

Copyright © 1998, by the author(s).  
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

**DELAY COGNIZANT VIDEO CODING**

by

Yuan-Chi Chang

Memorandum No. UCB/ERL M98/66

8 December 1998

COVER

**DELAY COGNIZANT VIDEO CODING**

by

Yuan-Chi Chang

Memorandum No. UCB/ERL M98/66

8 December 1998

**ELECTRONICS RESEARCH LABORATORY**

College of Engineering  
University of California, Berkeley  
94720

Delay Cognizant Video Coding

by

Yuan-Chi Chang

B.S. (National Taiwan University) 1991  
M.S. (University of California, Berkeley) 1996

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

in

Engineering - Electrical Engineering and Computer Sciences

in the

GRADUATE DIVISION  
of the  
UNIVERSITY OF CALIFORNIA, BERKELEY

Committee in charge:

Professor David G. Messerschmitt, Chair  
Professor David N. C. Tse  
Professor Stanley A. Klein

Fall 1998

**Delay Cognizant Video Coding**

**Copyright 1998**

**by**

**Yuan-Chi Chang**

## **Abstract**

### **Delay Cognizant Video Coding**

by

**Yuan-Chi Chang**

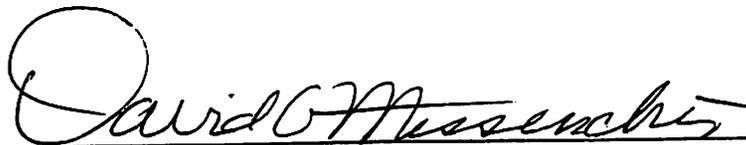
**Doctor of Philosophy in Engineering  
Electrical Engineering and Computer Sciences**

**University of California, Berkeley**

**Professor David G. Messerschmitt, Chair**

Two-way interactive video imposes a great challenge on the design of real-time multimedia communication systems because of its high bit rate and stringent delay constraints. While text-based communications like chatroom on the Internet will be upgraded with voice over IP (Internet Protocol), it is expected interactive video to be the next multimedia service enabling closer collaboration among physically distant parties. In this dissertation, we investigate how delay critical video applications may be supported with low perceptual delay and efficient network resource utilization. Our proposed solution is delay cognizant video coding (DCVC), a layered coding algorithm that takes advantage of differential delay transport services. DCVC segments and compresses video information into multiple layers, which are mapped to flows in the network. These flows have differential delay requirements and are carried by networks recognizing and making use of flows with relaxed delay, which are also less resource demanding and hence lower cost. At the decoder, flows are asynchronously composed to form displayed image frames. We describe one implementation of DCVC, whose segmentation is based on spatio-temporal frequency variation of video region. We found both its compression performance and quality to be competitive to state-of-the-art coders. We

demonstrate potentially significant advantages to apply DCVC to various networking environments including the current and next generation Internet as well as wireless. We show DCVC may be used to improve quality, increase network video capacity and strengthen error robustness in those networking environments. We conducted subjective testing to assess that DCVC after asynchronous rendering has comparable quality to traditional synchronous video. To investigate the network transport of DCVC flows, we explore link layer control mechanisms in wireless networks to support differential quality of service (QoS) of flows. Finally, we point out the link between DCVC and other layered video coding and propose a path to integrate them into a fully adaptive video coder.

 12/8/98  
Professor David G. Messerschmitt, Chair Date

# Contents

---

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Video Coding for Delay Critical Network Multimedia	3
1.2	Key Concepts of Delay Cognizant Video Coding	7
1.3	Multi-flow Transport Architecture	11
1.4	Dissertation Overview	15
<b>2</b>	<b>Delay Cognizant Coding Architecture</b>	<b>18</b>
2.1	Objective	19
2.2	Prior Work on Delay Segmentation	22
2.2.1	Mean-square Difference of Low-frequency Subband	22
2.2.2	Block Moving Distance	23
2.2.3	Spatio-temporal Subband Coefficient Variation	24
2.3	Encoding Algorithm	26
2.3.1	Segmentation Stage: Spatio-temporal Block Variation	26
2.3.2	Compression Stage: Motion Estimation with DCT	31
2.3.3	Motion Estimation, Transform Coding and Quantization	33
2.3.4	Area Filling	35
2.3.5	Rate Control	36
2.4	Decoding Algorithm	37
2.5	Summary	39
2.6	Appendix A: Optimal Flow Rate Assignments	40
<b>3</b>	<b>DCVC Applications</b>	<b>51</b>
3.1	Video on Broadband ISDN	53
3.1.1	Improving Video Quality	55
3.1.2	Increasing Video Capacity	58
3.2	Video on Delay/Cost Differentiated Network	59
3.3	Internet Video	61

3.4	Wireless Video .....	64
3.5	Summary .....	68
3.6	Appendix A: Effective Rate .....	69
<b>4</b>	<b>Quality Evaluation .....</b>	<b>71</b>
4.1	Background .....	72
4.2	Video Fidelity Characterization .....	77
4.3	Video Quality Characterization .....	78
4.3.1	Subjective Quality Evaluation .....	78
4.3.2	Computational modeling methods .....	82
4.4	Dynamic Noise Incurred Quality Degradation .....	84
4.5	Summary .....	88
4.6	Appendix A: Subjective Evaluation Procedure .....	88
4.7	Appendix B: Three-way Repeated Measures Analysis of Variance ...	91
4.8	Appendix C: Pseudo Codes for Generating the Double-ring Pulsing Circle Stimulus .....	93
<b>5</b>	<b>Power Control and Scheduling on CDMA Wireless .....</b>	<b>95</b>
5.1	Background .....	97
5.2	Scheduling Problem of Fixed-Rate Links .....	100
5.2.1	Power Control Feasibility Test .....	100
5.2.2	Rate Admissible Region .....	102
5.2.3	Scheduling Arbitrary Arrivals .....	104
5.2.4	Scheduling i.i.d Bernoulli Arrivals .....	106
5.3	Scheduling Problem of Multi-Rate Links .....	108
5.4	Matched Filter Receivers .....	110
5.4.1	Scenario 1 .....	111
5.4.2	Scenario 2 .....	112
5.4.3	Scenario 3 .....	114
5.4.4	Scenario 4 .....	115
5.5	Multiuser MMSE Receiver .....	117
5.5.1	Scenario 1 .....	120
5.5.2	Scenario 2 .....	122

5.5.3 Scenario 3 .....	123
5.6 Summary and Future Work .....	124
5.7 Appendix A: Proof of NP-completeness of the MC-CDMA Admission Control Problem .....	126
5.8 Appendix B: Proof of the Optimality of Longest Queue First Policy	127
<b>6 Future Work .....</b>	<b>134</b>
6.1 QoS Adaptive Video Coding .....	135
6.2 QoS Adaptive Network Control .....	138

# Acknowledgments

I came to Berkeley not because of great enthusiasm in engineering but because I wanted to find out how smart people can be! At Berkeley, I found myself surrounded by top talents not only from the U.S. but also from the rest of the world. It makes me feel smarter by learning, studying, and working with them. While brilliant people are all around and clever ideas are dime a dozen, I found the brightest scholars to be polite, humble, and open minded. I feel very fortunate to come to know and work with many of them. My graduate education and research would not be possible without their advice and encouragement.

