$ stages is formed where each state in stage i represents a decoder buffer fullness and a monitor function value. Therefore each node in the trellis is defined by its state pair $(B^d(i), L(i))$. Each branch links two nodes and represents a transition in decoder buffer and monitor function states from stage i to stage $i + 1$. Thus each branch corresponds to a choice of channel rate $C_{y(i)}(i)$ at time i and a choice of quantization rate $R_{x(i-\Delta N)}(i - \Delta N)$ for the $(i - \Delta N)$th video frame. The new state, $(B^d(i + 1), L(i + 1))$ can be obtained using (3.2) and (3.3). The transitions that violate the

constraints given by (3.6) and (3.7) can be avoided by discarding the corresponding branches. Each branch also has associated the distortion corresponding to coding a video frame with the chosen quantization bit rate. Thus a path, which consists of one branch for every stage, represents one possible solution of encoder and channel allocation, and the trellis grown in this manner represents all the possible admissible solutions.

The VA reduces the optimal path search complexity by keeping only one trellis path for each node of the trellis, namely the one with minimal distortion up to that node. We need to prove that paths can be pruned without eliminating the optimal solution.

Lemma 1 *For all the possible quantizer choices* $x(1), x(2), ..., x(i-\Delta N)$ *and channel rate choices* $y(1), y(2), ..., y(i)$ *which have the same buffer fullness* $B^d(i)$ *and monitor function value* $L(i)$ *at time* i*, only the one with the smallest aggregate distortion* $\sum_{j=1}^{i-\Delta N} D_j(j)$ *can be a candidate for the optimal overall solution. The other paths with higher distortion are sure to be sub-optimal.*

Proof: From (3.2), we have that $B^d(i+1)$ only depends on $B^d(i)$, $C_{y(i+1)}(i+1)$ and $R_{x(i+1-\Delta N)}(i+1-\Delta N)$ no matter how $x(1), x(2), ..., x(i-\Delta N)$ and $y(1), y(2), ..., y(i)$ are chosen. From (3.3), we have that $L(i+1)$ only depends on $L(i)$ and $C_{y(i+1)}(i+1)$, no matter how $y(1), y(2), ..., y(i)$ are chosen. Therefore, $B^d(i)$ and $L(i)$ completely summarize the state of the system and two different choices of $x(1), \ldots, x(i-\Delta N)$ and $y(1), \ldots, y(i)$ are completely equivalent as far as the rest of the sequence is concerned if they result in the same $B^d(i)$ and $L(i)$. Thus a path can be discarded in favour of another path with same parameters and lower distortion without loss of optimality. □

Note that it is also possible to perform pruning if there exists a dominant path. For example, if two paths A and B have the same distortion, but B has used more bits so far, it is possible to prune it out since A has achieved the same distortion with fewer bits.

Before giving the details of the algorithm, we provide examples of policing mechanisms which would fit into the class considered here. Note that these mechanisms are representative of the most popular approaches considered in the literature.

Example 1: Leaky Bucket In the VBR channel with leaky bucket constraints [37] the policing function keeps an imaginary buffer with input rate $C_{y(i)}(i)$ and constant output rate $\bar{C}$. A channel rate $C_{y(i)}(i)$ which causes the imaginary buffer to overflow is thus inadmissible. In this case the state variable $L(i)$ is the leaky bucket state at time i, $LB(i)$, which can be written as:

$$LB(i) = \max(LB(i-1) + C_{y(i)}(i) - \bar{C}, 0) \tag{3.8}$$

The criteria for admissibility of channel rate $C_{y(i)}(i)$ is:

$$\text{Admit } C_{y(i)}(i) \text{ if } \quad LB(i) \leq LB_{max}, \tag{3.9}$$

where LB_{max} is the maximum size of the imaginary buffer.

Example 2: Double leaky bucket The double leaky bucket policing mechanism [59, 31] simultaneously uses two leaky buckets with different set of parameters (drain rate and bucket size). In this case the monitor function state is uniquely defined by $DLB(i) = (LB_1(i), LB_2(i))$ a state variable with two components, which are the

states of the two leaky buckets. The rate is admissible if neither of the two leaky bucket constraints is violated, i.e.

$$\text{Admit } C_{y(i)}(i) \text{ if } \quad LB_1(i) \leq LB1_{max} \text{ and } LB_2(i) \leq LB2_{max} \tag{3.10}$$

where $LB1_{max}$ and $LB2_{max}$ are the respective maximum sizes of the imaginary buffers.

Example 3: Jumping window A jumping window constraint is such that the rate over intervals of W time units cannot exceed a given maximum value, R_{max} [37]. In this case the state variable to be used is $JW(i)$ which can be defined as

$$JW(i) = \begin{cases} C_{y(i)}(i) & \text{if } i = k \cdot W \text{ for some integer } k, \\ JW(i-1) + C_{y(i)}(i) & \text{otherwise} \end{cases} \tag{3.11}$$

and therefore the constraint is simply that

$$JW(i) \leq R_{max}. \tag{3.12}$$

Example 4: Sliding window In this scheme, the policing function monitors the channel transmission rates of the most recent W time units [38]. In this case a similar state variable formulation can be used but it results in a more complicated state than in the other examples. Here, assuming a sliding window of size W, we would need to define the state variable $SW(i)$ as

$$SW(i) = \Big(C_{y(i)}(i), C_{y(i-1)}(i-1), \ldots, C_{y(i-W+1)}(i-W+1)\Big) \tag{3.13}$$

so that the constraint is

$$\sum_{k=0}^{W-1} C_{y(i-k)}(i-k) \leq R_{max}. \tag{3.14}$$

3.3.3 Optimization algorithm

With **Lemma 1** and the constraints (3.6) and (3.7), sub-optimal and inadmissible solutions can be pruned out in every intermediate stage. The algorithm can be described as follows:

Algorithm 1 *Joint encoder and channel bit-allocation by Viterbi algorithm:*

Step 0: *Initialize the decoder buffer fullness $B^d(0)$ and monitor function $L(0)$. Each node in the trellis at stage i is defined by a pair $(B^d(i), L(i))$. Start the loop with $i = 1$.*

Step 1: *At stage i, add all possible branches to the end of every surviving path node $(B^d(i-1), L(i-1))$ at stage $i-1$. The new state is $(B^d(i), L(i))$, obtained as:*

$$\begin{aligned} B^d(i) &= \begin{cases} B^d(i-1) + C_{y(i)}(i), & \text{when } i \leq \Delta N, \\ B^d(i-1) + C_{y(i)}(i) - R_{x(i-\Delta N)}(i-\Delta N), & \text{when } i > \Delta N \end{cases} \\ L(i) &= \mathcal{F}\left(C_{y(i)}(i), L(i-1)\right) \in \mathcal{L} \\ & \quad \forall x(i) \in \{1, ..., M\}, \quad \forall y(i) \in \{1, ..., P\} \end{aligned}$$

where $C_{y(i)}(i)$ and $R_{x(i-\Delta N)}(i-\Delta N)$ are such that constraints (3.6) and (3.7) are not violated. Refer to Fig. 3.3.

Step 2: *For all the branches arriving at node $(B^d(i), L(i))$, keep only the one with smallest aggregate distortion $\sum_{j=1}^{i-\Delta N} D_j(j)$ and prune out the others. The smallest aggregate distortion path is the surviving path for that state.*

Step 3: *Increment i and go to Step 1, repeat until $i = N + \Delta N$.*

Step 4: *At stage $N+\Delta N$, find out the state transitions with smallest aggregate distortion $\sum_{j=1}^{N} D_j(j)$. The corresponding choices* $\mathbf{x}$ *and* $\mathbf{y}$ *are the best quantizers and channel rates choices for each frame. The associated $R_{x(i)}(i)$ and $C_{y(i)}(i)$ are the optimal encoder and channel bit-allocation for the given video sequence.*

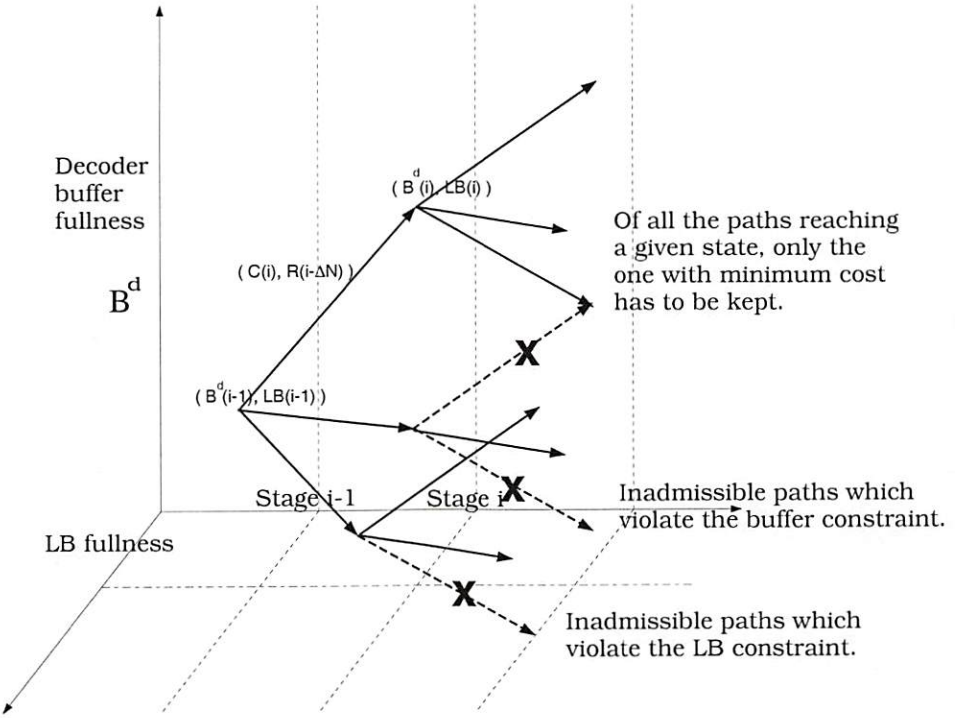


Figure 3.3: Buffer and leaky bucket constrained optimization in the VBR channel case.

3.4 Simulation and Experimental Results

For our experiments, we first use an H.261 [5, 9] encoder operating in "intra-frame only" mode. This choice simplifies the computation of the distortion for each operating point because the video quality for a video frame is independent of the choice of

quantization step size for other frames. Additional results for an MPEG inter-frame coder are presented in Section 3.5. To better control the encoder bit rate allocation, we apply the rate-control taking groups of blocks (GOBs) as the basic unit, so that each GOB can be quantized using a different quantization step.

The H.261 public domain software implementation of [60] was used in our experiments. Our basic results would not be affected if we chose a different set of quantizers or a different channel rate. In our experiments we use the "football" sequence, one of the standard video sequences used in MPEG standardization. Our luminance-only input sequence is in CIF format with 352x240 pixels, grouped into 10 GOBs and there are four possible choices of quantization step sizes, 8, 10, 12 and 31 (in H.261 the quantization step size can range from 1 to 31) for each GOB.

The time scale corresponds to the unit of time needed to display a GOB. The target average bit rate for each GOB is $\bar{C}$=5,200 bits/GOB (52,000 bits/frame). We choose this value of $\bar{C}$ as being roughly the average rate per GOB achievable with the selected set of quantizers above and the chosen video sequence. $\bar{C}$ is therefore the channel rate per GOB for the CBR case and will also be used as the drain rate of the LB in the VBR case. We use Peak Signal-to-Noise Ratio (PSNR) to measure the quality of the decoded frames[1].

We use the algorithm described in the previous section to find the best encoder and channel bit-allocation to achieve maximum PSNR in the VBR channel environment with the leaky bucket and double leaky bucket policing functions. A best encoder selection is also performed in a CBR transmission environment for comparison. We compare the best PSNR that the encoder can achieve with different policing function parameters and end-to-end delay constraints. We concentrate here on the

[1]Given the mean squared error, MSE, in a given frame, we have that $PSNR = 10\log_{10}(255^2/MSE)$.

comparison of CBR and VBR. We also select simple sets of parameters for channel rate, leaky bucket rates, etc. since our goal is to be able to qualitatively compare CBR and VBR, rather than provide definite figures for the gains of the former over the latter. These gains will depend on the application, rates, encoder and specific video sequences.

3.4.1 Comparison of CBR and VBR transmission

In the VBR channel with leaky bucket policing function, different leaky bucket sizes from $5 \times \bar{C}$ to $30 \times \bar{C}$ are used, and we set the bucket drain rate to $\bar{C}$, since this provides the most straightforward comparison to the CBR case. We use the algorithm presented in Section 3.3 to find the best joint encoder and channel rate selection with different leaky bucket parameter and different delay constraints.

Figs. 3.4 and 3.5 represent, respectively, the encoder buffer fullness during each GOB interval in the CBR channel and the encoder buffer and leaky bucket fullness during each GOB interval in the VBR channel with leaky bucket policing function. Note that the VBR case requires a slightly larger physical buffer at the encoder but provides a higher quality as seen in Fig. 3.6. Comparing Fig. 3.4 with Fig. 3.5 (top) it can be seen that the VBR buffer occupancy exceeds the physical buffer size of the CBR case (52,000 bits) by less than 10,000 bits. Note however that this additional buffer size does not result in additional end-to-end delay.

Fig. 3.6 represents the best average PSNR that the encoder can achieve with given average bit rate, end-to-end delay, and LB constraints. The optimal encoder bit-allocation for the CBR channel environment is the extreme case of VBR where LB_{max} is zero and can be obtained using the dynamic programming approach of [21]. By looking at the contour lines of the optimal PSNR from Fig. 3.6, which are shown

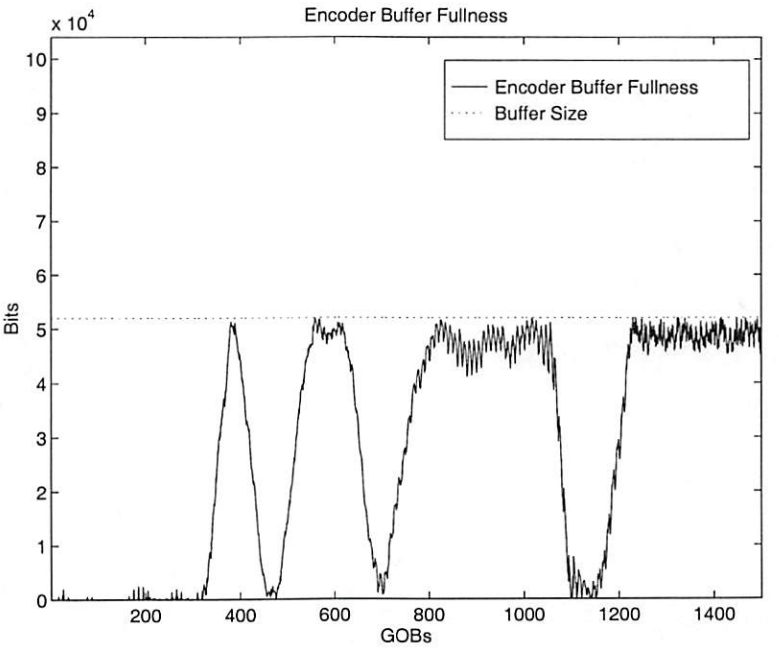


Figure 3.4: Encoder buffer fullness for CBR channel. Delay = 10 GOBs.

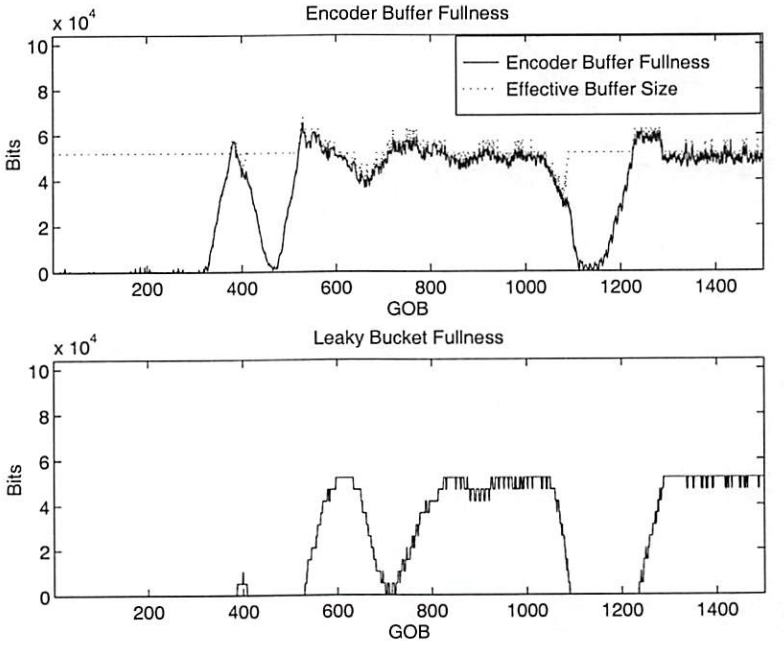


Figure 3.5: Encoder buffer and leaky bucket fullness for VBR channel. Delay = 10 GOBs, Leaky Bucket Size = $10 \cdot \bar{C}$ =52,000 bits. Note that using variable channel rates allows us to increase the effective buffer size when needed.

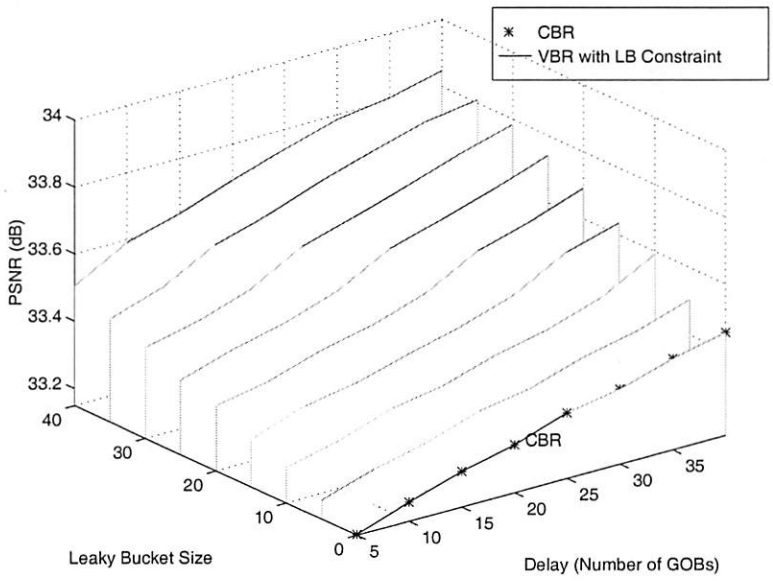


Figure 3.6: Average PSNR of video coded with different delay and leaky bucket constraints.

in Fig. 3.7, it can be seen that the PSNR level depends mostly on the sum of buffer size and LB size. Thus the upper bound on achievable video quality is the same for a given value of the sums of $\bar{C} \cdot \Delta N$ (equivalent to the effective buffer size in the CBR channel case) and the LB size. This also applies to the extreme CBR channel case, where the LB size is zero. This observation is further explored in [22] where it is shown that VBR with LB constraints is similar to CBR with a larger buffer size.