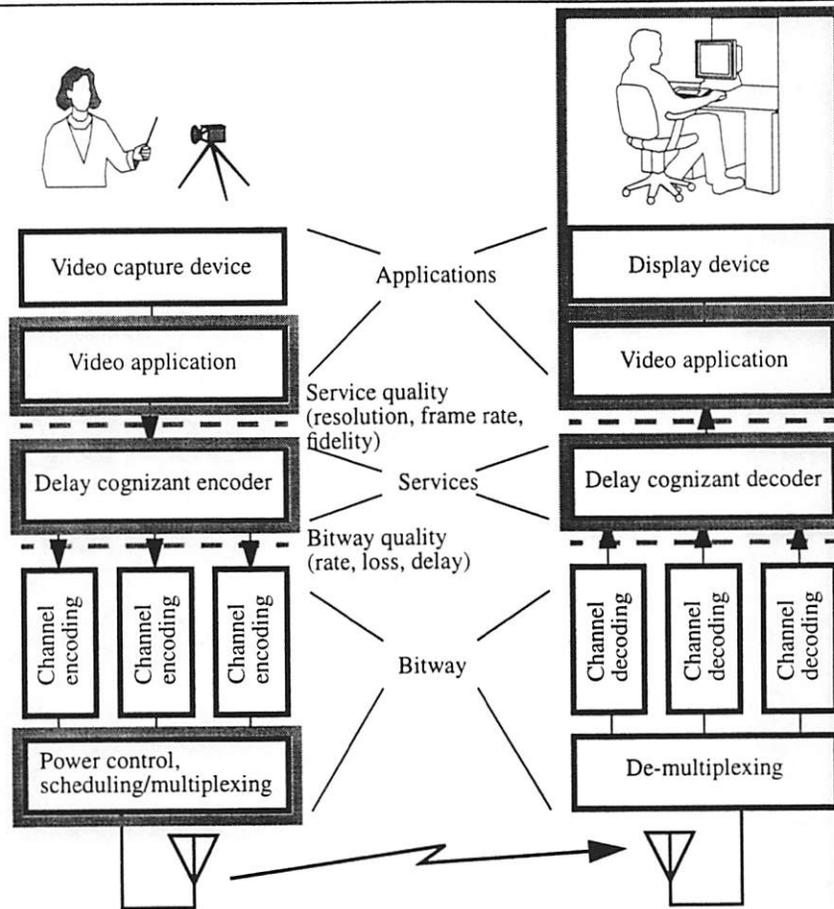
I am very grateful to my research advisor, Professor David Messerschmitt, for his guidance and support throughout my years at Berkeley. Besides the wealth of his technical knowledge, I truly admire his profound business insights and superior management skills. I would also like to thank Professor David Tse for his dedication and patience in advising my wireless project. It has been a privilege to work with him. I would like to thank two great scientists, Professor Stanley Klein and Dr. Thom Carney, for introducing me to the world of vision science. Dr. Carney's dedication to scientific research is inspiring. I would also like to thank Professor Avidah Zakhor and Professor J. George Shanthikumar for many comments and feedback before, during, and after my qualifying examination. For two years I was supported by an IBM Corporate Fellowship Award. Dr. Marc Willebeek-LeMair was kind to be my mentor to make the award possible. I would like to express my sincere gratitude to him, and other IBM researchers whom I had the luxury to work with. They include Dr. Zon-Yin Shae, Dr. Dilip Kandlur, and Dr. Chung-Sheng Li.

I feel really lucky to have many wonderful friends at Berkeley. I wish to thank them for the help, encouragement and fun time together. Finally, I would like to dedicate this dissertation to my parents, whose care and love supported me throughout my education. Their encouraging letters and phone calls helped me

work more diligently toward graduation. I would also like to congratulate my younger brother, who is getting Ph.D in Physics from the renowned Oxford University at about the same time as I do. Job well done, Brother!

# 1

## Introduction



Non-real-time multimedia communications, represented by the prevalence of the World Wide Web (WWW) and electronic mail, has been the main driving force for computer networking. While network data search and retrieval programs such as *gopher*, *archie*, *ftp* and *telnet*, existed before WWW was invented, it is the excitement intrigued by *image*-laden Web pages that started the revolution. While non-real-time communications still dominates the majority of Internet traffic, sev-

## Box 1.A Chapter-front Block Diagram

On the first page of each chapter, a video communication block diagram is shown with blocks marked with grey frames. The marked blocks are related to the theme of the chapter. The block diagram points out the chapter's theme in the system context.

The left half of the diagram belongs to the transmitting endpoints and the right half belongs to the receiving endpoints. The blocks are categorized based on their functionalities into three layers: *applications*, *services* and *bitway*. The bitway layer establishes connections, carries data between endpoints, and controls the bitway quality (rate, loss, delay) of connections. The services layer calls upon the bitway layer to provide a set of common generic capabilities to the application layer. Unlike other communication system diagram that typically connects blocks with a single connection, in this diagram the delay cognizant encoder applies multiple connections due to their differential requirements on bitway quality. [32] has a more detailed discussion on this three layer network architecture.

eral text-based real-time communication programs such as *talk* and *chatroom*, are very popular among Internet users. These programs offer only text but not full multimedia because of the limited access bandwidth available to end users. As the history of WWW demonstrated that the appeal of multimedia draws the crowd, adding multimedia content, and in particular, video, to real-time networking communications will arguably become the driving force of the next wave of networking revolution.

In this dissertation, we consider the design of a video coding algorithm for delay-critical (real-time) video applications in a networking environment with differential latency services. Delay-critical network video applications such as videoconferencing, remote learning, video editing and video chatroom, require low end-to-end latency. While the ultimate limitation on delay is governed by the speed of light, the low latency requirement poses a great challenge to both video coding and network transport. In today's telecommunication infrastructure, the two largest service networks, POTS (Plain Old Telephone System) for voice and Internet for data, offer vastly different service quality and prices. The former offers low latency connections suitable for delay critical applications but they are very costly. The latter provides data connections at low cost, but they have high loss and large latency. The existence of the two grade-of-service networks (with more coming) prompts

us to develop a coding algorithm that adapts to their differential delay/cost characteristics. In the future, taking advantage of high latency, low cost connections is even more crucial on unreliable links such as wireless because a low latency, highly reliable connection consumes much more network resources.

This dissertation describes our proposed coding algorithm, which is named Delay Cognizant Video Coding (DCVC) for its cognizance of differential delay services. DCVC performs segmentation and compression of video based on the delay requirements. It minimizes the amount of video traffic carried by the lowest delay connection, which is most costly. The rest of the video data is put on higher-delay carriers to take advantage of the low connection costs. While earlier video compression standards such as MPEG1 and MPEG2 [50] were designed for playback from storage media, the latest videoconferencing standards like H.261 and H.263 [2] do not take advantage of the differential delay network services, either. The multi-connection (later referred as flows) paradigm permits applications to make a better trade off between perceptual quality and cost. In addition to that, these differential delay video flows enabled a new slew of applications to improve video quality and to increase video capacity.

As the chapter front block diagram indicated, this chapter introduces issues addressed in the scope of the thesis. We start by elaborating the motivations to develop this new coding for delay critical network multimedia. Key concepts of DCVC are then introduced. DCVC assumes a differential delay flow transport architecture, whose reasoning is detailed next. And finally the organization of the dissertation is presented.

## **1.1 Video Coding for Delay Critical Network Multimedia**

Multimedia communications has been playing an increasingly important role in keeping the growth of the World Wide Web and the Internet. Starting with simple images and animations and later moving to streaming video and audio, multimedia content on the Web serves both purposes of information access and

entertainment. According to the Web server access statistics at the EECS department of UC Berkeley, image transmissions take 80% of the total access bit rate for that server. Most of the Internet applications, however, are non-real-time despite that a few real-time communication tools, like *talk* and *chatroom*, are very popular. Both *talk* and *chatroom* are text-based applications and thus do not require much bandwidth. As the history of WWW showed that the appeal of multimedia draws the crowd, adding multimedia content, and video in particular, to real-time networking communications shall also demonstrate the same benefit. Besides the obvious need to upgrade access bandwidth for carrying video, real-time communications requires timely delivery of information, which implies that end-to-end delay is an important quality metric. To envision the new challenges of provisioning delay critical network video, we shall first examine today's telephony communication networks.

An International Telecommunication Union (ITU) study concluded that telephony users find round-trip delays of greater than 300 ms more like a half-duplex connection than a conversation [1]. The 150 ms one-way end-to-end latency is an upper bound on the sum of various latency contributors such as propagation delay, queueing delay, and processing delay. In a global network, just the propagation delay alone, which is limited by the speed of light, can be as much as 90 ms on optical fiber. The current telephony network achieves low latency by applying a circuit switching architecture, in which a connected session is guaranteed a fixed bit rate. Since the service rate is guaranteed and the capacity is limited, the telephony network exercises admission control to regulate the number of active sessions.

Unlike telephony networks, the Internet uses packet switching, in which a connected session is serviced at non-guaranteed variable bit rates. Internet does not have admission control to prevent congestion. As a result, one measurement of Internet latency reported the average one-way delay can be as much as 100 ms between two nodes in the continental US [41]. Long delay and variable bandwidth

## **Box 1.B Marketing Trials for Voice Over IP**

At the time of writing, several Internet service providers, including AT&T, Sprint, and PSInet, have announced marketing trials for voice over IP services. In the cases of AT&T and Sprint, subscribers from San Francisco can call anywhere in the continental US for 7.5 cents a minute, seven days a week. No computers are required to sign up the program. Subscribers call a telephone gateway with the access number similar to a calling card call. The gateway digitizes voice and routes the voice packets through the companies' own network running IP protocols. A gateway at the remote end reassembles the voice packets, converts the digital signal to analog, and connects to the receiver's phone number in the usual way. The author subscribed to the Sprint service and found the voice quality almost indiscriminately from toll call quality. For more details, visit the web sites at <http://www.att.com/connectsave/> for AT&T's offering and <http://www.psinet.com/voice/> for PSInet's offering.

on the public Internet pose great challenges to the companies planning on voice over IP (Internet Protocol) services. Voice over IP (VOIP) was motivated by the potential cost reduction of combining voice networks and data networks into an integrated service network. However, in order to overcome the problem of long delay, network carriers use their own private IP networks and employ techniques like differentiating voice and data packets and/or assigning high priority to voice traffic. Because of the preferential treatment, the cost of VOIP services remains high. (Please see Box 1.B on page 5 for a description on VOIP marketing trials.) Compared to the flat rate Internet access with rates in the range of \$20/month, VOIP calls with rates like 7c/minute, are orders-of-magnitude more expensive. Although both VOIP and data services are carried by IP networks, they provide vastly different service quality and prices. The former offers low latency but costly connections suitable for delay critical applications. The latter provides high loss, large latency but much cheaper connections.

While VOIP services are being deployed, we claim efficient provisioning of *delay critical network video* over IP poses a even bigger challenge for the following reasons:

1. Video, even after compression, requires a much higher bit rate than voice. The commonly applied voice compression operates in the range of 6 to 20 kilobits

per second. The bit rate of compressed video, depending on the resolution, quality, and frame rate, can fall in between 10 kilobits per second to several megabits per second. Most commercial videoconferencing tools, like PictureTel and Intel ProShare, operate at the ISDN rate of 128 kilobits per second.

2. Voice and video must be perceptually synchronized, so that at least a portion of video has the same low delay requirement as voice traffic has. That portion needs to be transmitted with voice traffic.
3. Compressed video is very bursty ( $> 15: 1$ , peak-to-average ratio) and with the stringent delay constraint, the bursts cannot be smoothed. This implies that a far higher bit rate than the long-term average is required at times. Compared to the fixed rate circuit switching networks, statistical bandwidth sharing (statistical multiplexing) on the packet switching network can more efficiently carry the bursty video traffic. However, even if bandwidth reservation need not be made at the peak rate, it may still be much higher than the average rate.
4. On packet networks, delay jitter is introduced in the process of switching and routing packets. However, the traditional video display model of frame-by-frame synchronous reconstruction requires jitter-free data alignment. The network-introduced jitter is often removed by an artificial buffer at the receiver, introducing another contribution to delay. This further tightens the end-to-end network delay budget. We believe synchronous video reconstruction, which requires zero jitter, and packet switching networks, which introduces jitter, are fundamentally a mismatch. A new model for video coding, one which matches well to the characteristics of packet networking, is needed.

Acknowledging the above challenges, we believe a new type of video coding needs to be developed to take advantage of the differential delay services. Should the conventional video coding model be followed as is, its compressed bit-stream could only be transmitted along with voice on the most expensive connec-

tion. The bursty, bandwidth consuming video will then significantly increase the price of *video over IP* services.

Our proposed solution to take advantage of the differential delay network is delay cognizant video coding, to be described next.

## 1.2 Key Concepts of Delay Cognizant Video Coding

Delay cognizant video coding (DCVC) is a new type of layered coding algorithm for delay critical network video applications. Abandoning the implicit assumption in conventional video coding that every bit in a video stream has the same delay requirement, a DCVC encoder segments video information into multiple flows (layers) with differential delay requirements. This has two implications:

1. The lowest delay flow, which determines the perceptual delay, consumes only a portion of the total traffic. (Please see Box 1.C on page 8 for the definition of perceptual delay.)
2. Bursty traffic in the flows with relaxed delay requirements (other than the lowest delay flow) can be smoothed, resulting in more efficient use of network resources.

Both implications would result in a reduction of bit rate usage and a lowered cost.

A DCVC decoder applies a different mode of video reconstruction from the traditional, synchronous approach. Since video packets from the transmitter are

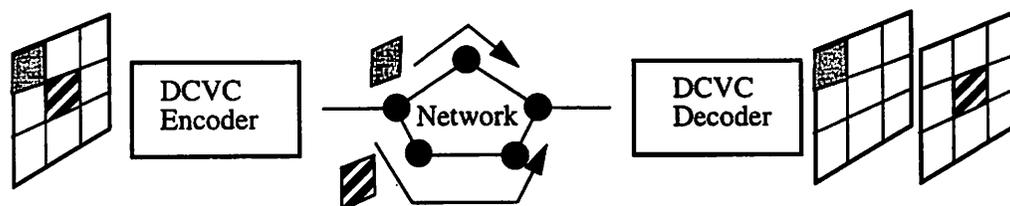


Figure 1.1 An illustration of differential delay flows and asynchronous reconstruction; the top path is shorter and thus the block arrives earlier than the other taking the bottom path

## Box 1.C Perceptual Delay

Perceptual delay is a subjective measure of delay in video communications. It can be quantified by the end-to-end latency of perceiving a motion event, such as nodding or gestures, happening at the transmitting end. It can also be characterized by the relative delay to the associated audio signal for maintaining lip sync. The former definition may apply without the presence of audio channel. Both definitions are measured by subjective evaluations. The longer the perceptual delay, the poorer the interactivity of the application.

carried by differential delay flows, they arrive at the receiver with different latencies. Rather than performing re-synchronization and suffering the resultant delay penalty, the decoder renders the data immediately upon arrival. Figure 1.1 illustrates an example of this asynchronous reconstruction, in which two video blocks acquired at the same instant in the transmitting terminal are assigned to different delay flows, arrive at the receiver at different instants, and are displayed immediately. In the illustrated example, delay segmentation is performed on blocks.

Properly segmenting video information into flows is crucial because the segmentation has a significant impact on perceived quality. To minimize visible artifacts in asynchronous reconstruction, DCVC assigns the most visually significant information to the lowest delay flow and the less visually significant information to higher delay flows. Visual significance is, at the early visual processing level, characterized by the spatio-temporal masking properties of the human visual systems (HVS), and at the cognizance level, characterized by image recognition and understanding. In our current DCVC design, we mainly exploit HVS masking. Conventional single-flow video compression (such as MPEG [50] and H.261/H.263 [2]) may be viewed as a special case of DCVC with two flows: one has the minimal (finite) delay and the other has the maximal (infinite) delay. The second flow simply never arrives and attributes to the quantization loss. DCVC adds more flows between these extremes to improve perceptual quality without appreciably adding to network resource consumption because the new flows have relaxed delay.

DCVC has a number of potential advantages over traditional, single flow video such as MPEG and H.261. It effectively addresses the four challenging issues discussed in the previous section. These potential advantages, qualitatively described here with quantifications in following chapters, include taking advantage of differential delay/cost networks, increasing video traffic capacity, reducing perceptual delay, and flexibly trading off traffic capacity and perceptual delay. More precisely:

1. DCVC reduces the traffic in the lowest delay flow without increasing perceptual delay. Traffic reductions come from lowering both the average bit rate as well as the magnitude of the traffic bursts (peak rate). For delay critical network video, average bit rate alone does not accurately reflect the required bit rate because of the dynamic nature of video. Reducing the size of the bursts is also important. Since the lowest delay connection is most expensive, reducing its carried traffic is beneficial in reducing cost.
2. DCVC puts a significant amount of data to higher delay flows. Traffic in these flows can be smoothed to further reduce the magnitude of the bursts, thereby increasing the effectiveness of statistical multiplexing. In general, higher delay flows are also less costly than the lowest delay flow.
3. DCVC can efficiently make use of the residual bit rate, which is the difference of the *effective bandwidth* [28] and the average rate, of a connection. Effective bandwidth was proposed to characterize the reserved bit rate at a network switch for stochastic bursty sources. When multiple bursty traffic flows merge at a switch, the residual bit rate becomes available as some active sessions are temporally silent. Since the exact moments of silent periods are unpredictable, conventional video coding cannot make use of the residual bit rate. DCVC, however, can send higher delay flows in the residual bit rate to progressively improve quality.