The comparison between CBR and VBR under two sets of LB constraints is depicted by Figs. 3.8, 3.9 and 3.10. Note that our average rates and end-to-end delays are *exactly the same for CBR and VBR*. Thus these two figures demonstrate the advantages of VBR transmission for the same overall rate. Figs. 3.8 and 3.9 show the rate and distortion per GOB. It can be seen that the increased PSNR in the VBR case is achieved by locally increasing the source rate over what would be possible in the CBR case. Fig. 3.10 shows the channel rates per GOB or frame and also indicates that, given the possibility of selecting a variable channel rate, as in

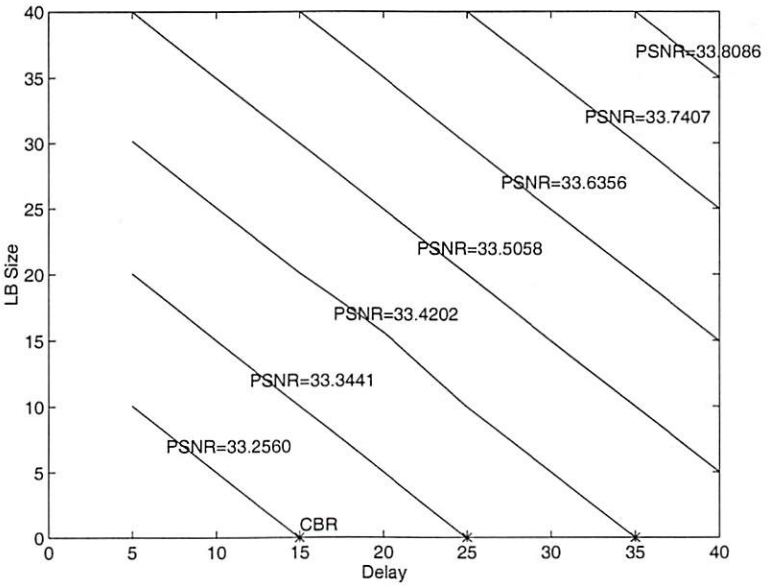


Figure 3.7: Contour of the average PSNR.

the VBR case, it is advantageous to do so[2]. Also, it can be observed that the larger the bucket sizes the more variable the channel rates will be.

3.4.2 Double leaky bucket policing function in VBR cannel

In the simulation for the VBR channel with double leaky buckets (DLB) policing function [59], the drain rate and bucket size are 5,200 bits/GOB and 52,000 bits for the larger bucket, and are 5,600 bits/GOB and 11,200 bits for the smaller bucket.

Fig. 3.11 shows the average PSNR of this DLB case compared to the single LB cases where each leaky bucket parameter is applied individually. From the above figure we can observe that the encoder is mainly constrained by the larger bucket in the DLB case if the end-to-end delay is large enough.

Fig. 3.12 shows the encoder buffer and LB fullness of the large bucket for each GOB interval. Compare the bucket fullness to that of the single leaky bucket shown

[2]Note that to simplify the optimization we only consider a discrete set of possible channel rates. Our results would be similar with increased granularity in the choice of channel rates.

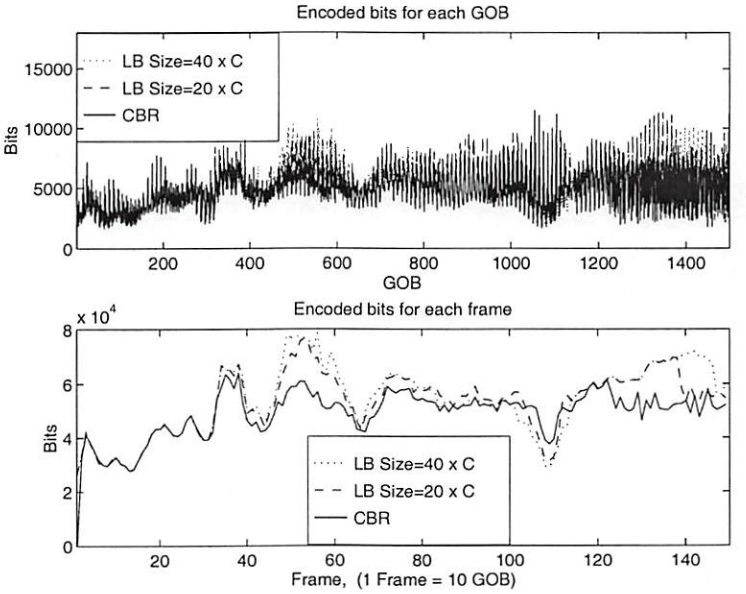


Figure 3.8: Trace of the encoded bits for each GOB (upper) and frame (bottom). The encoding rate and channel rate are jointly selected by the proposed algorithm based on DP.

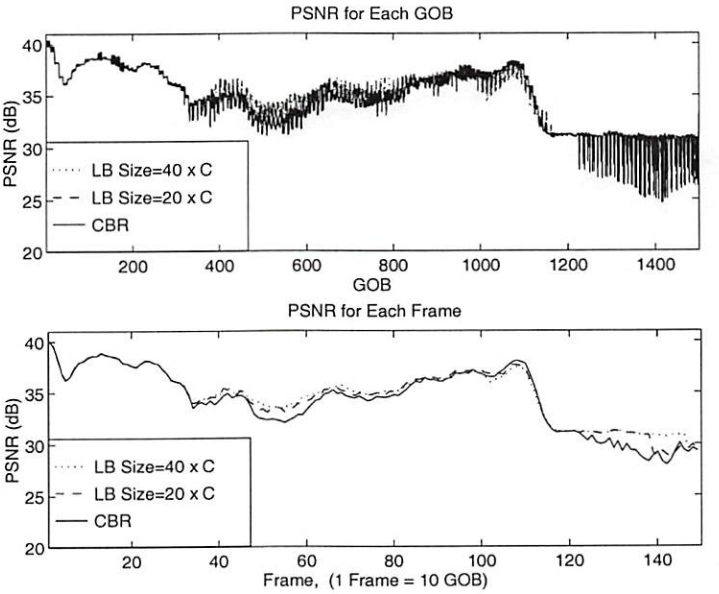


Figure 3.9: Trace of PSNR for each GOB (upper) and frame (bottom). The encoding rate and channel rate are jointly selected by the proposed algorithm based on DP.

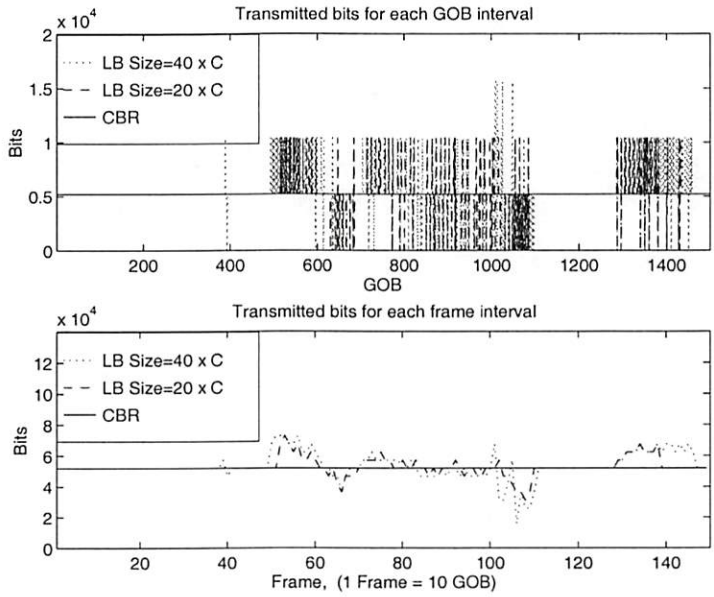


Figure 3.10: Transmitted bits for each GOB and frame interval when source rate and channel rate are jointly selected in the VBR channel with leaky bucket constraints

in Fig. 3.5, where the single leaky bucket is the same as the larger of the double leaky buckets. With 2 LB, the bucket fullness rises more slowly. This is because we have introduced the additional constraint of the smaller leaky bucket, thus imposing more restrictions on the short term rate, and producing less bursty channel rates.

3.5 MPEG Video Experiments

Video coding algorithms which exploit the temporal correlation between consecutive frames through motion compensation, such as those used in the MPEG standards [10, 4], result in greatly improved rate-distortion efficiency as compared to intra-frame-only methods as those considered in the preceding sections. However these inter-frame methods introduce a prediction loop and therefore also a dependency in the rate-distortion characteristics. For each quantization choice in a predictor frame a different R-D curve can be found for the predicted frame. For example,

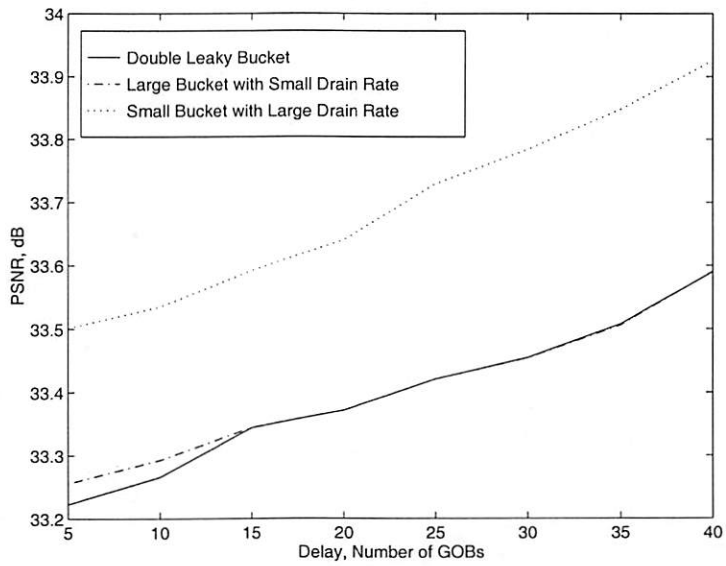


Figure 3.11: Best average PSNR of the video sequence for the double leaky bucket case and the other single leaky bucket cases.

the predicted frame will require more bits for the same quantizer if the predictor frame was coded with a coarse quantizer rather than a fine quantizer. This dependency complicates optimization procedures as it multiplies the number of allowable operating points and requires specific procedures for optimal design [61].

To provide results for MPEG video we propose an approximation to the optimal solution. Due to the dependency, a different R-D curve is generated for a given frame for each possible quantizer selection on the predictor. We alternately fix the predictors and compute the R-D data, then find the optimal solution as if the R-D points were independent using the algorithm of Section 3. Then we use the result of this optimization step to encode the predictor frames and re-start the iteration.

We use the public domain software encoder of [62]. Our results are summarized in Figs. 3.13, 3.14, 3.15, and 3.16. Our goal is again to compare CBR and VBR under LB constraints. Fig. 3.13 shows how the use of motion estimation results in a large number of bits being used for intra coded frames. Figs. 3.14 and 3.15

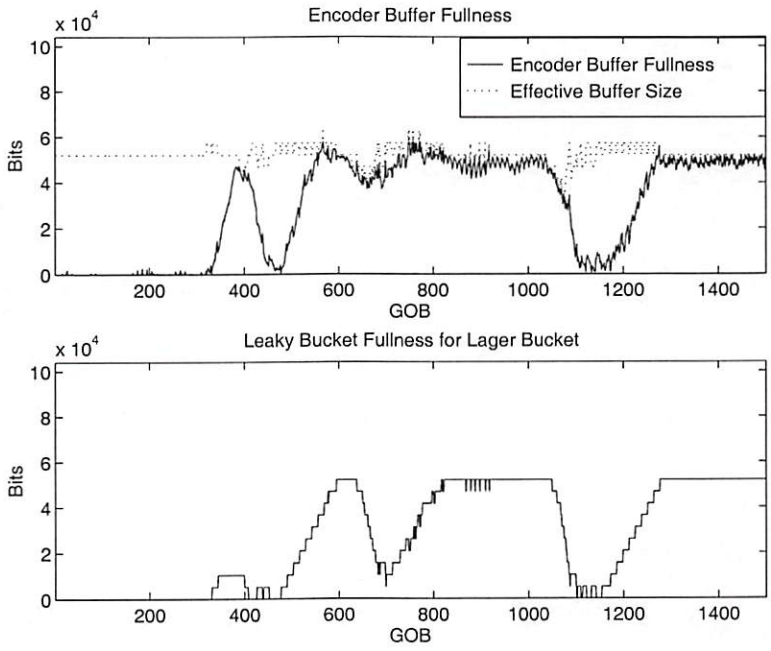


Figure 3.12: Encoder buffer and bucket fullness of larger bucket in DLB case. Delay = 10 GOBs. Larger bucket: size = 52,000 bits, drain rate = 5,200 bits/GOB. Smaller bucket: size = 10,200 bits, drain rate = 5,600 bits/GOB.

demonstrate as in previous sections the advantage of VBR transmission. When comparing the total number of bits used for each group of pictures (GOP) [3], it can be seen that VBR helps by allowing bits to be saved and used in later GOPs. Note that for many GOPs the difference in PSNR is very small. However for those GOPs where PSNR is lower than other GOPs in the video sequence (which means that those GOPs requires higher encoding rates to achieve similar video quality as other GOPs), VBR transmission outperform CBR transmission by about 1dB gain in PSNR.

While it may be expected that the VBR advantage over CBR would be greater for video compressed using inter-frame techniques such as motion compensation

[3]The set of frames including an intra-coded frame and all the predicted frames until the next intra frame. We use GOPs of size 6 and use 2 B-frames per P-frame (i.e., one GOP has the form IBBPBB).

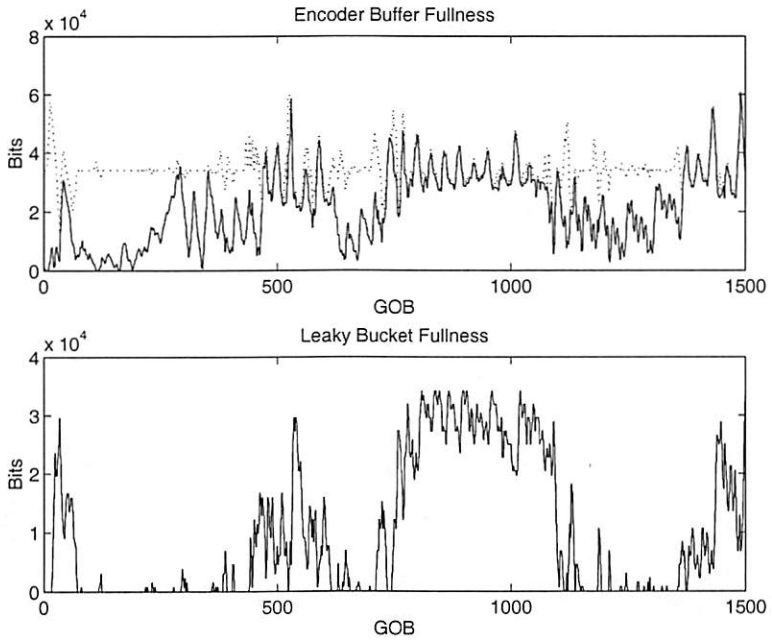


Figure 3.13: Encoder buffer and leaky bucket fullness for VBR channel. Target average rate $\bar{C}$ = 3,400 bits/GOB. Delay = 10 GOBs, Leaky Bucket Size = $10 \cdot \bar{C}$ =34,000 bits/GOB.

(instead of intra techniques as in the earlier section), Fig. 3.15 indicates that this is not the case. Quality performance improvements are also small in this case. In fact, in [22] show that performance improvements are bounded by the quality of CBR video using a physical buffer as large as the virtual buffer in the VBR case, regardless of the compression techniques used. This is also demonstrated in Fig. 3.16 which experimentally shows that VBR is equivalent to having a larger buffer without incurring in the additional end-to-end delay. Note that the difference in average PSNR is relatively small, however the difference in specific GOPs or scenes can be significant. Effectively, VBR borrows bits from "easy" video segments to increase the quality of "difficult" video segments.

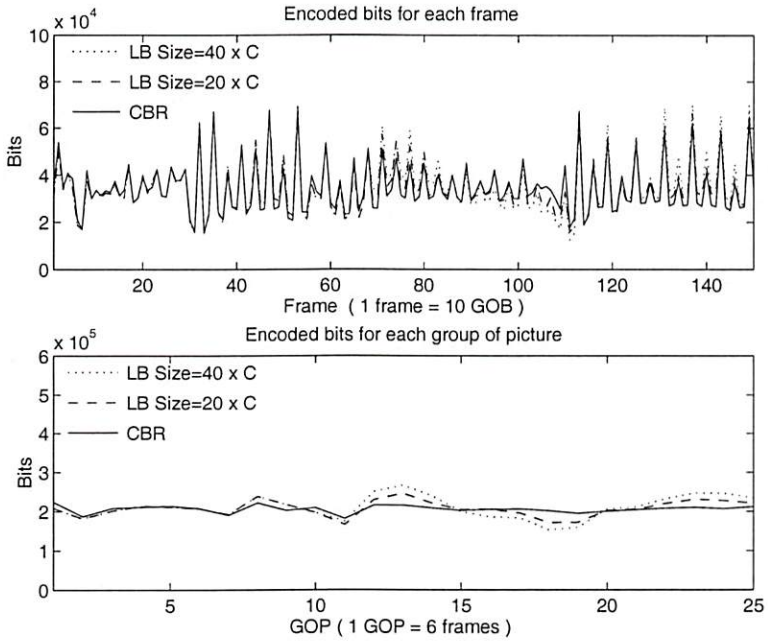


Figure 3.14: Trace of the encoded bits for each frame and group of picture (GOP) using MPEG encoder in the VBR channel. Target average rate $\bar{C}$ = 3,400 bits/GOB. Delay = 10 GOBs, Leaky Bucket Size = $10 \cdot \bar{C}$ =34,000 bits/GOB

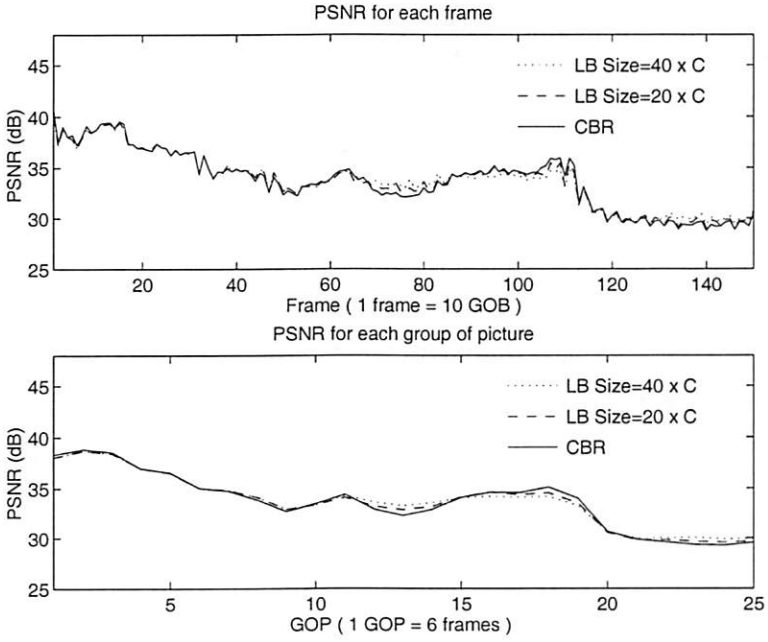


Figure 3.15: Trace of PSNR for each frame and group of picture (GOP) using MPEG encoder in the VBR channel. Target average rate $\bar{C}$ = 3,400 bits/GOB. Delay = 10 GOBs, Leaky Bucket Size = $10 \cdot \bar{C}$ =34,000 bits/GOB

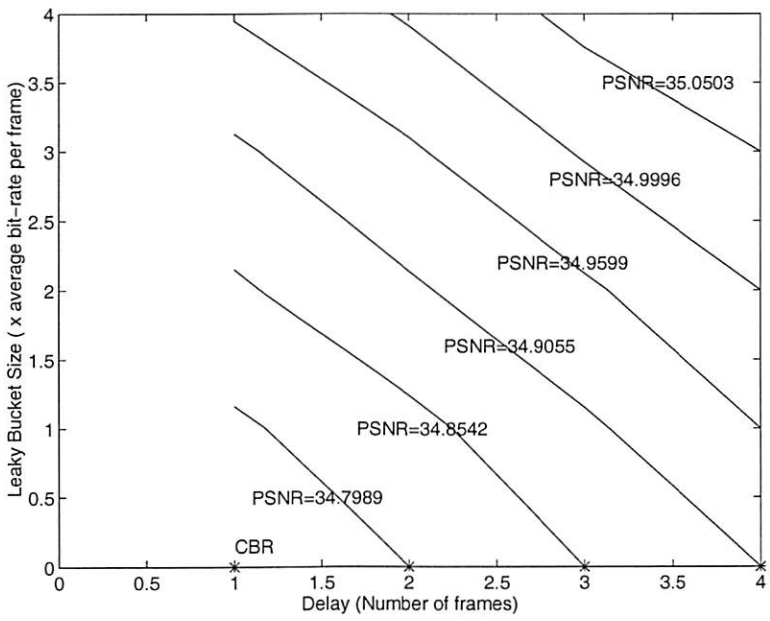


Figure 3.16: Contour of the average PSNR. Target average rate $\bar{C}$=5,200 bits/GOB

3.6 Conclusions

In this chapter we have studied real-time video transmission over networks where VBR transmission is possible under constraints on the channel rates that the encoder can select. Specifically we use ATM networks as an example of this type of VBR networks. We have shown how two sets of constraints come into play for such a system when it comes to selecting the source and channel rates, namely (i) constraints due to the constant end-to-end delay needed to maintain real time video playback, and (ii) constraints due to network policing of the channel rate usage. We have considered end-to-end delay, policing constraints and physical buffer sizes at encoder and decoder as our major design parameters and have shown how they each can affect the resulting video quality.