4. With the same quality as the single flow conventional video, multi-flow DCVC reduces its effective bandwidth to increase video capacity, defined as the number of concurrent sessions on a link.
5. For a time-varying link such as the wireless fading channel, the link rate adjusted for guaranteed reliability is also time-varying. This forces conventional video coding to take the common denominator of link rates at all fading conditions. However, DCVC can again take advantage of the time-varying rate since higher delay flows can be buffered for the most opportune moment for transmission. For example, as channel fading strikes, the transmitter can put off the transmission of higher delay flows and send the lowest delay flow with stronger error correction codes at a lower rate. Higher delay flows can be served later when the channel condition returns to normal.
6. The above arguments have emphasized bit rate savings and cost reductions, these savings can be redirected to decrease the latency of the low-delay flow thereby reducing perceptual delay. DCVC enables this additional dimension of trading off traffic capacity and perceptual delay.

From this description of DCVC, it is clear that we have made assumptions about the network transport. We assume there exists a differential delay multi-flow network transport service to support DCVC. This service provides the necessary functionalities to control, manage, monitor, and carry flows. A flow is an end-to-end connection opened by a networking application. An application can generate as many flows as needed while bits carried by the same flow receive identical quality-of-service in the network. The flows generated by DCVC are differentiated by end-to-end latencies. In this dissertation, we do not directly address the issues of designing and implementing the complete multi-flow architecture, which was described in Haskell, Messerschmitt and Yun [32]. Instead, the key concepts related to DCVC are summarized in the next section.

### 1.3 Multi-flow Transport Architecture

The differential delay flow transport architecture follows the spirit of *loosely coupled* joint source/channel coding (JSCC). JSCC represents a coordination between source and channel coding algorithms to maximize the traffic capacity of a link. The coordination implies that both source and channel are aware of each other's attributes, possibly in great detail. While the classic separation theorem of Shannon [19] states source and channel coding can be designed independently of each other without losing efficiency, the theorem does not apply to conditions requiring low delay, limited system complexity, nor multicast. JSCC can be roughly divided into two categories according to the extent of knowledge on each side. *Tightly coupled* JSCC provides full system attributes with fine details such as bitstream structure, signal constellations, and modulation techniques. On the other hand, *loosely coupled* JSCC wraps the details with high level characterization such as the rate and QoS parameters, which are pre-agreed upon by both source and channel.

Against the predominate trend in the literature of tightly coupled JSCC, loosely coupled JSCC is our favored architecture because the future networking environment will be more heterogeneous than today's network. A typical connection will transverse a number of links with orders of magnitude difference in QoS. This presents a problem to the conventional tightly coupled JSCC, which would require source coding adapt to all possible combinations of concatenated links. As shown in Figure 1.2, video coding can be tailored for a specific type of media; let it be Ethernet, asynchronous transfer mode (ATM) or code division multiple access (CDMA) wireless. Techniques such as error recovery, packetization, prioritizing, and retransmission may be designed to match the characteristics of the target medium. However, when two or more links with sufficiently different QoS constitute a connection, the tightly coupled JSCC fails.

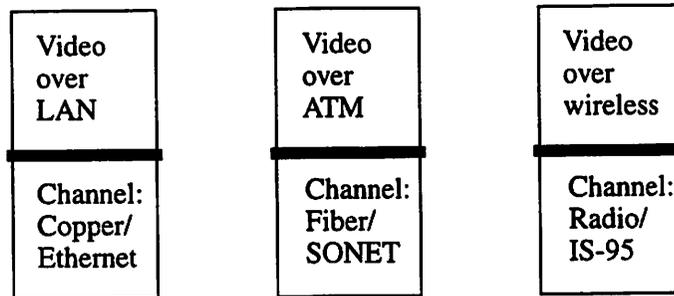


Figure 1.2 An example of tightly coupled joint source channel coding. Past research on video has designs tailored for specific media such as video over LAN, ATM, or wireless

Besides the issue of channel adaptability, we summarize some other desirable system attributes that the tightly coupled framework does not effectively address. [32][77] provide more detailed discussions.

- **Modularity:** tightly coupled JSCC does not allow system modules to be substituted easily without extensive redesign.
- **Scalability:** an upgrade on either source or channel side requires redesigning the other component.
- **Multicast:** one-to-many communications have potentially as many heterogeneous channels and receiver terminals. There is no longer a one-to-one mapping between source and channel coding, as required by tightly coupled JSCC.
- **Privacy:** tightly coupled JSCC prevents the provisioning of end-to-end security. Once encrypted, a source bitstream no longer has its structure exposed to the channel.

In [77], a “loosely coupled” JSCC framework was proposed with differential QoS flows by adding another layer of abstraction between source and channel coding. Its reference model is shown in Figure 1.3. Here is a brief description of the functionality of each layer in this model.

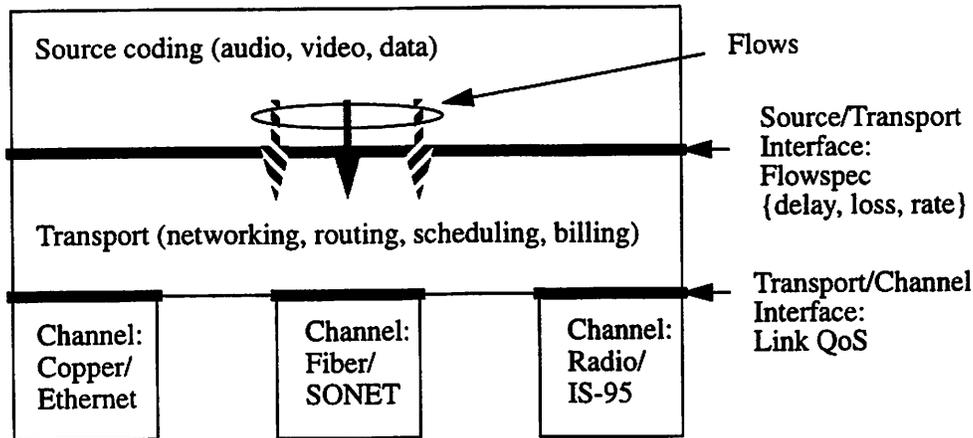


Figure 1.3 The three-layer architecture reference model for loosely coupled joint source channel coding

- Source coding: Compression and encryption of basic media types such as audio, video and data. A source coder generates a set of traffic correlated flows and assigns each flow a unique flow specification. A flowspec contains an ID, a traffic descriptor (such as leaky bucket parameters), and the delay and loss requirements of the flow. This layer corresponds to an augmented presentation layer in the open system interconnect (OSI) seven layer model [61].
- Source/transport interface: Service and cost negotiations for each flow. The flow data structure is maintained at this interface.
- Transport: Networking, routing, scheduling and billing of an end-to-end flow connection. This layer corresponds to the transport and networking layers in the OSI model.
- Transport/channel interface: Resource negotiation and impairment allocation of the end-to-end flowspec to individual channels.
- Channel coding: Control and transmission of a homogeneous physical channel. This layer corresponds to the link and physical layers in the OSI model.

The reference model represents a limited decoupling of JSCC to trade off scalability and modularity. QoS parameters shield the detailed networking implementations from upper-layer applications and yet preserve fine enough information to seize the benefits of JSCC. The communication system can still achieve much of the efficiency of tightly coupled JSCC by matching flow specs to optimized channel operations. For example, bits requiring high reliability are sent through reliable flows (possibly with forward error correction (FEC) in the channel) and others are carried by less reliable flows without FEC.