We have formulated an optimization problem where the goal is to select the source and channel rates to maximize the video quality without violating the above

constraints. We have introduced an algorithm based on dynamic programming which solves this optimization problem for most practical policing functions, including the leaky bucket and jumping window. We have used our algorithm to demonstrate experimentally the advantages of VBR transmission. We have shown how increased PSNR is possible in the VBR case with the same average channel rate and end-to-end delay.

The optimization algorithm presented in this chapter allows determination of the best video quality that can be achieved under VBR channel constraints. While this algorithm may not be viable for real-time on-the-fly compression, it does provide a useful benchmark for other suboptimal algorithms.

As shown in [22], a simple upper bound on achievable video quality given an LB constraint is the CBR video quality with the virtual buffer size equal to the sum of physical buffer size and leaky bucket size. The advantage of VBR transmission comes then from the fact that the same level of quality can be achieved with a smaller end-to-end delay than in CBR, assuming that the network can support VBR transmission with reasonably low transmission delay.

Chapter 4

Video Transmission with Real-Time Encoding and Decoding

4.1 Introduction

In this chapter, we focus on the problem of rate control for video transmission in applications that require real-time video encoding, data transferring and video decoding. This type of video transmission were categorized as type III applications in our classification for different types video applications in Chapter 2 (refer to Section 2.1 and Fig. 2.1). As has been discussed, a primary concern in those applications is the delay constraints on the transport of video data. The delay constraints are more restrictive in the bidirectional and interactive communication applications such as video conferencing and videophone. In order to enable real-time conversation in those visual communication applications, the end-to-end delay latency for video transmission has to be small or imperceptible to users.

In such real-time video applications, a low-delay codec is certainly necessary to reduce the overall delay latency of the video transmission. A common approach

to reduce the encoding delay is to use the slice-based encoding scheme. In a slice-based video codec, data of a video frame is sub-divided into smaller data units such as slices or Group of Blocks (GOB), which individually have shorter encoding delay than that of the whole video frame. The data transmission can begin as soon as each individual slice is encoded, instead of waiting for the completion of the encoding for the entire frame. The slice-based video encoding schemes have been used in MPEG standards, H.261, H.263[1], and the extension of H.263 standards. The result of the slice-based encoding is the reduction of the encoding delay for most data slices in each video frame.

Therefore, we will assume a slice-based video codec in our discussion for the real-time video transmission in this chapter. We will formulate the delay constraint for each slice by looking at the delay elements inside the encoding and decoding processes. The rate constraint for each slice is derived from the delay constraint and channel constraint, and again the optimal rate control is formulated as that which aims to minimize the distortion of reconstructed video at the decoder. Unlike the video applications discussed in Chapter 3 in which video encoding and rate control are performed off-line before data transmission begin (hence optimal rate control can be planned for the entire video sequence), the applications we discuss here have real-time encoding and transmission requirements. Rate control is performed locally on a sliding window of a video segment as video data is continuously captured at the input device and streamed to the destination. In Section 4.5 and 4.6, novel algorithms based on dynamic programming and Lagrangian optimization are proposed to solve the optimal rate allocation problem in this real-time video transmission environment.

[1]In H.261 and H.263, such partition is referred as Group of Block (GOB).

4.2 Delay Constraints

Following the discussion about the delay in a video transmission system in Section 2.2, we define ΔT as the end-to-end delay for video transmission which includes the delay from various components in the system: video capture, video encoding, data transmission and video decoding. As shown in (2.1), the end-to-end delay ΔT in a video transmission system can be expressed as:

$$\begin{aligned} \Delta T &= \Delta T_e \quad \text{(Encoder delay)} + \Delta T_{eb} \quad \text{(Encoder buffer delay)} \\ &+ \Delta T_c \quad \text{(Channel delay)} \\ &+ \Delta T_{db} \quad \text{(Decoder buffer delay)} + \Delta T_d \quad \text{(Decoder delay).} \end{aligned} \tag{4.1}$$

From the discussion regarding the end-to-end delay constraint in a video transmission system in Section 2.2, ΔT has to be constant in order to maintain correct decoding and displaying timing at the decoder. This delay constraint in a frame-based coding

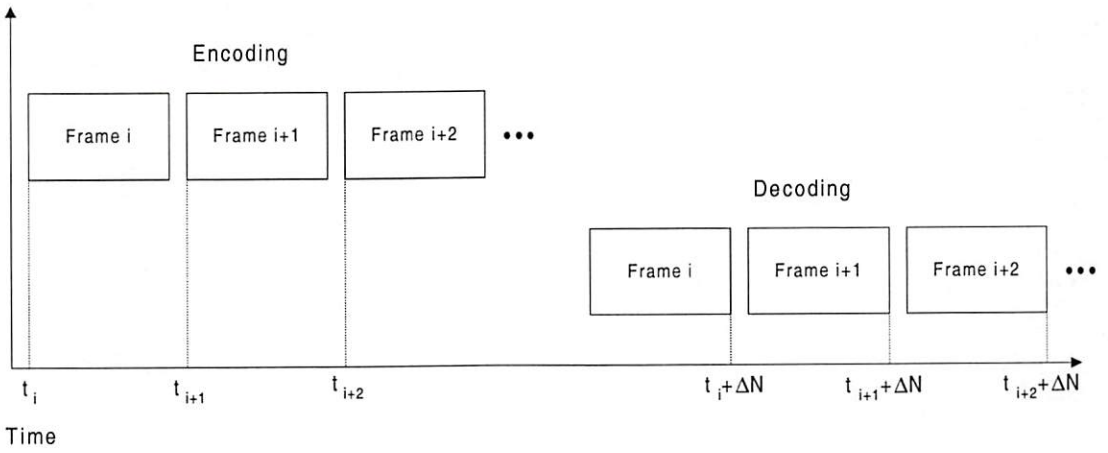


Figure 4.1: Delay constraint in the frame-based encoding scheme.

In the slice-based encoding scheme, the delay components for each slice is slightly different from (4.1), because every slice in a video frame will have different encoder delay. Assuming that a video frame is equally subdivided into G slices which are indexed from slice 1 to G, and it takes ΔT_{es} sec to encode each sliced data. Normally there is a delay latency, e.g., delay for motion estimation and compensation processes, before the encoding of first slice in a video frame. Define this delay component as ΔT_{ef}. Assuming that ΔT_{es} is constant for every slice, then the encoder delay for the g-th slice in a video frame, defined as $\Delta T_e^{(g)}$ can be expressed as:

$$\Delta T_e^{(g)} = \Delta T_{ef} + g \times \Delta T_{es}. \tag{4.2}$$

Hence, the encoder delay for each slice $\Delta T_e^{(g)}$ is variable and dependent on the slice index g. For convenience we index the g-th slice of frame i as slice (i, g). The transmission of slices is assumed to occur in the same order as they were encoded. Assuming that data of frame i are captured and sent to the encoder at time t_i, then slice (i, g) will be encoded and released to the encoder buffer at time $t_i + \Delta T_{ef} + g \cdot \Delta T_{es}$ as depicted in Fig. 4.2.

The constant end-to-end delay constraint ΔT is still applied to the slice-based video codec. That is, slices of frame i that are captured at time i will be displayed at time $t_i + \Delta T$. In order to meet this delay constraint, data of frame i has to be transmitted by the time $t_i + \Delta T - \Delta T_d - \Delta T_c$ (ΔT_d and ΔT_c are included to account for the decoder and channel delays). Denote u_i as this time constraint for transmitting frame i at the transmitter to guarantee that frame i can be decoded at the encoder in time, then u_i can be written as:

$$u_i = t_i + \Delta T - \Delta T_d - \Delta T_c \tag{4.3}$$

Fig. 4.2 shows such timing and delay constraints in delivering video data.

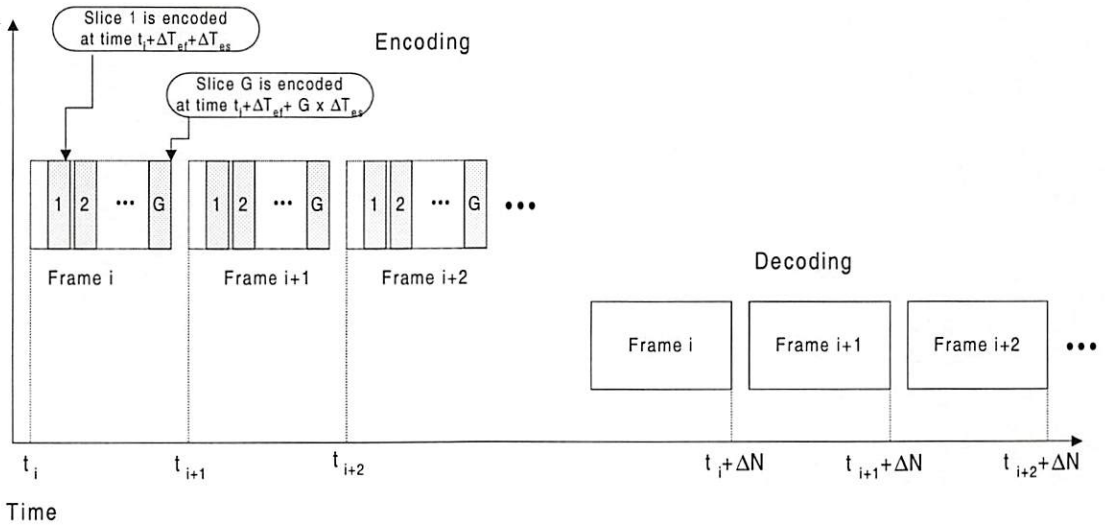


Figure 4.2: Delay constraint in the slice-based encoding scheme.

4.3 Encoding Rate Constraints

We assume a transmission environment where encoded video data are packetized for transmission. As seen in previous chapters that delay constraints can be translated into encoding rate constraints, similar rate constraints can also be formulated for the slice-based codec in the packet transmission environment. Assuming that packets are transmitted at a constant time interval T_p sec. Therefore, if packet transmission begins at time 0, then the p-th packet will be transmitted at time t where $p = \lfloor \frac{t}{T_p} \rfloor$. Define $C(p)$ as the payload size of packet p which is transmitted at time $p \times T_p$. At slice-based the video encoder, define $r(i, g)$ as the number of bits that are used to encode slice (i, g) in frame i.

The size of each encoded slice is variable due to the VBR nature of video compression. Therefore a compressed slice data may have been transported by several

different packets. Consider the buffer content and system state as shown in Fig. 4.3. Assuming that when the system is observed at time $t = p \times T_p$ (i.e., the instant that p-th packet is transmitted) slice (m, g_{tx}) of frame m is currently transmitted by the channel, and slice (n, g_{in}) of frame n is the last slice which is encoded and released to the encoder buffer. It is possible that after packet p is transported, only part of slice (m, g_{tx}) data are transmitted. Hence we denote $r'(m, g_{tx})$ as the number of bits for the remaining part of slice (m, g_{tx}) which is still in the encoder buffer and waiting for transmission. Therefore at time t, the encoder buffer contains data from slice $(m, g_{tx} + 1)$ to slice (n, g_{in}), and part of slice (m, g_{tx}) with $r'(m, g_{tx})$ bits data.

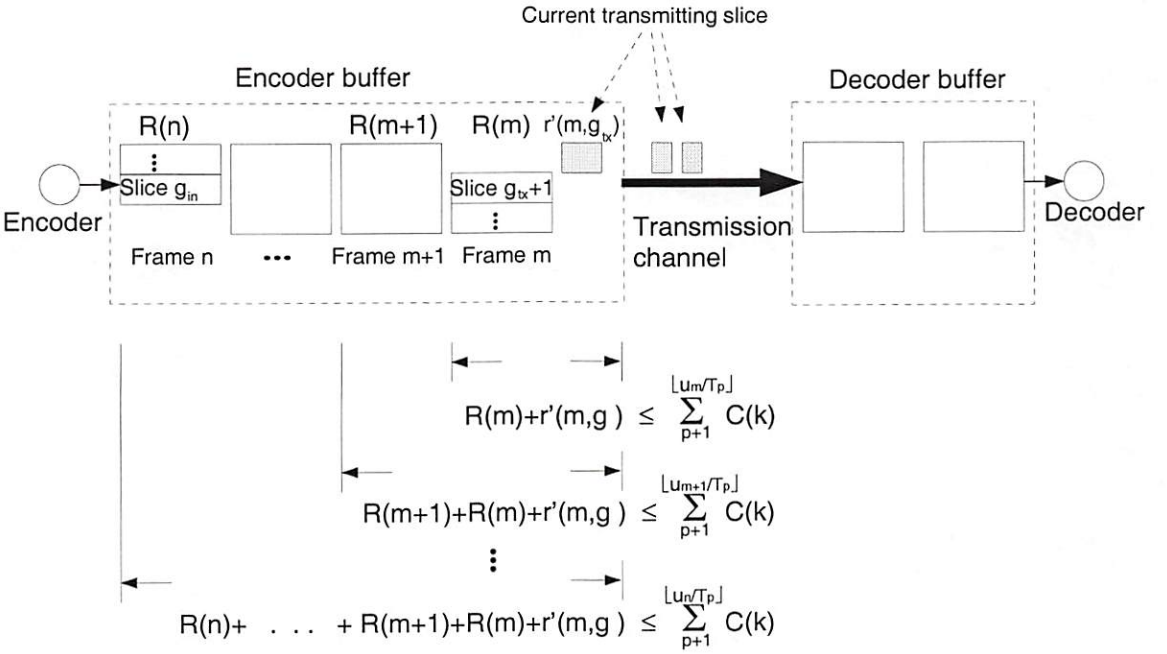


Figure 4.3: Encoding rate constraints for video slices in the encoder buffer at time t.

For convenience, we use $R(i)$ to represent the accumulated encoding rate for those slices of frame i in the encoder buffer, i.e.,

$$\begin{cases} R(m) = \sum_{g=g_{tx}+1}^{G} r(m,g); \\ R(i) = \sum r(i,g), \forall i = m+1, \ldots, n-1; \\ R(n) = \sum_{g=1}^{g_{in}} r(n,g). \end{cases}$$

Note that $R(m)$ and $R(n)$ only contain encoding rates for part of frame m and frame n, because some slices of those frames (the currently transmitted frame and currently captured frame) are not currently buffered in the encoder buffer at time t.

Because of the constant end-to-end delay constraint ΔT that are imposed on the transport of video data, all the slices of frame i have to be transmitted by time u_i in order to be decoded and displayed at time $t_i + \Delta T$. Hence at time t, the condition for the data of frame i to arrive at the encoder in time for decoding is that all the data corresponding to frame i, as well as to all the previous frames in the encoder buffer, has to be transmitted by the channel before time u_i as (see Fig. 4.3):

$$r'(m, g_{tx}) + \sum_{j=m}^{i} R(j) \leq \sum_{k=p+1}^{\lfloor u_i/T_p \rfloor} C(k), \text{ where } p = \lfloor \frac{t}{T_p} \rfloor. \tag{4.4}$$

Therefore the constraints on the encoding rates for those slices in the encoder buffer (i.e., slices in frame m to frame n in our example) can be summarized as:

$$\begin{aligned} R(m) &\leq \left[\sum_{k=p+1}^{\lfloor u_m/T_p \rfloor} C(k) \right] - r'(m, g_{tx}) \\ R(m+1) + R(m) &\leq \left[\sum_{k=p+1}^{\lfloor u_{m+1}/T_p \rfloor} C(k) \right] - r'(m, g_{tx}) \\ &\vdots \end{aligned} \tag{4.5}$$

Delay components and time indexes:	
ΔT (sec):	End-to-end delay for every frame,
$\Delta T^{(g)}$ (sec):	Encoder delay for g-th slice in each frame,
t_i:	Time at which data of frame i is input to the encoder,
u_i:	Due time at which data of frame i have to be transmitted,
T_f (sec):	Frame interval,
T_p (sec):	Packet interval,
G (slice):	Number of slices per frame.

Following system states are assumed at a given time t instant:	
slice (n, g_{in}):	The g_{in}-th slice of frame n is the currently encoded slice,
slice (m, g_{tx}):	The g_{tx}-th slice of frame n is the currently transmitted slice,
packet $p = \lfloor \frac{t}{T_p} \rfloor$:	The packet that is currently being transmitted data at time t.

Table 4.1: Summary of notations.

$$R(n) + \ldots + R(m+1) + R(m) \leq \left[\sum_{k=p+1}^{\lfloor u_n/T_p \rfloor} C(k) \right] - r'(m, g_{tx})$$

Given those constraints on the encoding rates, the selection of the encoding rates for those slices in the encoder buffer have to comply with those rate constraints in order to avoid violating the delay constraints. The variable and notations used in our formulations are summarized in Table 4.1.