Coordination and negotiation at the two interfaces require a lot of message passing among network entities. In [45], intelligent agents (a form of mobile code) were proposed to speed up the process. In order to carry out the agreed QoS coordination, a subnet (channel) must also have the ability to reserve its own resource and keep track of the service received by each flow. Although today QoS guaranteed network connections are only experimental, the next generation Internet Protocol (IP) has incorporated a 28-bit flow label and priority field into its packet header structure to support future deployment [8]. (Also see Box 1.D on page 15 for the new IPv6 packet header format.) The Internet Resource ReSerVation Protocol (RSVP) [9] and similar efforts for ATM support forms of QoS guaranteed flows.

Even outside this QoS flow architecture, video coding research has proposed multi-flow coders, in the name of layered coding (targeted at rate scalability [54][66][67] and error resiliency [44][58][64]). Rate and error (loss/corruption) represent two of the three key QoS parameters. Just as rate scalability and error resiliency have been integrated, ultimately a fully QoS-adaptive video coding algorithm must incorporate all three parameters. Delay is an important contributor to network capacity and resource consumption. DCVC is a step in this direction.

## Box 1.D IPv6 Packet Header Format

Internet Protocol version 6 (IPv6) has several important differences from IPv4, which is the current version being used [8][63]. The most notable one is the increase in address size from 32 bits to 128 bits. And to facilitate high-speed switching, IPv6 packet header length is fixed at 80 bytes, as compared to the variable size of IPv4 header. Finally, and most importantly, IPv6 incorporated a 4-bit priority field and a 24-bit flow label to enable the provisioning of differential QoS. The packet header format is shown below.

Version (4 bits)	Priority (4 bits)	Flow label (24 bits)		
Payload length (16 bits)		Next header (8 bits)	Hop limit (8 bits)	
Source address (128 bits)				
Destination address (128 bits)				

### 1.4 Dissertation Overview

The contributions of this dissertation are organized as follows. Our research methodology is to first develop the core technology - delay cognizant video coding (at the service layer) - and then expand to address related issues in upper and lower layers. DCVC is a non-traditional type of coding, so we need to demonstrate its usefulness in applications which benefit from DCVC. Thus after the algorithm was developed, the research focus moved up to the applications layer. In many cases, the quality of network video applications is judged by viewers. The novel coding might introduce new visual artifacts and a poor quality would reduce its applicability. This concern was addressed by conducting subjective quality evaluation experiments with human participation. While the issues of applications and quality are significant, an equally important subject is to show how DCVC flows can be transmitted in the bitway layer. DCVC assumes that the multi-flow, differential QoS transport is available. Moreover, we believe wireless networks are likely to benefit most from differential delay flows. Code Division Multiple Access (CDMA) wireless is chosen to show a feasible network architecture and mechanisms to support differential QoS flows. The ordering of chapters is organized in the above methodological sequence: coding, applications, quality and wireless transport.

Chapter 2 focuses on the coding aspect and starts with the design objectives of DCVC, which are to minimize the amount of lowest delay traffic and to maximize the delay tolerance of higher delay traffic. The justification of having multiple flows with differential delay is qualitatively analyzed by using a rate-distortion formulation. With the design objectives in mind, we examine prior work in delay segmentation schemes. Delay segmentation is crucial both to the quality and compression performance of DCVC. Three segmentation approaches are reviewed and while each of them was designed to address the weakness of the previous approach, latter designs incurred new problems. Based on these experiences, a simple, effective segmentation method was developed for the current version of DCVC. It segments video on a block-by-block basis and the criterion of segmentation is based on the variation of spatio-temporal frequencies. The architectures and major building blocks of both DCVC encoder and decoder are then described in detail.

Chapter 3 focuses on the applications aspect. Because the mechanism of provisioning differential delay flows varies in different network environments, we divide the chapter into four sections, each of which defines and describes its network and the way DCVC may be applied. The four network environments discussed are broadband ISDN (B-ISDN) [37], delay/cost differentiated networks, the Internet, and wireless networks. We show that DCVC can be applied to B-ISDN to make use of residual rate for improving video quality and increasing capacity. It can be used to reduce connection costs in the emerging delay/cost differentiated networks. Internet video can benefit from DCVC to increase error resiliency while still maintaining low latency. DCVC can also be applied to time varying wireless links to make use of available rate to improve video quality.

Chapter 4 focuses on the quality aspect. In spite of its promise of significant traffic capacity gain, DCVC must ensure at the degradation in quality to be acceptable even with long delay. To verify that the performance of our coding algorithm is satisfactory, we conducted both psychophysical and computational

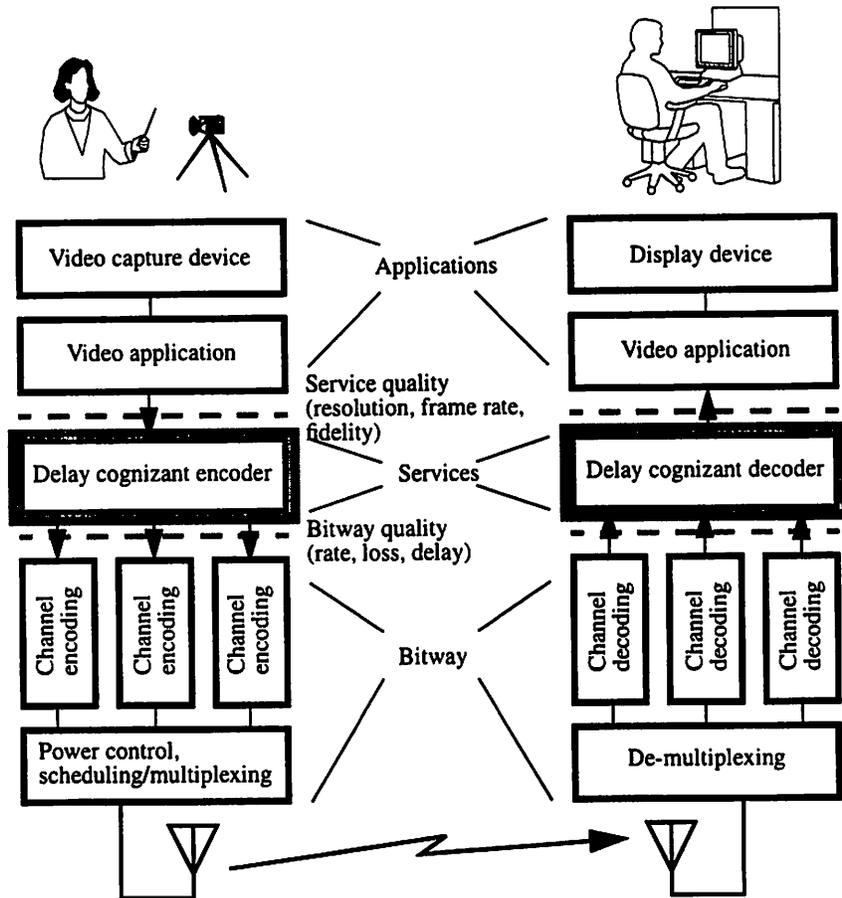
evaluations of DCVC video. Psychophysical studies rely on the participation of human subjects, who were shown video clips and were asked to judge their quality. We found that under heavy compression, sequences with long delay sometimes looked better than sequences without delay. We discuss possible causes of this surprising result and construct artificial stimulus to simulate those conditions.

Chapter 5 focuses on the wireless transport aspect. One premise of DCVC is a network infrastructure that supports differential delay flows, which hold promises to improve traffic efficiency. Since a wireless network, with scarce bandwidth and hostile channel environment, is likely to benefit most from the improvement, we chose wireless CDMA networks to study at the bitway layer. We examine issues of using power control and scheduling techniques in CDMA to control the QoS received by each flow. We first describe the scheduling problem of CDMA networks with fixed-rate links, which correspond to the second (the current) generation mobile cellular service. Scheduling and admission control problems in general are found to be NP-complete. We then discuss the scheduling and network capacity issues of multi-rate links, which are modeled after the third generation cellular, under various power and reliability constraints as well as two receiver structures. Multi-rate links are better suited for video transmissions for their flexible bit rate allocation. We found in most cases, by applying multi-user receivers, equal cell throughputs are achieved through spreading gain scalability or multiple code scalability.