4.4 Formulation of Optimal Rate Control

From (4.5), the encoding rate constraints at the instant of time t are related to the channel transmission rates $C(k)$ and the transmission time constraint u_i. In the transmission environment with varying transmission rates, ideally the bit rates of the encoded video data should be able to dynamically scaled up or down to cope with the variation of channel rate. This can be achieved by controlling the compression parameters at the video encoder. However for most video codecs, the bit rate of

the encoded video data are determined during the encoding process and can not be changed thereafter. When the future channel rates cannot been foreseen at the time of encoding, it may be required that the bit rate of the video data stream can be adjusted even *after video are compressed.* To achieve this rate scalability for the encoded video data stream, a possible system implementation can consist of having video data quantized with different quantizers stored in separate buffers, and each storing slice quantized with one particular quantizer. (see Fig. 4.4). Then the video data currently being transmitted will be drawn from the appropriate buffer. An alternative, and more elegant, approach would be to have video data encoded into layers of bitstreams: a base layer which contains minimum subset data that can be decoded into useful video at the decoder, and enhancement layers that contains the refinement data on top of the base layer video. Rate control thus can take advantage of this scalability to dynamically adjust, according to the current channel conditions, the number of layers that are being sent through the channel.

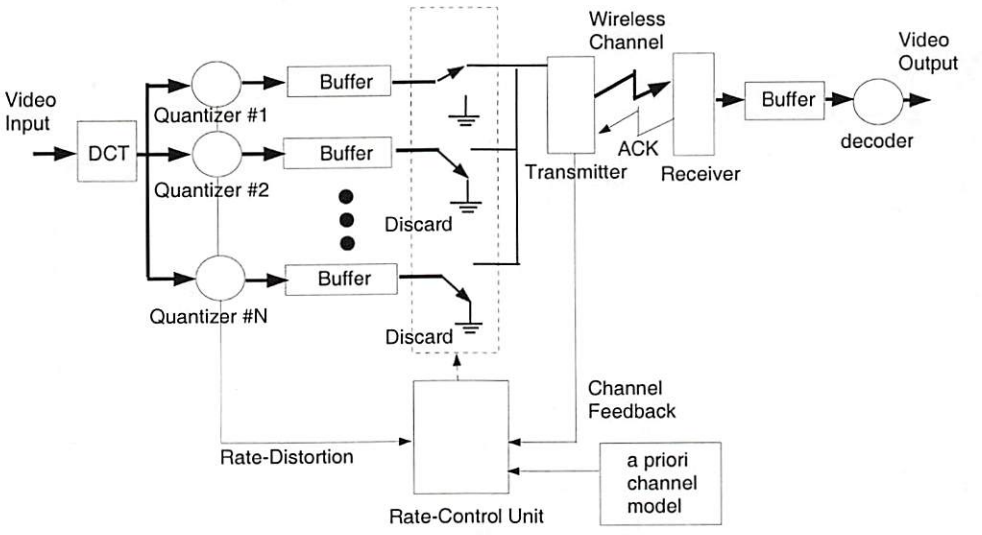


Figure 4.4: System block diagram.

In the following discussion, we still formulate the rate control problem by assuming that encoding rates are controlled by the selection of quantizers at the encoder. However, this selection of quantizers can be changed before video data are actually sent to the channel for transmission (See Fig. 4.4). Conceptually selecting the encoding quantizers to adjust the source bit rate is equivalent to selecting the number of layers that are transmitted to the decoder in the layer coding scheme. The concatenated layers of bit streams are also associated with an effective rate and distortion.

We assume that each slice is encoded with a quantizer chosen from a finite quantizer set $\mathcal{Q}$. While other rate control options to assign quantization scale at the frame layer or macroblock layer are also possible, the rate control algorithm discussed here can also be extended to other scenarios. In (4.5) the buffered video data in the encoder buffer contains data from slice (m, g_{tx}) to slice (n, g_{in}). From our assumption, the encoding rates $r(m, g_{tx}+1), \ldots, r(n, g_{in})$ can be modified while those slices are still stored in the encoder buffer. However $r'(m, g_{tx}+1)$, the remaining bits of the currently transmitted slice, cannot be adjusted because part of slice $(m, g_{tx}+1)$ has been sent out.

Denote $x(i, g) \in \mathcal{Q}$ as the choice of quantizer for slice (i, g), and $r_{x(i,g)}(i, g)$ and $d_{x(i,g)}(i, g)$ as the associated encoding rate and distortion. We also define a quantizer vector $\boldsymbol{X}(i)$ to represent the choices of quantizers for slices in frame i as:

$$\begin{cases} \boldsymbol{X}(m) = [x(m, g_{tx}+1), \ldots, x(m, G)], & \text{when } i = m; \\ \boldsymbol{X}(i) = [x(i, 1), \ldots, x(i, G)], & \text{when } m < i < n; \\ \boldsymbol{X}(n) = [x(n, 1), \ldots, x(i, g_{in})], & \text{when } i = n. \end{cases}$$

Note that $\boldsymbol{X}(m)$ and $\boldsymbol{X}(n)$ only represent the quantizer choices for the slices in the portion of frame m and n (the currently transmitted frame and the currently

captured frame, respectively) which are buffered in the encoder buffer. Denote $R_{\boldsymbol{X}(i)}(i)$ and $D_{\boldsymbol{X}(i)}(i)$ as the accumulated encoding rates and distortion for frame i given that quantizer choices $\boldsymbol{X}(i)$ are used to encode the slices in frame i. The goal of rate control is to choose the quantizers $\mathcal{X} = \{\boldsymbol{X}(m), \boldsymbol{X}(m+1), \ldots, \boldsymbol{X}(n)\}$ such that the accumulated distortion is minimum, meanwhile the selected encoded rates can meet the rate constraints as formulated in (4.5). This optimal rate control problem can be formulated as:

Formulation 2 *At time t ($t = p \times T_p$), find the optimal quantizer choices $\mathcal{X}^*$ such that,*

$$\mathcal{X}^* = \arg\min_{\mathcal{X}} \sum_{j=m}^{n} D_{\boldsymbol{X}(j)}(j), \tag{4.6}$$

subject to the constraint set:

$$\sum_{j=m}^{i} R_{\boldsymbol{X}(j)}(j) \leq \left(\sum_{k=p+1}^{\lfloor u_i/T_p \rfloor} C(k) \right) - r'(m, g_{tx}), \quad \forall i = m, \ldots, n \tag{4.7}$$

In the above optimal rate control formulation, the selection of quantizers are based on the known information that are available up to time t, i.e., the encoding rate and distortion for slices that are captured before time t. The rate control thus performed is inherently a local optimization on a slide window of the video segment, so the global optimality can not be guaranteed.

4.5 Rate Allocation by Dynamic Programming

We consider a video compression scheme in which all video frames of a video sequence are encoded as intra-frame. When each video frame is encoded as intra-frame, the

encoding rate and distortion for each slice is solely dependent on the quantizer selection (on the contrary the encoding rate and distortion will be affected by the quantizer choice of the reference frames if inter-frame coding is used). Given the rate control problem formulated as Formulation 2, we can use the Dynamic Programming (DP) technique to search for an optimal quantizer choice $\mathcal{X}^*$ on the sliding window of video segment from slice (m, g_{tx}) to slice (n, g_{in}).

Consider Fig. 4.5. Our goal in solving Formulation 2 is to find the best quantizer choices $\boldsymbol{X}(m), \ldots, \boldsymbol{X}(n)$ (all those currently in the buffer) so that none of the constraints of (4.5) are violated. A trellis is formed to search for such optimal quantizer choices. The y-axis in Fig. 4.5 represents the accumulated rate (state) and the x-axis represents the slice considered (stage). The algorithm begins from the the initial state, i.e., stage (m, g_{tx}), with initial buffer occupancy $r'(m, g_{tx})$. Each branch in the trellis represents a choice of quantizer for each slice, e.g., a branch linking stages (m, g_{tx}) and $(m, g_{tx}+1)$ represents a choice of quantizer for slice $(m, g_{tx}+1)$.

Define $\mathcal{X}(i, g)$ as the sequences of quantizer choices from the first slice in the encoder buffer up to slice (i, g), and $B_{\mathcal{X}(i,g)}(i, g)$ as the accumulated rate for those slices. That is,

$$\mathcal{X}(i, g) = \{X(m), \ldots, X(i-1), x(i, 1), \ldots, x(i, g)\}, \tag{4.8}$$

and

$$B_{\mathcal{X}(i,g)}(i, g) = r'(m, g_{tx}) + \sum_{j=m}^{i-1} R_{\boldsymbol{X}(j)}(j) + \sum_{h=1}^{g} r_{x(i,h)}(i, h). \tag{4.9}$$

In the trellis each state at stage (i, g) represents a possible level of accumulated rate. Denote $\mathcal{S}_{i,g}(B)$ as the state with B bits of accumulated rate. Because of the rate

constraints (4.5), only the states with state variable B that meet the constraints are valid. That is, state $\mathcal{S}_{i,g}(B)$ is valid if:

$$B \leq \sum_{k=p+1}^{\lfloor u_i/T_p \rfloor} C(k) \tag{4.10}$$

Suppose a set of quantizer choices $\mathcal{X}(i,g)$ results in an accumulated rate $B_{\mathcal{X}(i,g)}(i,g)$ (hence arrives at state $\mathcal{S}_{i,g}(B_{\mathcal{X}(i,g)}(i,g))$). State $\mathcal{S}_{i,g}(B_{\mathcal{X}(i,g)}(i,g))$ is associated with the accumulated distortion,

$$\sum_{j=m}^{i-1} D_{\boldsymbol{X}(j)}(j) + \sum_{h=1}^{g} d_{x(i,h)}(i,h) \tag{4.11}$$

as the cost for that state. Given that a choice of quantizer $x(i,g+1)$ is used to encode slice $(i,g+1)$ and results in the encoding rate $r_{x(i,g+1)}(i,g+1)$ and distortion $d_{x(i,g+1)}(i,g+1)$, then the resulting accumulated rate is:

$$B_{\mathcal{X}(i,g+1)}(i,g+1) = B_{\mathcal{X}(i,g)}(i,g) + r_{x(i,g+1)}(i,g+1) \tag{4.12}$$

and arrives at the state $\mathcal{S}_{i,g}(B_{\mathcal{X}(i,g+1)}(i,g+1))$. Such choice of quantizer is represented by a branch that connects the node of state $\mathcal{S}_{i,g}(B_{\mathcal{X}(i,g)}(i,g))$ at stage (i,g) to the node of state $\mathcal{S}_{i,g+1}(B_{\mathcal{X}(i,g+1)}(i,g+1))$ at stage $(i,g+1)$ with cost $\sum_{j=m}^{i-1} D_{\boldsymbol{X}(j)}(j) + \sum_{h=1}^{g+1} d_{x(i,h)}(i,h)$.

Any branch that violates the rate constraint (4.10) is pruned out. If two or more sets of quantizer choices result in the same accumulated encoding rate B and arrive at the same state $S_{i,g+1}(B)$ at stage $(i,g+1)$, only the path that results in the minimum accumulated distortion at the given state is kept and all the other sub-optimal paths are pruned out. This is based on Bellman's optimality principle [63] and it is easy

to see that (because rate and distortion are decoupled for each block) it also applies in this case. Therefore the cost associated with state $S_{i,g+1}(B_{\mathcal{X}(i,g+1)}(i,g+1))$ is the minimum among those quantizer choices $\tilde{\mathcal{X}}$ that result in the same accumulated rate, i.e., for those $\tilde{\mathcal{X}}$ that $B_{\tilde{\mathcal{X}}(i,g+1)}(i,g+1) = B_{\mathcal{X}(i,g+1)}(i,g+1)$, if the cost associated with the previous state, $S_{i,g}(B_{\mathcal{X}(i,g)}(i,g))$, is also minimum. Note that in the above discussion, if slice (i,g) is the last slice in frame i (i.e., $g = G$), then the next slice (or stage) will be slice $(i+1,1)$.

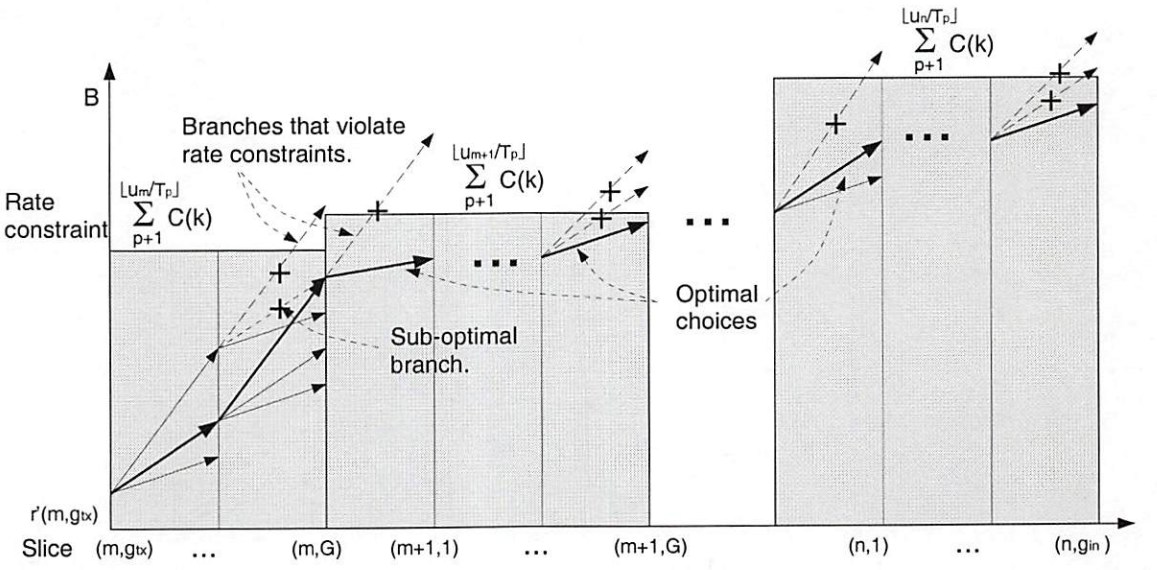


Figure 4.5: Trellis tree in dynamic programming for searching the optimal encoding rate allocation.

The path that leads to the minimum cost state at the final stage is thus the optimal solution to Formulation 2, and each branch on that path represents the optimal choice of quantizer for encoding each slice. By pruning sub-optimal solutions at every intermediate stage, the combination of quantizer choices that can achieve minimum distortion can thus be found without trying all the possible combinations. However the complexity can still be fairly high depending on the number of stages

and the number of states per stage. This prompts us to consider a faster optimization approach.

4.6 Rate Allocation by Lagrangian Optimization

Using Lagrangian optimization for rate control under multiple rate constraints was previously studied in [64, 54, 24]. The constrained optimization problem in Formulation 2 is equivalent to an unconstrained problem by introducing a non-negative Lagrange multiplier λ_i associated with each rate constraint (4.5) in Formulation 4.6 as:

Formulation 3 *At time t ($t = p \cdot T_p$), find the quantizer choice $\mathcal{X}^*$ such that*

$$\mathcal{X}^* = \arg\min_{\mathcal{X}} \sum_{j=m}^{n} D_{\boldsymbol{X}(j)}(j) + \sum_{i=m}^{n} \lambda_i \cdot \left(\sum_{j=m}^{i} R_{\boldsymbol{X}(j)}(j) \right) \tag{4.13}$$

Multipliers $\lambda_m, \ldots, \lambda_n$ are introduced to replace the $n-m+1$ constraints in (4.7). The problem that remains is to find out the appropriate multipliers $\lambda_m, \ldots, \lambda_n$ such that no constraint is violated. Define λ'_j as:

$$\lambda'_j = \textstyle\sum_{i=j}^{n} \lambda_i, \quad \forall j \in \{m, \ldots, n\}. \tag{4.14}$$

then (4.13) can be rearranged as:

$$\mathcal{X}^* = \arg\min_{\mathcal{X}} \sum_{j=m}^{n} \left(D_{\boldsymbol{X}(j)}(j) + \lambda'_j \cdot R_{\boldsymbol{X}(j)}(j) \right) \tag{4.15}$$

Since $\lambda_m, \ldots, \lambda_n$ are all non-negative values, from (4.14) we have

$$\lambda'_m \geq \lambda'_{m+1} \geq \ldots \geq \lambda'_n, \tag{4.16}$$

and the mapping $\{\lambda_m, \ldots, \lambda_n\} \rightarrow \{\lambda'_m \ldots, \lambda'_n\}$ is one-to-one. Thus Formulation 3 is equivalent to finding the appropriate non-negative values of $\{\lambda'_m, \ldots, \lambda'_n\}$ such that no constraint is violated.

If each video frame is encoded as intra-frame, the optimization problem (4.15) is equivalent to searching for the optimal quantizer choice for each slice independently as:

$$x^*(i,g) = \arg \min_{x(i,g) \in \mathcal{Q}} d_{x(i,g)}(i,g) + \lambda'_i \cdot r_{x(i,g)}(i,g) \qquad (4.17)$$
$$\forall (i,g) \in \{(m, g_{tx}+1), \ldots, (n, g_{in})\}.$$

Then the remaining problem is how to determine a set of Lagrange multipliers $\{\lambda'_m, \ldots, \lambda'_n\}$ such that the rate constraints are met. In [64, 24, 25] a similar problem is solved by iteratively increasing the lower bounds on the multipliers, defined as $\{\Lambda'_m, \ldots, \Lambda'_n\}$, such that the violation of constraints can be avoided, until the optimal encoding rate allocation is found. Initially the quantizer choices $\hat{\mathcal{X}} = \{\hat{X}(m), \ldots, \hat{X}(n)\}$ are selected by Lagrangian optimization subject to only one constraint on the total accumulated rates of all slices in the buffer: $\sum_{j=m}^{n} R_{\hat{X}(j)}(j) \leq \left[\sum_{k=p+1}^{\lfloor u_n/T_p \rfloor} C(k)\right] - r'(m, g_{tx})$. Only one multiplier λ_n is associated with the constraint, and other multipliers $\lambda_m, \ldots, \lambda_{n-1}$ are set to be 0 in (4.13) as:

$$\hat{\mathcal{X}} = \arg \min_{\mathcal{X}} \left(\sum_{j=m}^{n} D_{\boldsymbol{X}(j)}(j) \right) + \lambda_n \cdot \left(\sum_{j=m}^{n} R_{\boldsymbol{X}(j)}(j) \right) \qquad (4.18)$$

From (4.14), this is equivalent to setting $\lambda'_m = \ldots = \lambda'_n = \lambda_n$ in (4.17). The optimal quantizer choice and the value of λ_n can be found for each slice independently as:

$$x^*(i,g) = \arg \min_{x(i,g) \in \mathcal{Q}} d_{x(i,g)}(i,g) + \lambda'_n \cdot r_{x(i,g)}(i,g)$$

$$\forall (i,g) \in \{(m, g_{tx}+1), \ldots, (n, g_{in})\}.$$

The optimal quantizer choices and the appropriate value of λ'_n can be found simultaneously by the bisection search technique. Given that $\hat{\mathcal{X}}$ is selected, if no other violation of constraints (4.7) are caused by using $\hat{\mathcal{X}}$, then $\hat{\mathcal{X}}$ is the solution to Formulation 3. Otherwise, the quantizer choices $\hat{\mathcal{X}}$ are not the desired solution and other rate constraints also have to be taken into account in the optimization process by including more Lagrangian multipliers.