Finally, Chapter 6 concludes the dissertation and points out future extensions to this research. The main challenges ahead are to integrate layered coding techniques of differential QoS flows and to make use of these flows.

# 2

## Delay Cognizant Coding Architecture



Our primary goal is to leverage the differential delay/cost network transport service to increase video traffic capacity, reduce video service cost, shorten perceptual delay, and adapt to time-varying wireless links. Recognizing that conventional video coding does not provide a solution to meet the above challenges, we pursued the direction of delay cognizant video coding (DCVC). The most distinguishing features of DCVC are differential delay flows and asynchronous video

rendering. While flows of existing layered video coding techniques are differentiated either by reliability or bit rate requirements, DCVC flows address the delay aspect. Video data in a flow has the same delay requirement and data in different flows experiences different latencies. Video reconstruction at the receiving end abolishes the traditional, synchronous video rendering model. Instead, received data is rendered immediately upon arrival. This new model eliminates the need for a de-jittering buffer at the receiver and thus removes another significant contributor to the already tight end-to-end delay budget.

In this chapter, we introduce the encoder and decoder architectures of a DCVC design. Although the aforementioned benefits are alluring, no prior research that we are aware of outside of this group has attempted to design such a coder and even our efforts followed a long journey to reach the current level of understanding. To make the objectives clear, we first list the design criteria and characterize the objectives in a rate-distortion formulation. We then discuss prior design approaches and the lessons learned. Finally, we present the encoding and decoding algorithms.

## **2.1 Objective**

DCVC has no hard objective measure of delay, since visual information is generally displayed without synchronization. The focus is thus on perceptual delay (see Box 1.C on page 8) - the delay perceived by end users. An objective measure does not exist because rendered video at the receiver display may be composed of data from multiple capturing instants (frames) in a time period of hundreds of milliseconds. There is thus no one-to-one mapping between a captured frame at the sender and a composite frame at the receiver. However, with the lowest delay flow carrying the most visually significant information, the perceptual delay is primarily determined by the delay in the lowest delay flow.

Although the number of differential delay flows is arbitrary, there is a tradeoff between the flow-incurred overhead and the benefits brought by multiple

## Box 2.A Effective Bit Rate

Effective bit rate, or often referred as effective bandwidth in networking research, was proposed to characterize the bit rate of a stochastic bursty traffic source such as compressed video. Assuming that traffic flows arriving at a network switch are independent and a small nonzero packet loss probability (typically  $10^{-5}$  or lower) due to switch buffer overflow is allowed, both theoretical and experimental analysis showed the effective bit rate can be less than the peak rate of the source. Due to the bursty nature of video, this traffic capacity gain can be significant. Effective bit rate is a function of this loss probability, the size of the switching buffer, and the stochastic properties of the traffic source. Always less than the peak rate of a VBR source, effective bit rate is still greater than or equal to the average rate.

flows. In the rest of this dissertation, we focus on two flows with finite delays, low- and high-delay flows, recognizing that this can be easily generalized. Should quantization loss be counted as another flow with infinite delay, there would be three flows. However, we follow the convention in the literature to exclude quantization loss as a flow.

A DCVC algorithm attempts to optimize the following cost function while satisfying the minimum quality constraint.

$$\min_{R_1, R_2, d} C(R_1, R_2, d) \quad (\text{Eq 2.1})$$

$$\text{subject to } Q(R_1, R_2, d) > q_0 \quad (\text{Eq 2.2})$$

$R_1$  and  $R_2$  are the effective bit rates (see Box 2.A on page 20 for explanation on effective rates) of the low-delay and high-delay flows, respectively.  $q_0$  is the minimal acceptable quality and  $d$  stands for the delay offset between the two flows, assuming the delay requirement of the low-delay flow is fixed. We assume the network cost function  $C$  increases with  $R_1$  and  $R_2$ , and decreases in  $d$ . Similarly, the quality function  $Q$  is an increasing function of  $R_1$  and  $R_2$ , and a decreasing function of  $d$ . Aside from their first order properties, based on rate-distortion theory,  $Q$  is concave in  $R_1$  and  $R_2$  when quality is defined as the negation of distortion.

The above function characteristics lead to a conclusion about the optimal bit rate allocations of  $R_1$  and  $R_2$ , as described in Section 2.6. It is not always advantageous to have both flows: under the condition that the two flows have positively correlated traffic and the marginal cost increases faster than the marginal quality, the optimum is to have just one flow. However, this condition is arguably rare. In the other cases, having the second (high-delay) flow minimizes the cost. Specifically, we found the optimal condition can be achieved as follows:

1. Minimize total compressed traffic, while maximizing the portion in the high-delay flow and minimizing the portion in the low-delay flow.
2. Maximize the allowable delay offset that can be attained with acceptable quality.

The first objective is to reduce traffic in the low-delay flow so that minimal network resources are reserved to support its tight delay and jitter requirements. Traffic in the high delay flow has relaxed delay bounds, which gives the transport layer the most flexibility in transmission prioritizing and scheduling. Note that an implicit assumption of this objective is that the bit rate requirement of the low-delay flow should be less than the bit rate of the conventional, single flow video with the same quality.

The second objective is to ensure the delay relaxation is sufficient for traffic smoothing purposes. We added the constraint on acceptable visible artifacts because we expect artifacts to occur with asynchronous reconstruction. It is worth pointing out that DCVC is not another form of compression. A good compression algorithm should have removed all *invisible* artifacts subject to HVS properties and therefore, delaying the rendition of any additional information any further is going to generate visible artifacts. These artifacts become more noticeable as the delay offset increases. Unfortunately, prior HVS research has revealed little about the kinds of video information that has the most impact when delayed. It is an important issue in need of more research.

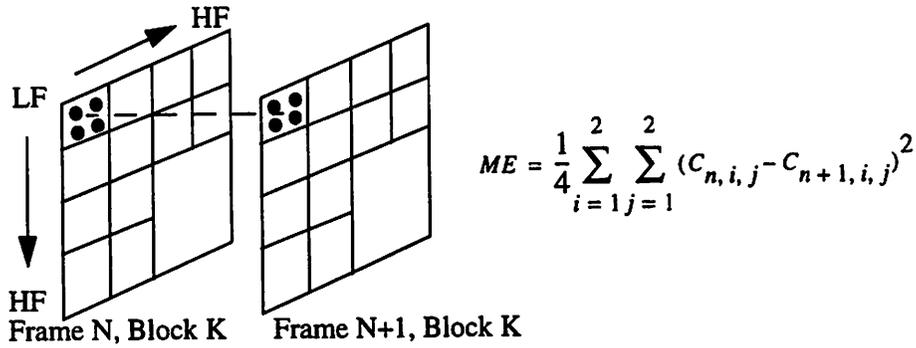


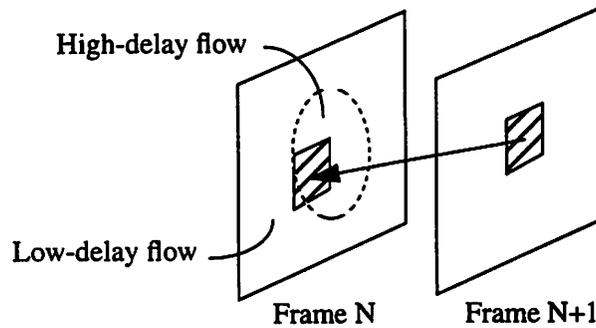
Figure 2.1 An illustration of obtaining the motion estimate of a block by taking the mean-square difference of the low-frequency subband.

## 2.2 Prior Work on Delay Segmentation

The exploration of the *right* segmentation scheme has been taking a path in developing heuristics rather than rigorous mathematical formulations. This methodology is the result of insufficient understanding of HVS properties to mathematically characterize video quality, especially in the case of asynchronous rendering. Our remedy for the situation, also taken by video standard-issuing committees, is to conduct subjective quality evaluation, whose results are reported in Chapter 4. In the following, we report three prior approaches taken in our group and explain their shortfalls.