Assuming that frame v, where $v < n$, is the "last" frame which violates the rate constraints, given that the quantizer choices $\hat{\mathcal{X}}$ are used, i.e.,

$$\exists\, v \text{ where } \sum_{j=m}^{v} R_{\mathcal{X}(j)}(j) > \left[\sum_{k=p+1}^{\lfloor u_v/T_p \rfloor} C(k)\right] - r'(r, g_{tx}), \tag{4.19}$$

and there is no other rate constraint violation for the video segment from frame $v+1$ to frame n. In order to avoid the constraint violation as in (4.19), the encoding rates for the video segments from frame m to frame v have to be reduced. Another Lagrangian multiplier λ_v has to be included in (4.13) to account for the constraint as:

$$\hat{\mathcal{X}} = \arg\min_{\mathcal{X}} \left(\sum_{j=m}^{n} D_{\boldsymbol{X}(j)}(j)\right) + \lambda_v \cdot \left(\sum_{j=m}^{v} R_{\boldsymbol{X}(j)}(j)\right) + \lambda_n \cdot \left(\sum_{j=m}^{n} R_{\boldsymbol{X}(j)}(j)\right) \tag{4.20}$$

From (4.14) it is equivalent to setting λ'_i as:

$$\lambda'_i = \begin{cases} \lambda'_v = \lambda_v + \lambda_n, & \text{when } m \le i \le v; \\ \lambda'_n = \lambda_n, & \text{when } v+1 \le i \le n. \end{cases} \tag{4.21}$$

The lower bound of λ'_v (defined as Λ'_v) which prevents the violation of constraint at frame v can be found by using similar bisection search technique on the video segment from frame m to frame v as:

$$\Lambda'_v = \arg\min_{\lambda'_v} \sum_{j=m}^{v} \left(D_{\boldsymbol{X}(j)}(j) + \lambda'_v \cdot R_{\boldsymbol{X}(j)}(j) \right) \tag{4.22}$$

The optimal choices of quantizers are searched again where the multipliers $\lambda'_m, \ldots, \lambda'_v$ are lower-bounded as:

$$\begin{aligned} x^*(i,g) = \arg\min_{\substack{x(i)\in\mathcal{Q} \\ \lambda'_i \geq \Lambda'_i}} & d_{x(i,g)}(i,g) + \lambda'_i \cdot r_{x(i,g)}(i,g) \\ & \forall (i,g) \in \{(m, g_{tx}+1), \ldots, (n, g_{in})\}. \end{aligned} \tag{4.23}$$

Note that from (4.21), $\Lambda'_m = \ldots = \Lambda'_{v-1} = \Lambda'_v$ in above optimization. The search for the optimal quantizer $\mathcal{X}$ and the appropriate multipliers $\{\lambda'_m, \ldots, \lambda'_n\}$ is repeated until the choice of quantizers that does not violate any rate constraints. Refer [64, 24] for detailed description of the algorithm and the proof of optimality.

4.7 Conclusions

In this chapter we considered the rate control problem for the video transmission which requires real-time encoding, decoding and transferring the video data. We considered a slice-based video codec which can achieve lower encoding delay. The delay constraints and the associated rate constraints were more closely investigated at the slice level. With the rate constraints derived as (4.5), we proposed two algorithms based on dynamic programming and Lagrangian optimization to find the optimal choices of quantizers which minimizes the distortion of the video. From (4.5)

we can observe that encoding rate in this scenario is also constrained by the future channel transmission rates. Extending the rate control approaches in this chapter, in next chapter we will look at the problem when channel errors may occur therefore channel transmission rates in (4.5) are unknown.

Chapter 5

Rate Control for Video Transmission over Burst Error Channel

5.1 Introduction

Extending the study on rate control for real-time video transmission in last chapter, in this chapter we will look at the problem of video transmission over unreliable channels. The unreliable channels under study are characterized by their bursty nature with periods of correct transmission alternating with periods of high error rates. Wireless links [65] and the Internet are the two examples of channels that have such burst error characteristics. In this work, we concentrate on how a real time video application can be supported over such a time varying burst-error channel, rather than on the specifics of the physical layer of the channel. We will only assume that the channel behavior can be characterized by a simple burst-error model and will provide experimental results for two such models. In particular we consider a scenario consisting of packet based transmission with Automatic Repeat reQuest (ARQ) error control and a back channel.

In the previous chapter we have shown how the end-to-end delay constraint on the video transmission can be translated into rate constraint at the encoder (see Section 4.3). It can be shown that the applicable rate constraints time depend on future channel rates. However, the exact future channel rates are unknown in an error-prone channel because the effective channel throughput may degrade when error occurs. To overcome this problem in our rate control approach, we propose to use a channel model and channel feedback, which indicates the current channel condition, to statistically estimate the future channel rates as was assumed in the previous chapter. Hence, the rate constraints in this problem setup are expressed in the form of expected future channel rates instead of the real future channel rates. We use an ARQ error control scheme to retransmit the erroneous data packet until data are correctly received. However data loss still may occur if the erroneous packet can not be retransmitted to the decoder within the delay constraint. Uncorrected errors induce decoding errors at the decoder, and may cause significant quality degradation on the reconstructed video. Therefore, the rate control for video transmission in this unreliable transmission environment is formalized as an optimization problem to minimize the video distortion that is caused both by encoding and transmission errors.

Two approaches to utilize the information of channel model and channel feedback into the rate control mechanism are proposed in our research. The first one seeks to minimize the distortion for the *expected rate constraints* given the channel model and current observation. The second approach seeks to allocate bits so as to *minimize the expected distortion* for the given model. We use algorithms based on dynamic programming and Lagrangian optimization to solve the rate control problems. Our simulation results demonstrate that both the distortion of the received video and

the number of data packet loss during the transmission can effectively reduced when the channel feedback and channel model are incorporated in the rate control.

5.2 Channel Error Control

Most video compression schemes are designed to achieve high compression ratio under the assumption that video data is stored and transfered in an error-free environment. When compressed video data is transmitted through noisy channels, transmission errors may result in significant quality degradation on the reconstructed video. This is particularly evident in standard coders such as those based on MPEG-1, MPEG-2, H.261 or H.263, where Variable Length Coding (VLC) and predictive coding, such as motion compensation, are used. In video codecs that use those data compression approaches, the decoder is more likely to lose synchronization with the encoder and decoding errors may propagate through several frames when errors occur in transmission. Several error resilience techniques can be use to enable robust transmission of video data over noisy channels. One approach toward a robust video transmission is to enhance the error-resilience capability in the video codec. Techniques such as re-synchronization header (to re-synchronize the decoder with the syntax of the video bitstream), independently segmented decoding (to stop error propagation) and reversible VLC [66, 67] (to isolate the erroneous bits in a video segment) can be used to remedy the inherit vulnerability of the most video codecs to the errors. A number of researchers also suggest to partition the encoded video data stream and give higher transmission priority to the important information in the video data stream context, such as headers, motion vectors, or low-frequency

DCT components [68, 69, 70]. However it is in general preferable to ensure as error-free a transmission as possible, and error resilience is better done at the underlying channel level before errors affecting the content of the video data.

To provide the required protection on transmitting data one can use error control techniques, which can be roughly categorized into open-loop (e.g., forward error correction, FEC) and closed-loop (e.g., automatic repeat request, ARQ). Obviously error correction comes at the cost of reduced bandwidth available for transmission due to the correction overhead in the FEC case, or the increased delay in the ARQ case. While FEC is often used for wireless mobile channels [71, 72], in a two-way communication system the available feedback channel can be used for error resilience by allowing the receiver to request the retransmission of erroneous packets using ARQ [73]. Using ARQ and other variants of ARQ-based error control (e.g., hybrid ARQ) for the mobile radio channels has been recently proposed as an alternative to a purely FEC based approach [74, 75, 76]. In [77, 78], ARQ feedback is also used for error concealment of the transmitted video. ARQ approaches, assuming the existence of a back channel and sufficiently long end-to-end delays, are appealing in that retransmission is only required during periods of poor channel conditions. Thus ARQ schemes are inherently variable rate, and the effect channel throughput is adaptive to the real-time channel condition. However, to take full advantage of the error control capabilities of an ARQ scheme, we propose to *combine the ARQ feedback mechanism with the rate control mechanism at the video encoder.* By combining the ARQ feedback with a rate control algorithm at the encoder one can achieve an intuitively appealing result: the rate for the encoded video is reduced during the periods of poor channel conditions.

We concentrate on a selective-repeat (SR) ARQ scheme where packets are continuously transmitted without waiting to receive acknowledgments of previously transmitted packets. In the SR ARQ scheme, the reception of a packet is acknowledged by the receiver by sending either an acknowledgement (ACK) or a negative acknowledgement (NAK) to the transmitter. Only the erroneous packets are retransmitted. A time-out mechanism is used so that, if the feedback information is corrupted, data is retransmitted anyway. Packets that have not been sent are stored in the ARQ buffer until they are acknowledged. Packets that have been sent are stored in the encoder buffer. The decoder buffer can be used to rearrange the received packets which are out-of-order due to retransmission. The diagram of the buffers in the communication system with ARQ error control is depicted in Fig. 5.1. Because video transmission is subject to a delay constraint as discussed, the retransmission of any packet is attempted only when its due time has not been exceed. Data losses occur during the channel fading intervals whenever the data cannot be retransmitted before its due time. Therefore in a delay constrained video transmission application, the effectiveness of ARQ may be reduced [79]. However substantial gain in the bandwidth utilization in an ARQ system can be achieved compared to that in an FEC system when no data re-transmission is required during the period that the channel is in good condition. In our research, we focus on the communication system with ARQ error control, and use probabilistic models of the channel behavior which can be applied to Formulations 4 and 5.

5.3 Rate Control Approaches

Under the ARQ scheme, the effective data throughput for the erroneous packets is equivalent to 0 bit, because the data carried by those erroneous packets are discarded

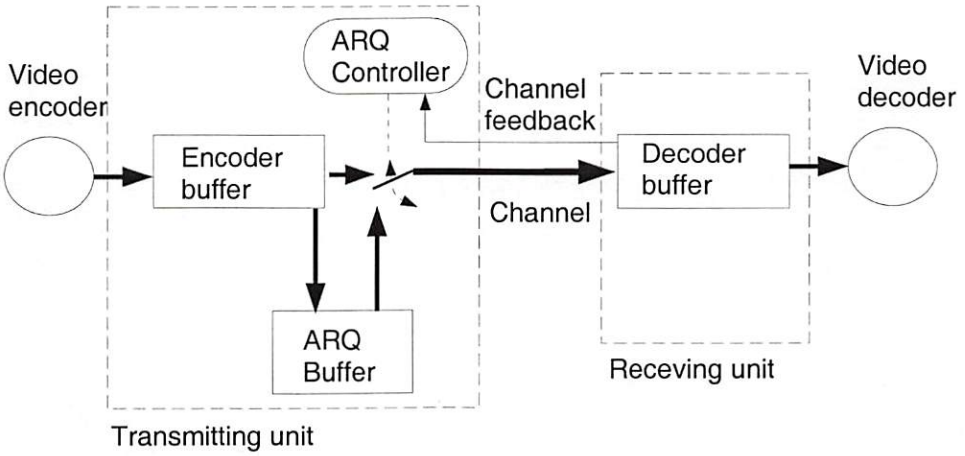


Figure 5.1: Diagram of buffers in the system.

at the receiver and are subject to be retransmitted by the following packets. We define $C(k)$ as the number of bits effectively transmitted by the k-th packet. If the packet size is fixed, then $C(k)$ simply equals to $\bar{C}$ or 0 depending on whether the k-th packet is successfully transmitted or not.

Thus, in our system the effective channel rate $C(k)$ can be either the nominal packet payload $\bar{C}$ (if the packet is received correctly) or zero (if an error occurred). Therefore we cannot guarantee that the rate constraints of (4.5) will not be violated, and thus that no losses will occur, because this would require knowledge of the future channel transmission rates $C(k)$ for packets k, where $k > p = \lfloor \frac{t}{T_p} \rfloor$ (note that we defined packet p as the currently transmitted packet at time t in previous chapter). In this chapter we show how to make use of a probabilistic model for the channel and observations of the current channel state in the context of this rate control problem.

In the above ARQ based system model, we will assume that BL (backlog) bits in the buffer are used to store packets that have been transmitted but not yet acknowledged. Assuming that the delay in receiving acknowledgements is constant and equal to ΔT_b. The feedback delay thus is equivalent to $b = \lceil \frac{\Delta T_b}{T_p} \rceil$ packet intervals and we will need $BL = b \times \bar{C}$ bits to store the b packets that are waiting to

be acknowledged. Since in the worst case all b packets will have to be retransmitted, we will take into account this BL bits backlog data in deriving our rate constraints. The rate constraint (4.4) thus becomes:

$$BL + r'(m, g_{tx}) + \sum_{j=m}^{i} R(j) \leq \sum_{k=p+1}^{\lfloor u_i/T_p \rfloor} C(k), \quad \text{where} \quad t = p \times T_p \tag{5.1}$$

Note that in (5.1) the feedback delay b does not necessary to be constant. In an ARQ system, however, b can be easily measured from the delay in receiving the acknowledgement. Since the rate constraints formulated in (5.1) are for a sliding window of the video segment at time t, the necessary backlog bits BL can be derived from the real-time measurement of the effective feedback delay.

The rate constraints (4.5) for all the slice data inside the encoder buffer now are:

$$\begin{aligned} R(m) &\leq \left[\sum_{k=p+1}^{\lfloor u_m/T_p \rfloor} C(k) \right] - r'(m, g_{tx}) - BL \\ R(m+1) + R(m) &\leq \left[\sum_{k=p+1}^{\lfloor u_{m+1}/T_p \rfloor} C(k) \right] - r'(m, g_{tx}) - BL \\ &\vdots \\ R(n) + \ldots + R(m+1) + R(m) &\leq \left[\sum_{k=p+1}^{\lfloor u_n/T_p \rfloor} C(k) \right] - r'(m, g_{tx}) - BL \end{aligned} \tag{5.2}$$

to include the BL bits backlog in the rate constraint formulation.

We propose two alternative formulations, which both assume that, given the observation and the *a priori* model, estimates of future channel behavior can be obtained. Our first approach consists of modifying Formulation 2 so as to use *expected rate constraints.* given the current state of the buffer. In the second approach, we instead minimize the *expected distortion*, where the distortion of a given block depends not only on the choice of quantizer but on the probability that the block is

lost. In what follows we will denote $S(p)$ as the channel condition when packet p is transmitted at time t. Because of the feedback delay ΔT_b, at time t the latest observation of the channel state will be $S(p-b)$, where $b = \lceil \Delta T_b / T_p \rceil$ is the delay (in number of packets) with which we obtain feedback information. For example, if $b = 0$ the decoder would know immediately whether transmission in the prior time slot was successful. Typically we will assume $b > 0$ since the encoder has to wait for acknowledgements from the decoder to be received in order to determine whether transmission was successful.

Assuming that the encoder can estimate the expected value of the future channel rates (as will be discussed), we can replace the rate constraints in (4.5) by their expected values:

$$\sum_{j=m}^{i} R_{\boldsymbol{X}(j)}(j) \leq E\left[\sum_{k=p+1}^{\lfloor u_i/T_p \rfloor} C(k) \mid S(p-b)\right] - r'(m, g_{tx}) - BL, \quad \forall i \in \{m, \ldots, n\} \tag{5.3}$$

so that our problem can be formulated as:

Formulation 4 Rate control under estimated rate constraints

Find the optimal quantizer choices $\mathcal{X}^*$ at time t such that,

$$\mathcal{X}^* = \arg\min_{\mathcal{X}} \sum_{j=m}^{n} D_{\boldsymbol{X}(j)}(j), \tag{5.4}$$

subject to the expected rate constraints:

$$\sum_{j=m}^{i} R_{\boldsymbol{X}(j)}(j) \leq \left(\sum_{k=p+1}^{\lfloor u_i/T_p \rfloor} E[C(k) \mid S(p-b)]\right) - r'(m, g_{tx}) - BL, \quad \forall i \in \{m, \ldots, n\} \tag{5.5}$$

In the above formulation data loss caused by exceeding the delay constraints may still happen even if the encoding rates $\boldsymbol{R}_{\boldsymbol{X}(m)}(m), \ldots, \boldsymbol{R}_{\boldsymbol{X}(n)}(n)$ meet all the expected constraints (5.3), because the actual channel rates may be lower than the predicted values. Since data loss may result in significant distortion in the decoded video, it may be better to replace the average expected rate in (5.3) by, say, the future rates which are guaranteed with probability 90%. This will obviously result in a more conservative rate allocation at the encoder and hence higher distortion at the decoder. This would be a form of effectively trading off the source rate distortion (sending fewer bits) for the distortion due to losses (if fewer bits are sent they are more likely to be received correctly). This trade-off can be made explicit if we assume that the distortion incurred by data loss can be estimated. Then, an alternative rate control approach will seek to minimize the "expected" distortion, which combines the distortion caused by encoding and that caused by data loss.

More specifically, denote $d(i,g)$ as the encoding distortion of slice (i,g) if that slice is received correctly, and $d_0(i,g)$ as the incurred distortion on slice (i,g) when that slice is lost. Let $p_{loss}(i,g)$ be the probability that slice (i,g) does not arrive at the decoder in time. This will happen if the data of this slice and previous slices can not be transmitted by the time u_i. That is, slice (i,g) will be lost if:

$$BL + r'(m, g_{tx}) + \sum_{j=m}^{i-1} R(j) + \sum_{h=1}^{g} r(i,h) > \left(\sum_{k=p+1}^{\lfloor u_i/T_p \rfloor} C(k) \right). \tag{5.6}$$

The loss probability, which has to be estimated from the observation of channel state $S(t-b)$, can be defined as:

$p_{loss}(i,g) =$

$$\Pr\left[BL + r'(m, g_{tx}) + \sum_{j=m}^{i-1} R(j) + \sum_{h=1}^{g} r(i,h) > \left(\sum_{k=p+1}^{\lfloor u_i/T_p \rfloor} C(k)\right) \mid S(p-b)\right] \quad (5.7)$$

Then the expected value of the distortion for slice (i, g) can be defined as:

$$E\left[d(i,g) \mid S(p-b)\right] = (1 - p_{loss}(i,g)) \times d(i,g) + p_{loss}(i,g) \times d_0(i,g) \quad (5.8)$$

and the expected distortion of frame i thus is:

$$E\left[D(i) \mid S(p-b)\right] = \sum_{j=1}^{G} E\left[d(i,j) \mid S(p-b)\right] \quad (5.9)$$

The optimal rate control problem can be reformulated as:

Formulation 5 Rate Control for Minimum Expected Distortion

Find the optimal quantizer choices $\mathcal{X}^*$ at time t such that,

$$\mathcal{X}^* = \arg\min_{\mathcal{X}} \sum_{j=m}^{n} E\left[D_{\boldsymbol{X}(j)}(j) \mid S(p-b)\right] \quad (5.10)$$

5.4 Probabilistic Modeling of Channel Behavior

The formulations we propose are very general and do not rely on any specific characteristics of the statistical channel behavior. However, the available solutions may differ substantially depending on the specific channel characteristics. Indeed, for channels with random (rather than bursty) errors, the proposed real time feedback approach may not provide any gains in performance, as compared to a closed loop FEC approach. We now present specific parameters and models for the burst error channels that will be used in our experiments. While the optimization techniques

to be presented later have general applicability, we focus our discussion on the case of burst-error channels.

5.4.1 Physical Channel Layer

The channel under consideration is a wireless CDMA spread spectrum system [80] in the mobile transmission environment [72], where channel errors tend to occur in bursts during channel fading periods. The wireless channel consists of two radio links, namely uplink (mobile-to-base) and downlink (base-to-mobile). The encoded video bitstream is packetized into constant-size packets for transmission. On the uplink transmitter, the user data is first spread using a 1/3 convolutional encoder. The encoded bits are then further spread using 64-ary orthogonal signaling followed by symbol interleaving and QPSK PN spreading. The receiver employs both antenna and multipath diversity where a number of correlators (each correlator corresponds to a path) are assigned to each antenna. A fast closed loop power control is used to combat Rayleigh fading. On the downlink transmitter, user information is encoded using a half-rate convolutional encoder and the encoded bits are QPSK spread and transmitted using BPSK modulation. Spatial diversity through the use of different antennas (3 in our case) is used to combat fading. Also, the signal transmission is pilot assisted. Using the pilot signal received at the receiver, maximal ratio combining of different paths is achieved. The combiner is followed by de-interleave and soft decision Viterbi decoder. Refer [81, 75] for more detailed description of the uplink and downlink transceivers.

As indicated in the introduction, we explore a closed loop error control scheme based on ARQ. The delay resulting from retransmission is explicitly accounted for at the encoder (so no retransmissions are attempted if a certain video slice can no longer

be used by the decoder.) Note that one could resort to an interleaved FEC scheme, such as that defined in ITU-T recommendation I.363 [36], where error resilience is also obtained by introducing extra delay. However, to effectively spread out the clustered error bits, the degree of interleaving may have to be significant and thus results in long interleaving delay. Instead, here we choose to use shorter interleaving periods (see transceiver description above and in [81, 75]). What we model is the resulting error probability for each of the data packets (after processing).

5.4.2 Channel models

Previous studies [82, 83] show that a first-order Markov chain, such as the two-state Markov model [84, 85] or a finite-state model [86] provide a good approximation in modeling the error process at the packet level in fading channels. Here we use a *two-state Markov model* and a *N-state Markov model* to emulate the process of packet errors. Note that the transition probabilities of the two models are chosen such as to have the same overall probability of error, although the average burst lengths will be different.

Two-state Markov model: In this model, the channel switches between a "good state" and a "bad state", s_0 and s_1, respectively: packets are transmitted correctly when the channel is in state s_0, and errors occur when the channel is in state s_1 [1] p_{ij}, for $i, j \in \{0, 1\}$, are the transition probabilities (see Fig. 5.2).

[1] More general classes of two-state Markov models can also be used, where for example each state in the model has associated a different probability of error.

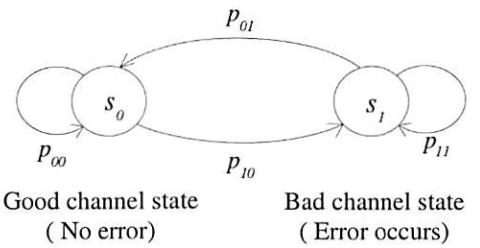


Figure 5.2: Two state Markov channel model.

The transition probability matrix for this two-state Markov channel then can be set up as:

$$\boldsymbol{P} = \begin{bmatrix} p_{00} & p_{01} \\ p_{10} & p_{11} \end{bmatrix} \tag{5.11}$$

N-state Markov model: We use a more simplified model than the more general finite-state Markov model as described in [86]. In this N-state model, introduced in [42, 23] (see Fig. 5.3), the channel states are defined as s_n, $n = 0, \ldots, N-1$ in which s_0 represents the "good state" and all other states represent the "bad states". When the channel is in state s_n, $n \in \{0, \ldots, N-2\}$, the transition of the channel state is either to the next higher state or back to state s_0 based on the status of the currently received data-frame. If the channel is in state s_{N-1}, it will always return to state s_0. With this model, it is only possible to generate burst errors of at most length $N-1$.This N-state Markov channel is depicted in Fig. 5.3, and Table. 5.1 shows sets of transition probabilities that are used to emulate a downlink and uplink wireless CDMA spread spectrum system that were discussed in Section 5.4.1.

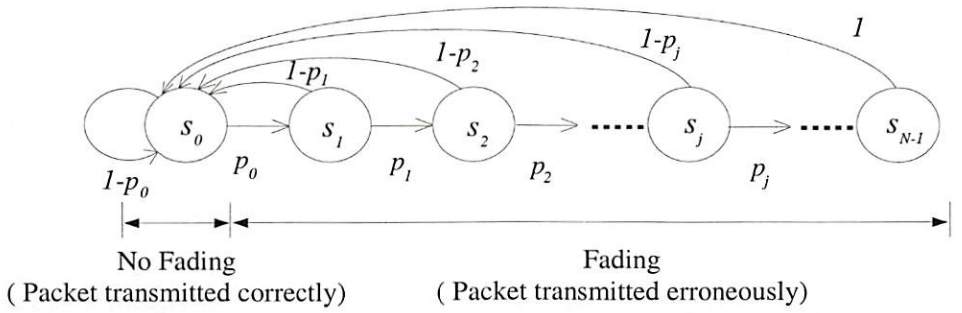


Figure 5.3: N-state Markov channel model.

Define $p_n = \Pr(s_{n+1}|s_n)$ as the transitional probability from state s_n to s_{n+1}. The transition probability matrix for this N-state Markov chain model can be set up as:

$$\boldsymbol{P} = \begin{bmatrix} 1-p_0 & p_0 & 0 & 0 & \cdots & 0 \\ 1-p_1 & 0 & p_1 & 0 & \cdots & 0 \\ 1-p_2 & 0 & 0 & p_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & 0 \\ 1-p_{N-2} & 0 & 0 & 0 & \cdots & p_{N-2} \\ 1 & 0 & 0 & 0 & \cdots & 0 \end{bmatrix} \tag{5.12}$$

The state transition probabilities for the uplink and downlink channels at BER = 10^{-3} are shown in the following table, where N was found to be 14 and 5 (equivalent to maximum burst error lengths of 65 msec and 20 msec) for the downlink and the uplink channel, respectively. These values are found by matching the parameters of the Markov chains to simulations of the transceivers.

	Downlink	Uplink
p_0	0.001469	0.064292
p_1	0.516068	0.100324
p_2	0.778388	0.164083
p_3	0.854118	0.149606
p_4	0.936639	0.526316
p_5	0.873529	0.000000
p_6	0.905724	
p_7	0.881041	
p_8	0.831224	
p_9	0.893401	
p_{10}	0.863636	
p_{11}	0.717105	
p_{12}	0.853211	
p_{13}	0.763441	
p_{14}	0.000000	

Table 5.1: Transitional probability for the downlink and uplink channels

5.5 Channel Rate Estimation

Assuming that at time t when packet p is transmitted, the channel state $S(p-b)$ is known from the channel feedback. We now describe how to estimate the average rates from Formulation 4.

Two-state channel model: In this two-state Markov channel model with transition probabilities (5.11), define state probabilities:

$$\boldsymbol{\pi}(k \mid S(p-b)) = [\pi_0(k \mid S(p-b)), \pi_1(k \mid S(p-b))] \tag{5.13}$$

as the probabilities for the channel to stay in state s_0 and s_1 respectively at time when packet k is transmitted given that the channel initially stays in state $S(p-b)$

when packet $p-b$ is transmitted. The initial state probability $\boldsymbol{\pi}(p-b \mid S(p-b))$ given the fact that channel stays in state $S(p-b)$ can be set up as:

$$\forall\, n \in \{0,1\}, \quad \pi_n(p-b \mid S(p-b)) = \begin{cases} 1, & \text{when } S(p-b) = s_n; \\ 0, & \text{otherwise.} \end{cases} \tag{5.14}$$

In the Markov model, the state probabilities $\boldsymbol{\pi}(k \mid S(p-b))$ when packet k is transmitted can be derived from the state probabilities $\boldsymbol{\pi}(k-1 \mid S(p-b))$ of the previous packet transmission and the transition probability matrix $\boldsymbol{P}$ as:

$$\boldsymbol{\pi}(k \mid S(p-b)) = \boldsymbol{\pi}(k-1 \mid S(p-b)) \cdot \boldsymbol{P} \tag{5.15}$$

By recursively using (5.15), the channel state probabilities when packet k, $k > p-b$, is transmitted can be calculated from $\boldsymbol{\pi}(p-b \mid S(p-b))$ and $\boldsymbol{P}$ as:

$$\boldsymbol{\pi}(k \mid S(p-b)) = \boldsymbol{\pi}(p-b \mid S(p-b)) \cdot \boldsymbol{P}^{k-p+b} \tag{5.16}$$

In our channel model, packets are transmitted correctly (i.e., $\bar{C}$ bits are transmitted) when the channel stays at state s_0, while errors occur when the channel in state s_1 (i.e., 0 bits are transmitted). Therefore $\pi_0(k)$ and $\pi_1(k)$ are, respectively, the probabilities of correct and incorrect transmission of packet k. The expected channel rate $E[C(k) \mid S(p-b)]$ given the observation of channel state $S(p-b)$ can be calculated as:

$$E[C(k) \mid S(p-b)] = \bar{C} \times \pi_0(k \mid S(p-b)) \tag{5.17}$$

and thus the sum of expected channel rate in (5.3) can be written as

$$\sum_{k=p+1}^{\lfloor u_i/T_p \rfloor} E[C(k) \mid S(p-b)] = \bar{C} \cdot \sum_{k=p+1}^{\lfloor u_1/T_p \rfloor} \pi_0(k \mid S(p-b)) \tag{5.18}$$

N-state channel model: A similar approach can be used to derive the expected channel rates from the N-state Markov channel model. Define state probability $\pi_n(k \mid S(p-b))$ as the probability given that the channel is in state s_n when packet k is transmitted given that the channel state observation $S(p-b)$, and $\boldsymbol{\pi}(k \mid S(p-b)) = [\pi_0(k \mid S(p-b)), \pi_1(k \mid S(p-b)), ...\pi_{N-1}(k \mid S(p-b))]$. The initial state probability $\boldsymbol{\pi}(p-b \mid S(p-b))$ given the fact that the channel stays in state $S(p-b)$ when packet $p-b$ is transmitted can be set up as:

$$\forall\, n \in \{0, \ldots, N-1\}, \quad \pi_n(p-b \mid S(p-b)) = \begin{cases} 1, & \text{when } S(p-b) = s_n; \\ 0, & \text{otherwise.} \end{cases} \tag{5.19}$$

The state probability vector $\boldsymbol{\pi}(k)$ when packet k is transmitted can also be calculated by using (5.16) with the transition probability matrix $\boldsymbol{P}$ defined as (5.12), and the expected channel rates can be obtained as (5.17).

5.6 Expected Distortion

From (5.7), the probability of losing the slice (i, g), $p_{loss}(i, g)$ depends on the accumulated rate of slice (i, g) and previous encoded slices which are stored in the encoder buffer, and on the future channel rates. We define $B(i, g) = BL + r'(m, g_{tx}) + \sum_{j=m}^{i-1} R(j) + \sum_{h=1}^{g} r(i, h)$ as the accumulated rate of those video data, which are in the encoder buffer, up to slice (i, g). Therefore based on the channel state observation $S(p-b)$ when packet $p-b$ is transmitted, we define a time-varying probability

distribution function $\Phi_{i,g}(B(i,g), t \mid S(p-b))$ of losing data of slice g of frame i at time t, with the accumulated encoding rate $B(i,g)$ as variable, as:

$$\Phi_{i,g}(B(i,g), t \mid S(p-b)) = \Pr\left[B(i,g) > \left(\sum_{k=p+1}^{\lfloor u_i/T_p \rfloor} C(k)\right) \mid S(p-b)\right]. \quad (5.20)$$

Given the accumulated encoding rate $B(i,g)$, the encoder can estimate the distortion $d(i,g)$ as:

$$\begin{aligned} E\left[d(i,g)\right] &= \left(1 - \Phi_{i,g}(B(i,g), t \mid S(p-b))\right) \times d(i,g) \\ &\quad + \Phi_{i,g}(B(i,g), t \mid S(p-b)) \times d_0(i,g) \end{aligned} \quad (5.21)$$

The function $\Phi_{i,g}(B(i,g), t \mid S(p-b))$ can be derived from the channel model and channel observation as follows.

Two-state channel model: For any given value $B(i,g)$, define η as the number of packets needed for transmitting those $B(i,g)$ bits of data as:

$$\eta = \left\lceil \frac{B(i,g)}{\bar{C}} \right\rceil \quad (5.22)$$

where $\bar{C}$ is the packet size. Then the probability that $\sum_{k=p+1}^{\lfloor u_i/T_p \rfloor} C(k)$ is smaller than $B(i,g)$ is equivalent to the probability that less than η packets are successfully transmitted during the time interval from now (time t) to the transmitting time constraint u_i, or during the transmission of packet $p+1$ to packet $\lfloor u_i/T_p \rfloor$.

Define $q_{n,r}(p,k)$, where $n \in \{0,1\}$ and $r \leq k-p$, as the probability that, beginning from the transmission of packet $p+1$, the channel visits state s_0 (successful packet transmission) r times and arrives at state s_n at the time when packet k is transmitted. Because we are counting the number of packets that are successfully

transmitted (i.e., r) beginning from the transport of packet $p+1$, hence r is initialized as 0 when $k = p$, i.e., $q_{n,r}(p,p) = 0$, $\forall r \neq 0$. Given the observed channel state $S(p-b)$ when packet $p-b$ is transmitted, $q_{n,0}(p,p)$ can be initialized as:

$$q_{n,0}(p,p) = \pi_n(p \mid S(p-b)) \quad \forall n \in \{0,1\}. \tag{5.23}$$

The value of $q_{n,r}(p,k)$, $p+1 \leq k \leq \lfloor u_i/T_p \rfloor$, can be obtained recursively from the Markov model as:

$$\forall\, r: \quad 0 \leq r \leq (k-p-1),$$
$$\begin{cases} q_{0,r}(k) = q_{0,r}(k-1) \cdot p_{00} + q_{1,r}(k-1) \cdot p_{10} \\ q_{1,r+1}(k) = q_{0,r}(k-1) \cdot p_{01} + q_{1,r}(k-1) \cdot p_{11} \end{cases}$$

with

$$\begin{aligned} q_{0,0}(p,p+1) &= \pi_0(t \mid S(p-b)) \cdot p_{00} + \pi_1(t \mid S(p-b)) \cdot p_{10} \\ q_{1,1}(p,p+1) &= \pi_0(t \mid S(p-b)) \cdot p_{01} + \pi_1(t \mid S(p-b)) \cdot p_{11} \end{aligned} \tag{5.24}$$

Since transmitting $B(i,g)$ bits of data requires that at least η packets be transmitted, $\Phi_{i,g}(B(i,g), t \mid S(p-b))$ can be written as

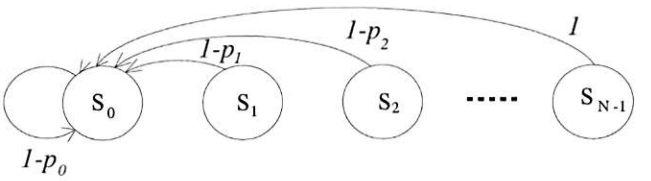
$$\Phi_{i,g}(B(i,g), t \mid S(p-b)) = \sum_{r=0}^{\eta-1} \Big(q_{0,r}(p, \lfloor u_i/T_p \rfloor) + q_{1,r}(p, \lfloor u_i/T_p \rfloor) \Big). \tag{5.25}$$

N-state channel model: In this N-state Markov model, we define $q_{n,r}(p,k)$, $n \in \{0,1,\ldots,N-1\}$, as the probability that, begins from the transmission of packet $p+1$, the channel visits state s_0 (successful packet transmission) r times and arrive at state s_n at the time when packet k is transmitted. Initially when $k = p$,

$q_{n,0}(p,p)$ can be initialized with the estimated state probabilities $\pi_n(p \mid S(p-b))$ given the observed channel state $S(p-b)$ as:

$$q_{n,0}(p,p) = \pi_n(p \mid S(p-b)), \quad \forall n \in \{0,\ldots,N-1\}. \tag{5.26}$$

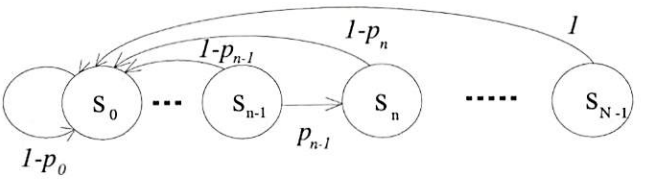
The value of $q_{n,r}(p,k)$ can be obtained recursively from the Markov chain model as: When $n = 0$:



$$q_{0,r}(p,k) = \sum_{n=0}^{N-1}(1-p_n)\cdot q_{n,r}(p,k-1) \tag{5.27}$$

(The channel transitions to state s_0 and no error occurs.)

When $n = 1,\ldots,N-1$:



$$q_{n,r+1}(p,k) = p_{n-1}\cdot q_{n-1,r}(p,k-1), \quad \forall s \neq 0 \tag{5.28}$$

(The channel transitions to states other than s_0 and error occurs.)

Therefore the value of the probability distribution function $\Phi_{i,g}(B(i,g), t \mid S(p-b))$ for a given value $B(i,g)$ is

$$\Phi_{i,g}(B(i,g), t \mid S(p-b)) = \sum_{r=0}^{\eta-1} \sum_{n=0}^{N-1} q_{n,r}(p, \lfloor u_i/T_p \rfloor), \quad \text{where } \eta = \left\lceil \frac{B(i,g)}{C} \right\rceil \tag{5.29}$$

Note that for both models, since the number of states is discrete and the transition probabilities are known *a priori*, it is possible to use tables to generate the relevant constraints and probabilities from the channel observations.

5.7 Encoding Rate Selection under Estimated Rate Constraints

The rate control problem of Formulation 4 can be solved using the estimated rate constraints from our two channel models. Here we solve the rate allocation problem using the algorithms, namely dynamic programming and Lagrangian optimization, which were proposed in Section 4.5 and 4.6.

5.7.1 Dynamic programming

When the estimated rate constraint can be derived in (5.1), we can use the dynamic programming approach described in Section 4.5 to solve the rate control problem formulated in Formulation 4 except that the condition for a valid state $\mathcal{S}_{i,g}(B)$ used in (4.10) now is replaced by:

$$\mathcal{S}_{i,g}(B) \quad \text{is valid if:} \quad B \leq E\left[\sum_{k=p+1}^{\lfloor u_i/T_p \rfloor} C(k)\right] \tag{5.30}$$

The algorithm is described as follows:

Step 0: Initialize the decoder buffer fullness B as $r'(m, g_{tx}) + BL$. Each node in the trellis at stage (i, g) is defined by the state $\mathcal{S}_{i,g}(B)$ to represent a set of quantizer choices that result in accumulated rate B up to slice (i, g). Start the loop from slice $(m, g_{tx} + 1)$.