### 2.2.1 Mean-square Difference of Low-frequency Subband

The earliest work reported in [59][60] segmented video into blocks of size 8 by 8 and performed block-based segmentation. As shown in Figure 2.1, a two-level orthogonal spatial subband decomposition is performed on the block to obtain the lowest frequency subband, which has its four coefficients marked by black dots in the figure. The block is then compared with its predecessor in time to obtain the motion estimate, which is defined as the mean-square difference of the four subband coefficients. A block is considered visually significant if its motion estimate exceeded some preset thresholds. The visually significant block is then transmitted through the low-delay flow, possibly in lower resolution. If the motion



**Figure 2.2** An illustration of using block moving distance to determine the flow to carry; if a similar block is found inside the circle, the block is sent to the high-delay flow. If not, the block is sent to the low-delay flow.

estimate is below the thresholds, the block is sent to higher delay flows in higher resolution. For a detailed discussion, see reference [59].

One shortfall of this approach pointed out in [59] is that blocks judged to have low motion in the pixel domain may result in large differences in the subband domain. For example, when a high textured region slowly moves across a block, its visual significance may be low but the movement results in high values of motion estimate. Another problem of this approach is that it only takes into account the low-frequency subband, which represents only one-sixteenth of the total 64 subband coefficients. Movements of sharp lines, which are often visually significant, can not be accurately reflected with just low frequency components.

## 2.2.2 Block Moving Distance

To remedy the shortfall of the aforementioned approach, a different method of delay segmentation was developed and reported in [12]. This approach is also block-based but the estimate of visual significance is made on block moving distance. As illustrated in Figure 2.2, a search for the best match is performed on the block. If a similar one is found within a short range, the block is considered low motion and less significant. If none is located in the search range, the block represents new information and is considered visually significant. In the former case, it is sent to the high-delay flow and in the latter, it is sent to the low-delay flow. Since it is rare that an exact same block is found, a fuzzy control based algo-

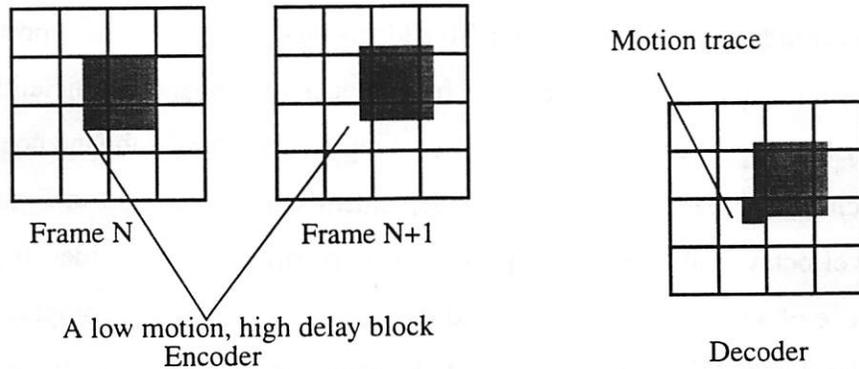


Figure 2.3 An example to show that uncovered background may result in motion trace at the decoder.

algorithm was developed to fuzzify the similarity measure to a continuous level. The algorithm takes into account both similarity and block moving distance in determining flow assignments.

Although this method deals with textured regions more effectively, motion traces are observed under aggressive segmentation. Motion traces occur when a block containing uncovered background is sent to the high-delay flow. A good example is a slow moving, uniform region. As shown in Figure 2.3, the block at the lower left corner of the gray rectangle in the frame  $N+1$  has a short moving distance and therefore is sent to the high-delay flow. To make the case easy to describe, we assume all other blocks go to the low-delay flow. At the decoder, the whole frame is updated through low-delay except the lower left corner, as shown in the figure. The gray rectangle thus moves with a trace behind. In the case of uniform areas, the motion trace can create significant visual artifacts. It is not true for high-textured regions, however. We found similar motion traces are much less noticeable for textured objects.

### 2.2.3 Spatio-temporal Subband Coefficient Variation

Both of the previous two methods relied on similarity measures of blocks, either by the mean-square difference of subband coefficients or by the absolute sum of pixel value differences. The main difficulty in measuring similarity by aggregated differences is to determine segmentation thresholds. If a low-delay

block is incorrectly classified to the high-delay flow, blocking artifacts due to asynchronous rendering really stand out. To address this problem, in this approach, the size of segmentation units is reduced from blocks to subband coefficients. In this unpublished approach, we tried conditional replenishment of subband coefficients, in which each coefficient is treated independently. As illustrated in Figure 2.4, a two-level octave subband decomposition is performed on each video frame. The difference of a subband coefficient and its predecessor in time is compared against a threshold, which is a function of spatial frequencies. In the figure, three different levels of thresholds are shown. Since each coefficient variation is tested only on a single threshold, there no longer exists the complexity of determining thresholds for block aggregated differences. We found in segmenting head-and-shoulder scenes, less than 5% of the total coefficients are carried by the low-delay flow. The delay offset can be as great as 330 msec without incurring noticeable quality degradation in our experiments when they were shown with the original video.

The shortfall of this approach, however, does not come from segmentation but from the step after, which is compression. Since the addresses of coefficient locations needs to be communicated to the receiver, the segmentation map needs to be compressed in addition to the coefficient values. We found the compression of the segmentation map of 5% coefficients turns out to be rather inefficient because its distribution has irregular shapes and spread, as the example in Figure 2.5 demonstrates. The additional overhead of communicating the locations of low-delay

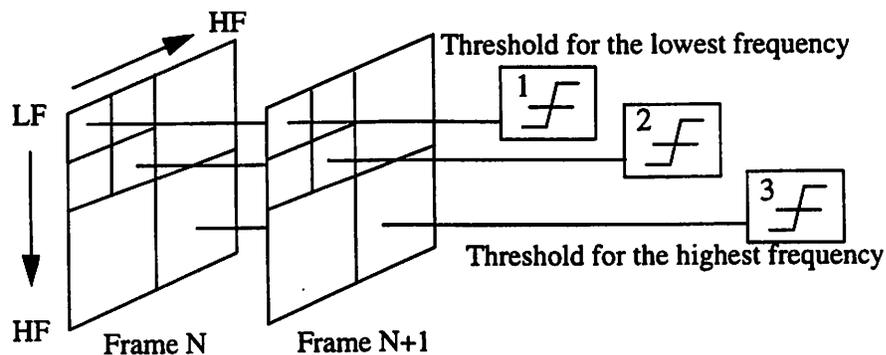


Figure 2.4 An illustration of estimating spatio-temporal subband coefficient variation. Thresholds are a function of spatial frequencies.

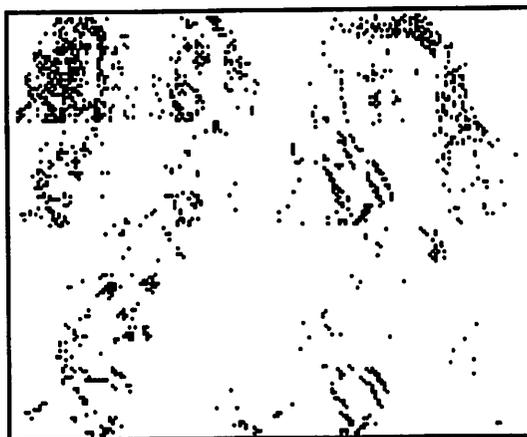


Figure 2.5 An example of the segmentation map with the locations of subband coefficients marked by black dots.

subband coefficients significantly increases the amount of low-delay traffic. Adding the overhead often leads to higher bit rates than conventional algorithms and makes it impractical for DCVC. To reduce this overhead, the segmentation granularity has been re-enlarged from pixels to blocks as described next in the current scheme.

## 2.3 Encoding Algorithm

The DCVC encoder shown in Figure 2.6 is divided into two stages: segmentation and compression, each of which is framed in the figure. A video frame is first processed by the segmentation stage to extract the low-delay information. High-delay data is obtained by subtracting low-delay data from the video frame. Extracted flow data is then passed to the compression stage to remove spatial and temporal redundancy. We describe the algorithm in the order of the processing: segmentation first and then compression.

### 2.3.1 Segmentation Stage: Spatio-temporal Block Variation

The current design applies a block-based segmentation amid its low addressing overhead. A captured video frame is first divided into blocks of size 8 by 8. Each block is then independently assigned to either the low- or high-delay

flow, based on the spatio-temporal frequency variation in the block. The flow diagram of this stage is marked and shown in the left dotted box in Figure 2.6.