Step 1: At stage (i, g), add all possible branches to the end of every surviving path node $\mathcal{S}_{i,g-1}(B)$ at stage $(i, g-1)$. The new state in next stage is $\mathcal{S}_{i,g}(B + r_{x(i,g)}(i, g))$, given that quantizer $x(i, g)$ is used to encode the slice (i, g) and $\mathcal{S}_{i,g}(B + r_{x(i,g)}(i, g))$ is a valid state as defined in (5.30), i.e., where

$$B + r_{x(i,g)}(i, g) \leq E\left[\sum_{k=p+1}^{\lfloor u_i/T_p \rfloor} C(k)\right] \tag{5.31}$$

Step 2: For all the branches arriving at node $\mathcal{S}_{i,g}(B)$, keep only the one with smallest aggregate distortion $\sum_{j=m}^{i-1} D_{\boldsymbol{X}(j)}(j) + \sum_{h=1}^{g} d_{x(i,h)}(i, h)$ and prune out the others. The smallest aggregate distortion path is the surviving path for that state.

Step 3: Increment the stage to next slice and go to Step 1 until the last slice (slice (n, g_{in})) in the encoder buffer.

Step 4: At stage (n, g_{in}), find out the state transitions with smallest aggregate distortion. The corresponding choices $\mathcal{X}^*$ are the best quantizers rates choices for slices in the encoder buffer, given that the expected rate constraint (5.3) which is estimated at time t.

5.7.2 Lagrangian optimization

The Lagrangian optimization discussed in Section 4.6 can be used to solve the rate allocation problem defined in Formulation 5. The multiple rate constraints now are derived from the estimated channel rates as (5.5). The algorithm can be summarized as follows:

Step 0 Initially the quantizer choices $\hat{\mathcal{X}} = \{\hat{\boldsymbol{X}}(m), \ldots, \boldsymbol{X}(n)\}$ are obtained by using a single Lagrange multiplier λ'_n for all blocks in (4.17), subject to only one constraint:

$\sum_{j=m}^{n} R_{\boldsymbol{X}(j)}(j) \leq E\left[\sum_{k=p+1}^{\lfloor u_n/T_p \rfloor} C(k)\right] - r'(m, g_{tx}) - BL.$

Step 1 If $\hat{\mathcal{X}}$ is such that all rate constraints in (5.3) are met, then $\hat{\mathcal{X}}$ is the optimal solution $\mathcal{X}^*$ for Formulation 3. Otherwise, assume that frame v is the "last" frame which violates the rate constraint (i.e., $v < n$ and no other frame between frame $v+1$ and frame n violates the rate constraint). Find the minimum value of Lagrange multiplier $\Lambda'_v = \min \lambda'_v$ for the video segment from frame m to frame v which just prevents violation of the rate constraint:

$\sum_{j=m}^{v} R(j) \leq E\left[\sum_{k=p+1}^{\lfloor u_v/T_p \rfloor} C(k)\right] - r'(m, g_{tx}) - BL.$

Step 2 Find the quantizer choices $\hat{\mathcal{X}} = \{\hat{\boldsymbol{X}}(n), \ldots, \hat{\boldsymbol{X}}(n)\}$ as in Step 0 except that the Lagrange multiplier for the video segment from frame m to frame v is lower-bounded by Λ'_v as $\lambda'_v \longleftarrow \max(\Lambda'_v, \lambda'_v)$.

Step 3 Go to Step 1. Repeat until all the rate constraints in (5.3) are met.

5.8 Encoding Rate Selection for Minimum Expected Distortion

The dynamic programming approach discussed in Section 4.5 can be used to find the optimal quantizer choices formulated in Formulation 5. To use a dynamic programming framework to minimize the expected distortion, the cost associated with state $S_{i,g}(B)$ is the sum of the expected distortions $\sum_{j=m}^{i} E\left[D_{\boldsymbol{X}(j)}(j) \mid S(t-b)\right]$ along the path. From (5.20) we can observe that the loss probability $\Phi_{i,g}(B, t \mid S(t-b))$ only depends on the accumulated encoding rate B. Since each state $S_{i,g}(B)$ is uniquely defined by its accumulated encoding rate B, we can associate a unique loss probability $\Phi_{i,g}(B, t \mid S(t-b))$ to each state, and this independently of future quantization choices. Therefore all the paths that arrive at the same state $S_{i,g}(B)$ will have the same loss probability no matter what were their previous states. Thus the optimality principle also applies in this case and paths that are sub-optimal (higher expected distortion) up to a given state are also guaranteed to be sub-optimal overall. We can solve the problem using dynamic programming as described before, with the only modification that the branch cost is now the expected distortion, rather than the deterministic distortion due to coding as in Section 4.5.

However, the Lagrangian optimization approach can not be used here as the rate-control algorithm because the choice of quantizers for other video blocks can affect the value of expected of distortion $E\left[D_{\boldsymbol{X}(j)}(j)\right]$. To be more specific, if the problem is formulated as that of finding the quantizer choice $x(i)$ to minimize the cost function $J_{i,g}(\lambda_i', x(i,g)) = E[d_{x(i,g)}(i,g)] + \lambda_i' \cdot r_{x(i,g)}(i,g)$, the choice of previous encoding rate may affect the value of $E[d_{xi,g}(i,g)]$ (since it determines B) and thus the optimization can not be achieved independently for each block as in (4.13).

5.9 Experimental Results and Conclusions

In order to asses the effectiveness of the proposed rate-control algorithms, we implement those algorithms in a simulated burst-error transmission environment based on the models we described for downlink and uplink channels in Section 5.4.2. We provide the simulation results for the following rate control algorithms:

- **DP-Est:** The algorithm based on dynamic programming with estimated rate constraint as formulated in Formulation 4,
- **LAG-Est:** The algorithm based on Lagrangian optimization to the same problem introduced in Section 4.6.
- **DP-Min:** The algorithm based on dynamic programming to minimizing the expected distortion as formulated in Formulation 5,
- **DP-No Feedback:** The dynamic programming approach with an average rate constraints is also used in the case when no knowledge of the channel is available; in this case the video encoder assumes the average rate is available (i.e., $\bar{C} \times P_e$, where P_e is the probability of packet loss),
- **DP-Adv:** Finally, we also consider the unrealistic scenario where the encoder has advance knowledge of the future channel rates. This gives us an indication of the loss in performance due to imperfect channel knowledge in the other algorithms.

In each experiment we generate error patterns and the results we provide are averaged over several realizations of the channel error patterns. In each case the encoder has knowledge of the statistical model of the channel behavior and makes use of it in the rate control algorithm. Table 5.9 summarizes the characteristics of the

	Downlink Channel		Uplink Channel	
	Two-state	N-state	Two-state	N-state
Pr(Good state)	0.9940	0.9940	0.9328	0.9328
Pr(Good $\rightarrow$ Bad)	0.001035	0.001469	0.03382	0.06429
Pr(Bad $\rightarrow$ Good)	0.1720	0.2442	0.46945	0.8924
Avg. burst length (packets)	5.8136	4.0950	2.1302	1.1205

Table 5.2: Summary of the characteristics of the channels used in our experiments

various models used in our simulations. Note that average losses for the downlink channels are smaller but the corresponding burst durations are also longer. The uplink channel behavior we simulate with the 2-state Markov model is very close to being a channel with uniformly distributed losses, since the average burst length is close to 1.

The video test sequence "Susie" (first 100 frames) is used in our experiments. The input sequence is in QIF format (176 $\times$ 144 pixels for each frame), and is encoded with an H.261 encoder at quantization step sizes chosen from four values: 12, 14, 20 and 30. The H.261 encoder is used in Intra mode, which allows us to allocate quantizers independently to each frame. In the QCIF format, each frame is subdivided into macroblocks (MB) with size 16 $\times$ 16 pixels. Therefore each frame consists of 99 (11 $\times$ 9) macroblocks. In our simulation, we select the frame rate such that the duration of 3 MB's equals to one packet transmitting interval. Every three MB's are grouped together as a slice with a single quantizer being assigned to each video block. A packet is transmitted by the channel every 5 msec with 41 bytes payload, thus video data is transmitted at the rate of about 6 frames/sec on average. In our simulations we assume that $b = 2$ (i.e., the state of the channel is known with a delay of two packet intervals).

Our results are summarized in Figs. 5.4, 5.5 for the N-state Markov model and Figs. 5.6, 5.7 for the two state Markov model. We provide results of both PSNR[2] and packet loss rates. We plot our average distortion and loss results for different end-to-end delay values.

Based on our experimental results it is easy to see that the performance, as is to be expected, improves as we increase the information available about the channel state. Thus, performance when no feedback is given (**DP-No Feedback**) is worse than in the case where real time feedback and a channel model are available, which in turn has worse performance than the case where future rates are known.

It can also be seen that the approach based on expected distortion (**DP-Min**) generally outperforms the expected rate approaches (**DP-Est** and **LAG-Est**). In general the distortion due to packet losses will be much higher than that due to using a coarse quantizer so, even in the system based on expected distortion, the rate control algorithms will tend to minimize the packet losses. We can observe that in all cases (except in the case where no feedback is available, obviously) the distortion and packet loss is reduced when the end to end delay in the system increases. Note in particular that the losses can in some cases made very close to zero.

Finally, it is worth pointing out the difference between uplink and downlink channels. The former has errors that are nearly random and error bursts tend to be very short (of the order of magnitude of the feedback delay) thus the difference in performance between having and not having exact knowledge of the channel rates is relatively modest. This indicates that the dynamics of the channel are too fast with respect to the response time of the rate control, thus most rate control approaches perform similarly (the algorithm without feedback still performs worse because it

[2]The Peak Signal to Noise Ratio for a sequence is defined as $PNSR = 10 * log_{10}(255^2/MSE)$, where MSE is the average Mean Squared Error for the whole sequence

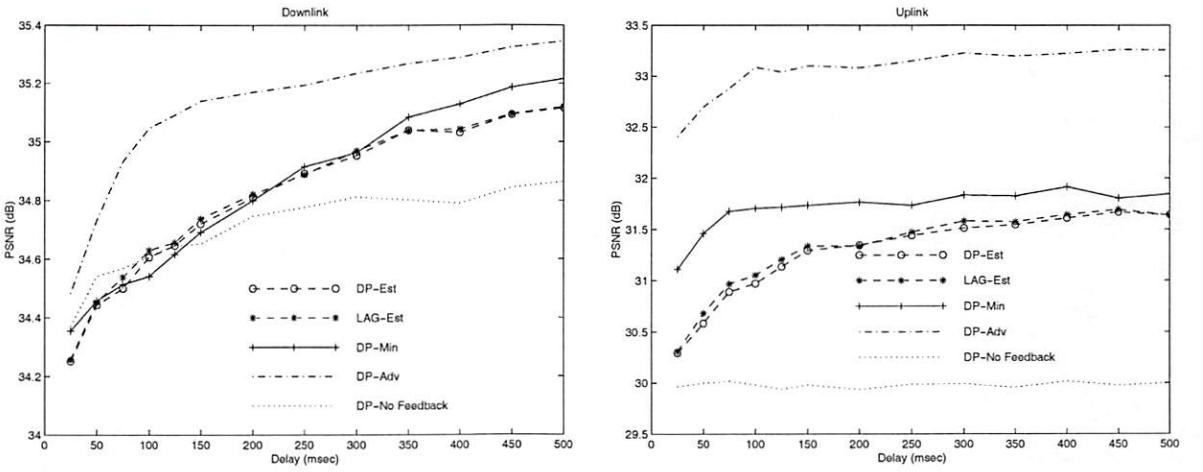


Figure 5.4: N-state Markov channel model: Resulting PSNR of the decoded video by **DP-Est**, **LAG-Est**, **DP-Min** algorithms under end-to-end delay constraint from 50 msec to 400 msec. The results of**DP-Adv** and **DP-No Feedback** algorithms are also shown for benchmarking comparison.

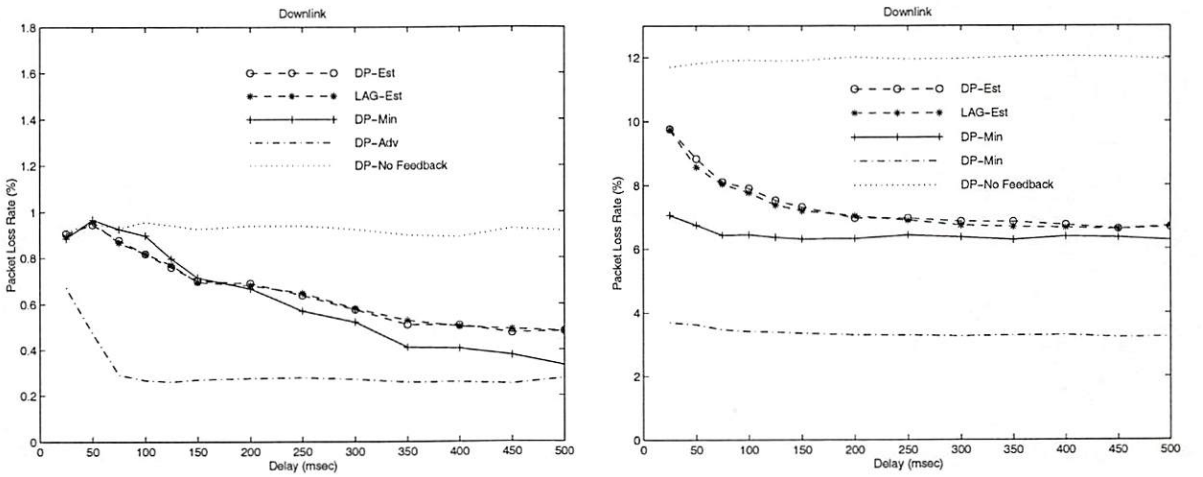


Figure 5.5: N-state Markov channel model: Resulting packet loss rate by **DP-Est**, **LAG-Est**, **DP-Min** algorithms under end-to-end delay constraint from 50 msec to 400 msec. The results of **DP-Adv** and **DP-No Feedback** algorithms are also shown for benchmarking comparison.

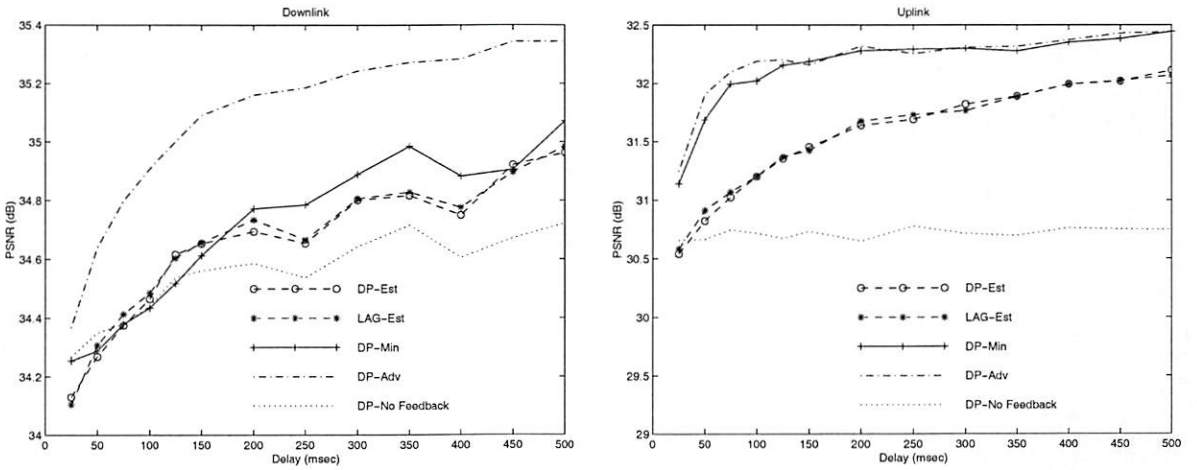


Figure 5.6: Two-state Markov channel model: Resulting PSNR of the decoded video by **DP-Est**, **LAG-Est**, **DP-Min** algorithms under end-to-end delay constraint from 50 msec to 400 msec. The results of **DP-Adv** and **DP-No Feedback** algorithms are also shown for benchmarking comparison.

has no knowledge of the channel state, i.e., data is not recoded when there are channel losses). Note that by comparison exact knowledge of the channel behavior does result in improvements in the downlink channel. This can be justified by the longer average burst sizes and higher variances in burst sizes.

It is also worth noting that the Lagrangian optimization (**LAG-Est**) approach is much faster than the dynamic programming approaches (**DP-Est** and **DP-Min**) and provides comparable performance. This method can thus be a good candidate for a practical implementation of a rate control system.

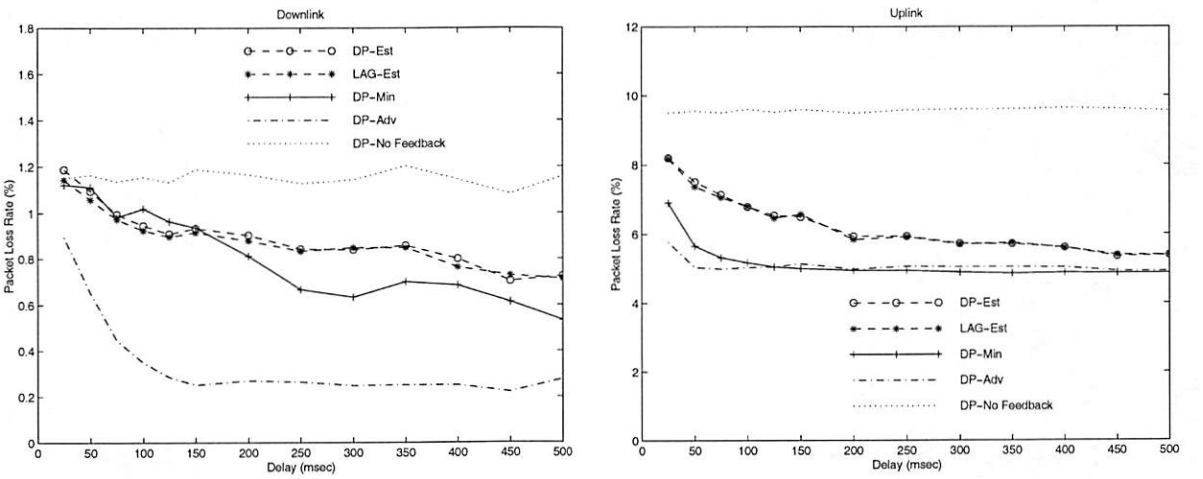


Figure 5.7: Two-state Markov channel model: Resulting packet loss rate by **DP-Est**, **LAG-Est**, **DP-Min** algorithms under end-to-end delay constraint from 50 msec to 400 msec. The results of **DP-Adv** and **DP-No Feedback** algorithms are also shown for benchmarking comparison.

Chapter 6

Conclusions and Extensions

6.1 Summary of the Research

In this research, we investigated the rate control problem for video transmission over different types of transmission channels. We first surveyed different configurations of video transmission applications and associated constraining factors, including the delay and channel transmission throughput. Recognizing that the video encoding rates in video transmission applications are constrained by those factors, we formalized the delay and channel constraints into the constraints of video encoding rates. With the explicitly formulated rate constraints, the rate control problem for video transmission subject to the delay and channel constraints were translated into the encoding rate allocation problem which are subject to the associated rate constraints.

This basic structure of rate control approach was used throughout this research toward the problem of video transmission over various transmission environments. In the problem of video transmission over a constrained-VBR channel, a joint rate-selection scheme was proposed to select the encoding rates and channel rates in the way that the transmitted video quality is maximum. The rates are selected by an algorithm based on dynamic programming, hence the selected encoding rates and

channel rates are the optimal solution toward the formulated rate control problem. This proposed algorithm can provide a benchmarking upper bound of video quality that can be achieved given a set of delay constraint and channel constraint. Hence the algorithm for joint rates selection can be used as a tool to analysis substantial performance gain that can be achieved in the variable bit rate (VBR) transmission environment compared to that in the constant bit rate (CBR) transmission environment.

We also extended the proposed rate control approaches into the problem of video transmission over a burst-error channel. In this time-varying transmission environment, the proposed rate control approach adaptively adjust the encoding rates to the change of the channel condition using the feedback of the channel condition and the information of a priori channel model. We showed that better performance of the video transmission, in terms of lower packet loss rate and better reconstructed video quality, can be achieved by proper integrated source rate control and channel error control.

6.2 Future Extension

In addition to the transmission environment discussed in this study, other possible extensions of our video rate control approaches are listed in the follow:

- **Video encoder with predictive coding:**

  The rate control approaches we have explored so far are based on the assumption that that video frames are "intra-coded". The optimization problem are thus simplified by de-correlated the encoding rate and distortion of each video frame from that of other frames in the video sequence. However, while video

frames are predictive coded, the search of an optimal solution will become extreme complicate because of the dependence that are introduced in encoding rates and distortion of those frames.

In Chapter 3 an iterative approaches has been used to search for a sub-optimal solutions for the rate control problem in which video are predictive coded (see Section 3.5). While the preliminary simulation shows that the proposed iterative approach did converge to a local optimal solution, we feel that this iterative approach is worth of more investigation, and may provide a solution for the rate control problem in commonly used video codecs such as MPEG, where video is predictive encoded.

- **Integration of rate control with other error control scheme:**

  In Chapter 5 we studied the problem of video transmission over a noisy channel, and utilized an ARQ error control scheme for channel error resilience. Nevertheless, other error control schemes, such as Hybrid ARQ, can be used in the system, and in is possible that our rate control approach can be extend to incorporated other rate control schemes.

- **Video transmission with varying channel delay**

  In our formulation of delay constraints, we assumed a constant or a known channel delay. In the real world transmission system this assumption may not be necessary true. However, it is possible to extend the dynamic encoding rate allocation scheme to be adaptive to the variation of channel delay, given that a model for the channel delay exist. This will be a similar approach as we used in Chapter 5 that the rate control utilizes the feedback information about the current channel delay, and adjusts the source encoding rates accordingly.

This extension will be useful for video transmission over Internet, which can be characterized as a transmission channel with significant delay variation.

Reference List

[1] Advanced Television Systems Committee, "ATSC digital television standard," Sep. 1995.

[2] U. Reimers, "Digital video broadcasting," in *IEEE Communications Magazine*, pp. 104–110, J.un 1998.

[3] ISO/IEC 11172, "Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s," 1993.

[4] ISO/IEC 13818, "Information technology – Generic coding of moving pictures and associated audio information," 1996.

[5] ITU-T Recommendation H.261, "Video codec for audiovisual services at p×64 kbit/s," 1993.

[6] ITU-T Recommendation H.263, "Video coding for low bitrate communication," 1996.

[7] Draft ITU-T Recommendation H.263 Version 2, "Video coding for low bit rate communication," Sep. 1997.

[8] A. N. Netravali and B. Haskell, *Digital Picture: Representation and Compression.* New York: Plenum Press, 1988.

[9] M. Liou, "Overview of the px64 kbit/s video coding standard," *Comm. of the ACM*, vol. 34, pp. 59–63, Apr. 1991.

[10] D. LeGall, "MPEG: a video compression standard for multimedia applications," *Communications of the ACM*, vol. 34, pp. 46–58, Apr. 1991.

[11] J. R. Jain and A. K. Jain, "Displacement measurement and its application in interframe image coding," *IEEE Trans. on Comm.*, vol. COM-29, pp. 1799–1808, Dec. 1981.

[12] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishiguro, "Motion-compensated interframe coding for video conferencing," in *Proc. NTC'81*, pp. G.5.3.1–G.5.3.4, 1981.

[13] A. Puri and R. Aravind, "Motion-compensated video coding," *IEEE Trans. on Circ. and Sys. for Video Tech.*, vol. 1, pp. 351–361, Dec. 1991.

[14] R. Srinivasan and K. R. Rao, "Predictive coding based on efficient motion estimation," in *Proc. Intl. Conf. on Communication*, vol. 1, pp. 521–526, 1988.

[15] T. Wiegand, M. Lightstone, D. Mukherjee, T. G. Campbell, and S. K. Mitra, "Rate-distortion optimized mode selection for very low bit rate video coding and the emerging H.263 standard," *IEEE Trans. on Circ. and Sys. for Video Tech.*, vol. 6, pp. 182–190, Apr. 1996.

[16] B. G. Haskell, A. Puri, and A. N. Netravali, *Digital Video: An Introduction to MPEG-2.* Chapman and Hall, 1997.

[17] J. L. Mitchell, W. B. Pennebaker, C. E. Fogg, and D. J. LeGall, *MPEG Video compression standard.* Chapman and Hall, 1997.

[18] J. O. Limb, "Buffering of data generated by the coding of moving images," *Bell Syst. Tech. J.*, vol. 51, pp. 261–289, Jan. 1972.

[19] ISO/IEC/JTC1/SC29/WG11/93-225b, "Test model 5," Sep. 1997.

[20] A. R. Reibman and B. G. Haskell, "Constraints on variable bit-rate video for ATM networks," *IEEE Trans. on Circ. and Sys. for Video Tech.*, vol. 2, pp. 361–372, Dec. 1992.

[21] A. Ortega, K. Ramchandran, and M. Vetterli, "Optimal trellis-based buffered compression and fast approximations," *IEEE Trans. on Image Proc.*, vol. 3, pp. 26–40, Jan. 1994.

[22] C.-Y. Hsu, A. Ortega, and A. R. Reibman, "Joint selection of source and channel rate for VBR video transmission under ATM policing constraints," *IEEE J. on Sel. Areas in Comm.*, pp. 1016–1028, Aug. 1997.

[23] C.-Y. Hsu, A. Ortega, and M. Khansari, "Rate control for robust video transmission over wireless channels," in *Proc. of Visual Communic. and Image Proc., VCIP'97*, (San Jose, California), pp. 1200–1211, Feb. 1997.

[24] C.-Y. Hsu and A. Ortega, "Joint selection of source and channel rate for vbr video transmission under atm policing constraints," in *International Conference on Acoustics, Speech, and Signal Processing, ICASSP'98*, (Seattle, WA), 1998.

[25] C.-Y. Hsu, A. Ortega, and M. Khansari, "Rate control for robust video transmission over burst-error wireless channels," *IEEE J. on Sel. Areas in Comm., Special Issue on Multi-Media Network Radios*, 1998. Accept for publication.

[26] IEEE 809.2a Specification of ISLAN16-T, *Isochronous Services with Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Media Access Control (MAC) Service*, 1995.

[27] M. R. Pickering and J. F. Arnold, "A perceptually efficient VBR rate control algorithm," *IEEE Trans. on Image Proc.*, vol. 3, pp. 527–532, Sep 1994.

[28] T. T. Lakshman, A. Ortega, and A. R. Reibman, "VBR video: trade-offs and potentials," *Proc. of the IEEE*, vol. 86, pp. 952–973, May 1998.

[29] B. G. Haskell, "Buffer and channel sharing by several interframe picturephone coders," *Bell Syst. Tech. J.*, vol. 51, pp. 261–289, Jan. 1972.

[30] B. Maglaris, D. Anastassiou, P. Sen, G. Karlsson, and J. D. Robbins, "Performance models of statistical multiplexing in packet video communications," *IEEE Trans. on Comm.*, vol. 36, pp. 834–843, Jul. 1988.

[31] A. Ortega, M. W. Garrett, and M. Vetterli, "Rate constraints for video transmission over ATM networks based on joint source/network criteria," *Annales des Télécommunications*, vol. 50, pp. 603–616, Jul.-Aug. 1995.

[32] ITU-T Recommendation H.310, "Broadband audiovisual communication systems and terminals," 1996.

[33] ITU-T Recommendation H.321, "Adaptation of H.320 visual telephone terminals to B-ISDN environments," 1996.

[34] S. Okubo, S. Dunstan, G. Morrison, M. Nilsonn, H. Radha, D. L. Skran, and G. Thom, "ITU-T standardization of audiovisual communication systems in ATM and LAN environment," *IEEE J. on Sel. Areas in Comm.*, vol. 15, pp. 965–982, Aug. 1997.

[35] ITU-T Recommandation I.362, "B-ISDN ATM Adaptation Layer (AAL) functional description," Apr. 1991.

[36] ITU-T Recommandation I.363, "B-ISDN ATM Adaptation Layer (AAL) specification," Jun. 1993.

[37] E. P. Rathgeb, "Modeling and performance comparison of policing mechanisms for ATM networks," *IEEE J. on Sel. Areas in Comm.*, vol. 9, pp. 325–334, Apr. 1991.

[38] L. Dittmann, S. B. Jacobsen, and K. Moth, "Flow enforcement algorithms for ATM networks," *IEEE J. on Sel. Areas in Comm.*, vol. 9, pp. 343–350, Apr. 1991.

[39] ITU-T/SG15/LBC-95-267, "Robust H.263 compatible video transmission for mobile applications." National Semiconductor Corporation, 1995.

[40] ITU-T/SG15/LBC-96-026, "Comparison of the two proposals in documents LBC-95-267 and LBC-95-309 relying on FEC, ARQ and NAKs."

[41] ITU-T/SG15/LBC-95-309, "Sub-video with retransmission and intra-refreshing in mobil/wireless environments," Oct. 1995.

[42] A. Ortega and M. Khansari, "Rate control for video coding over variable bit rate channels with applications to wireless transmission," in *Proc. of the 2nd. Intl. Conf. on Image Proc., ICIP'95*, (Washington, D.C.), Oct. 1995.

[43] W. Verbiest, L. Pinnoo, and B. Voeten, "The impact of the ATM concept on video coding," *IEEE J. on Sel. Areas in Comm.*, vol. 6, pp. 1623–1632, Dec. 1988.

[44] J. Y. Hui, "Resource allocation for broadband networks," *IEEE J. on Sel. Areas in Comm.*, vol. 6, pp. 1598–1608, Dec. 1988.

[45] G. de Veciana, G. Kesidis, and J. Walrand, "Resource management in wide-area ATM networks using effective bandwidths," *IEEE J. on Sel. Areas in Comm.*, vol. 13, pp. 1081–1090, Aug. 1995.

[46] H. Kanakia, P. P. Mishra, and A. R. Reibman, "An adaptive congestion control scheme for real-time packet video transport," *IEEE/ACM Trans. on Networking*, vol. 3, pp. 671–682, Dec. 1995.

[47] M. Reisslein and K. W. Ross, "Call admission for prerecorded sources with packet loss," *IEEE J. on Sel. Areas in Comm.*, vol. 15, pp. 1167–1180, Aug. 1997.

[48] ATM Forum, *ATM User-Network Interface Specification, Version 3.0.* Prentice-Hall, 1993.

[49] C.-Y. Hsu and A. Ortega, "Joint encoder and VBR channel optimization with buffer and leaky bucket constraints," in *Symposium on Multimedia Communications and Video Coding*, (Brooklyn, NY), Oct. 1995.

[50] A. R. Reibman and A. W. Berger, "Traffic descriptors for VBR video teleconferencing," *IEEE/ACM Transactions on Networking*, vol. 3, pp. 329–339, Apr. 1995.

[51] H. Harasaki and M. Yano, "A study on VBR coder control under usage parameter control," in *Proc. of Fifth International Workshop on Packet Video*, (Berlin), pp. F2.1–F2.5, Mar. 1993.

[52] M. Kawashima and H. Tominaga, "A study on VBR video transmission under usage parameter control," in *Proc. Fifth International Workshop on Packet Video*, (Berlin), pp. F3.1–F3.5, 1993.

[53] M. Hamdi, J. W. Roberts, and P. Rolin, "Rate control for VBR video coders in broad-band networks," *IEEE J. on Sel. Areas in Comm.*, vol. 15, pp. 1040–1051, Aug. 1997.

[54] J.-J. Chen and D. W. Lin, "Optimal bit allocation for coding of video signals over ATM networks," *IEEE J. on Sel. Areas in Comm.*, vol. 15, pp. 1002–1015, Aug. 1997.

[55] W. Ding, "Joint encoder and channel rate control of VBR video over ATM networks," *IEEE Trans. on Circ. and Sys. for Video Tech.*, vol. 7, pp. 266–278, Apr. 1997.

[56] L.-J. Lin, A. Ortega, and C.-C. Kuo, "Gradient-based buffer control technique for MPEG," in *SPIE Visual Communications and Image Processing, VCIP'95*, (Taipei, Taiwan), pp. 2502–2513, May 1995.

[57] G. D. Forney, "The Viterbi algorithm," *Proc. of the IEEE*, vol. 61, pp. 268–278, Mar. 1973.

[58] A. J. Viterbi and J. K. Omura, *Principles of Digital Communication and Coding.* McGraw-Hill, 1979.

[59] A. Ortega and M. Vetterli, "Multiple leaky buckets for increased statistical multiplexing of ATM video," in *Proc. of the 6th Packet Video Workshop*, (Portland, OR), Sep. 1994.

[60] The Portable Video Research Group, Stanford University, "PVRG-P64 CODEC v. 1.2." Available URL: ftp://havefun.stanford.edu/pub/p64/P64v1.2.tar.Z.

[61] K. Ramchandran, A. Ortega, and M. Vetterli, "Bit allocation for dependent quantization with applications to multiresolution and MPEG video coders," *IEEE Trans. on Image Proc.*, vol. 3, pp. 533–545, Sep. 1994.

[62] The Portable Video Research Group, Stanford University, "PVRG-MPEG." Available URL: ftp://havefun.stanford.edu/pub/mpeg/MPEGv1.2.tar.Z.

[63] D. P. Bertsekas, *Dynamic Programming.* Prentice-Hall, 1987.

[64] A. Ortega, "Optimal rate allocation under multiple rate constraints," in *Data Compression Conference'96*, (Snowbird, Utah), Mar. 1996.

[65] B. Sklar, "Raleigh fading channels in mobile digital communication systems, part I: Characterization," in *IEEE commun. Mag.*, pp. 90–100, Sep. 1997.

[66] G. Wen and J. Willasenor, "A class of reversible variable length codes for robust image and video coding," in *Proc. IEEE Intl. Conf. on Image Proc.,ICIP'97*, vol. 2, (Santa Barbara, CA), pp. 65–68, Oct. 1997.

[67] Y. Takashima, M. Wada, and H. Murakami, "Reversible variable length codes," *IEEE Trans. Commun.*, vol. 43, pp. 158–162, Feb./Mar./Apr. 1995.

[68] M. Ghanbari, "Two-layer coding of video signals for VBR networks," in *IEEE J. on Sel. Areas in Comm.*, vol. 7, pp. 771–781, Jun. 1989.

[69] P. Bahl and I. Chlamtac, "H.263 based video codec for real-time visual communications over wireless radio networks," in *Proc. IEEE ICUPC'97*, (San Diego, CA), pp. 773–779, Oct. 1997.

[70] Y.-C. Chang and D. G. Messerschmitt, "Segmentation and compression of video for delay-flow multimedia networks," in *International Conference on Acoustics, Speech, and Signal Processing, ICASSP'98*, (Seattle, WA), May 1998.

[71] R. Stedman, H. Gharavi, L. Hanzo, and R. Steele, "Transmission of subband-coded images via mobile channels," *IEEE Trans. on Circ. and Sys. for Video Tech.*, vol. 3, no. 1, pp. 15–26, 1993.

[72] Y. J. Liu and Y. Q. Zhang, "A mobile data code division multiple access (CDMA) system with power control and its application to low-bit-rate image transmission," in *Proc. Vehicular Technology Conference, VTC'97*, (Phoenix, AZ), pp. 770–773, May 1993.

[73] S. Lin, D. J. Costello, and M. Miller, "Automatic repeat request error control schemes," *IEEE Communications Magazine*, pp. 5–17, 1984.

[74] M. Zorzi, R. R. Rao, and L. B. Milstein, "ARQ error control for fading mobile radio channels," *IEEE Trans. on Veh. Tech.*, vol. 46, pp. 445–455, May 1997.

[75] M. Khansari, A. Jalali, E. Dubois, and P. Mermelstein, "Low bit-rate video transmission over fading channels for wireless microcellular systems," *IEEE Trans. on Circ. and Sys. for Video Tech.*, pp. 1–11, Feb. 1996.

[76] H. Liu and M. E. Zarki, "Performance of H.263 video transmission over wireless channels using hybrid ARQ," *IEEE J. on Sel. Areas in Comm.*, vol. 15, pp. 1775–1786, Dec. 1997.

[77] N. Färber, E. Steinbach, and B. Girod, "Robust H.263 compatible video transmission over wireless channels," in *Proc. PCS'96*, (Melbourne), pp. 575–578, Mar. 1996.

[78] E. Steinbach, N. Färber, and B. Girod, "Standard compatible extension of H.263 for robust video transmission in mobile environments," *IEEE Trans. on Circ. and Sys. for Video Tech.*, vol. 7, pp. 872–881, Dec. 1997.

[79] M. Zorzi and R. R. Rao, "ARQ error control for delay-constrained communications on short-range burst-error channels," in *Proc. IEEE Vehicular Technology Conference, VTC'97*, (Phoenix, AZ), May 1997.

[80] TIA/EIA/IS-95, *Mobile-station-base-station compatibility standard dual-mode wideband spread spectrum cellular system*, 1993.

[81] M. Khansari, A. Jalali, E. Dubois, and P. Mermelstein, "Robust low bit-rate video transmission over wireless access systems," in *Proc. of ICC'94*, vol. 1, pp. 571–575, May 1994.

[82] H. S. Wang, "On verifying the first-order Markovian assumption for a Reyleight fading channel model," in *Proc. IEEE ICUPC'94*, (San Diego, CA), pp. 160–164, Sep. 1994.

[83] M. Zorzi, R. R. Rao, and L. Milstein, "On the accurancy of a first-order Markov model for data transmission on fading channels," in *Proc. IEEE ICUPC'95*, (Tokyo, Japan), Nov. 1995.

[84] E. N. Gilbert, "Capacity of burst-noise channel," *Bell Syst. Tech. J.*, vol. 39, pp. 1253–1265, Sep. 1960.

[85] E. O. Elliott, "Estimates of error rates for codes on burst-noise channels," *Bell Syst. Tech. J.*, vol. 42, pp. 1977–1997, Sep. 1963.

[86] B. D. Fritchman, "A binary channel characterization using partitioned Markov chains," *IEEE Trans. Info. Theory*, vol. IT-13, no. 2, pp. 221–227, 1966.