Like the first two methods described in prior work, this segmentation algorithm also identifies visually significant blocks by measuring the similarity of a block and its predecessor in time. Experiences learned from prior shortfalls in block similarity measures indicated difference aggregations should be avoided. Instead, each frequency coefficient is compared against its corresponding threshold. There are a total of 128 test conditions, all of which must be satisfied for a block to be assigned to the high-delay flow. The 128 conditions are composed of 2 conditions each for every Discrete Cosine Transform (DCT) coefficient of the tested block, which has 64 coefficients. Since each coefficient is independently tested, it suffices to look at just one pair of such conditions:

$$\text{Condition 1: } |P_{i,j,n,t} - P_{i,j,n,t-1}| < V_{i,j}$$

$$\text{Condition 2: } |P_{i,j,n,t} - P_{i,j,n,update}| < V_{i,j}$$

In the above expressions,  $P_{i,j,n,t}$  is the  $(i, j)$ th DCT coefficient for block  $n$  at time  $t$ ;  $P_{i,j,n,update}$  is the  $(i, j)$ th coefficient of block  $n$  stored in a buffer for the latest update;  $V_{i,j}$  is a fixed preset threshold for the  $(i, j)$ th coefficient. The 8x8 table of  $\{V_{i,j}\}$  used in all the reported experiments is listed in Table 2.1 (for 8 bit pixels).

Table 2.1 The 8x8  $\{V_{ij}\}$  table of DCT coefficient thresholds; DC value is at the upper-left corner.

30	15	15	15	15	15	30	30
15	15	15	15	15	15	30	30
15	15	15	15	30	30	30	30
15	15	15	30	30	30	30	45
15	15	15	30	30	30	45	45
15	15	30	30	30	45	45	45
15	30	30	30	45	45	45	45
30	30	45	45	45	45	45	45

The first condition is to limit the variation of spatial frequencies in two consecutive frames. The subtraction operation can be viewed as a 2-tap high-pass Haar filter operating in the temporal dimension. The second condition is to limit the variation relative to the latest update that is the last block assigned to the low-delay flow. The two threshold blocks in Figure 2.6 are marked as Condition 1 (C1) and Condition 2 (C2).

The temporal variation of a block consists of steep changes as well as small perturbations as shown in Figure 2.7. Steep changes are typically originated from movements of objects with sharp contrast while small perturbations may come from slow variations of textures. To minimize visible artifacts, steep changes cannot be ignored and DCVC must act immediately by updating the block (region) with the low-delay flow. What DCVC can take advantage of are the small perturbations, which can be delayed without causing strong artifacts. The first condition (C1) is to monitor steep changes of a coefficient in consecutive frames. This condition only, however, is insufficient, as we found in experiments. In some scenes, the changes are more mild and gradual, which may not violate C1 that is set to detect steep changes. These changes, when accumulated, become more significant to the extent that they cannot be delayed. The second condition (C2) is designed to capture these gradual changes (or drifts). It limits the size of perturbations with respect to the latest update in time (block assigned to the low-delay flow). As a simple example to see their effects, a threshold of 15 is assigned to the following series of numbers.

40 40 40 40 40 40 40 50 60 60 76 75...

The numbers that trigger the thresholds are underlined. The first triggering is contributed by C2 while the second one is contributed by C1.

Although it is a block-based segmentation, this method is fairly similar to the spatio-temporal subband coefficient variation method described in

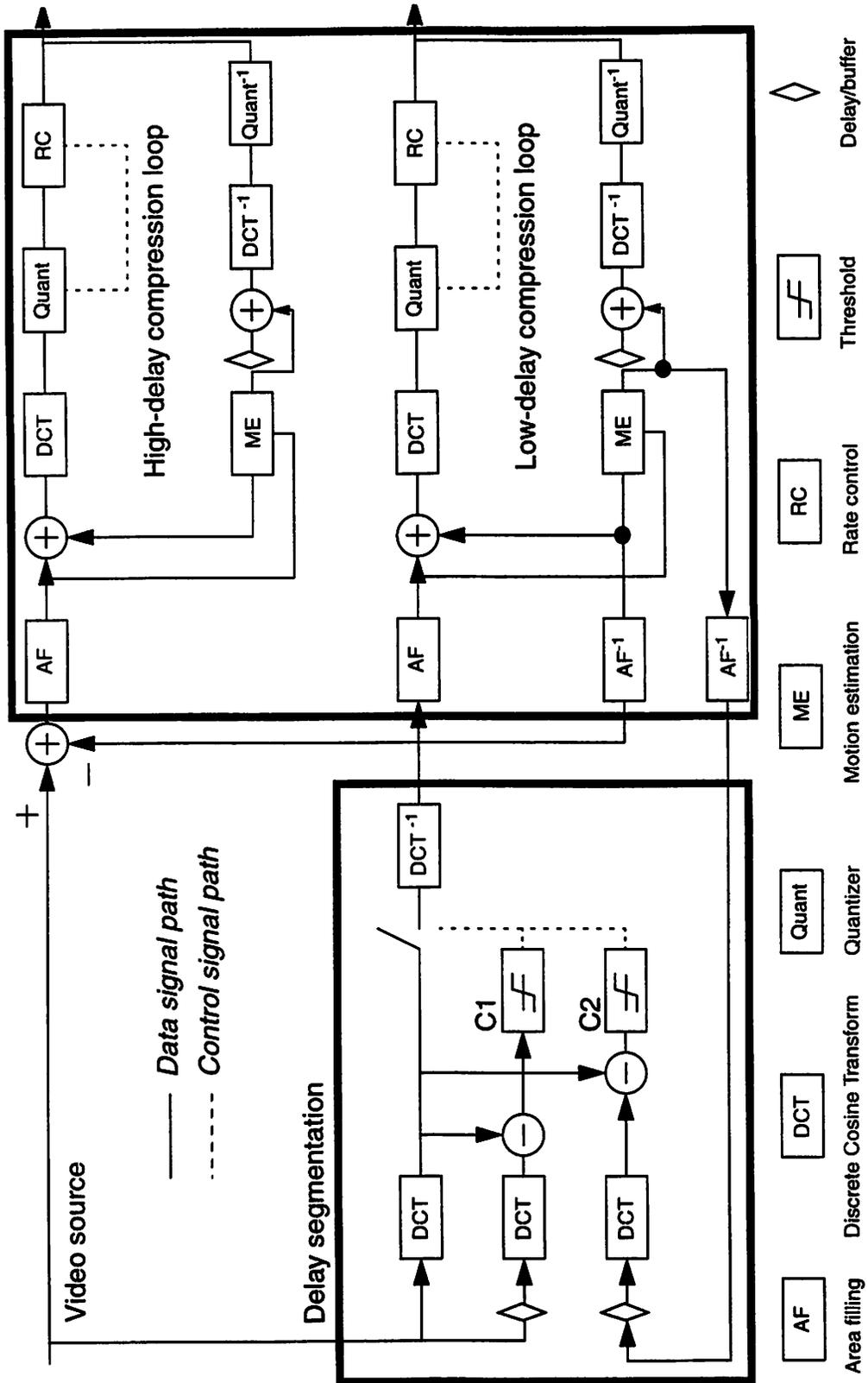


Figure 2.6 Delay cognizant video encoder block diagram

Section 2.2.3. In both methods, frequency components are tested individually. The

main difference is that in the current approach, the whole block is sent to the low-delay flow if any of its frequency components has a significant change. While it seems unnecessary to send the components with small or no changes, the benefit brought by reducing addressing overhead outweighs the bit rate increase of transmitting additional components.

The choice of  $\{V_{i,j}\}$  directly affects the performance of DCVC. More blocks will be assigned to the high-delay flow, which is our objective, by simply increasing  $\{V_{i,j}\}$ . However, this would cause significant visible artifacts especially for low spatial frequencies. Through informal viewing experiments, we noticed higher spatial frequency components could tolerate a greater temporal variation without adversely affecting quality. Our observation seems to be consistent with human vision studies on the roll-off of Contrast Sensitivity Function (CSF) at high spatial frequencies [80]. A rigorous mathematical formulation to optimize the thresholds  $V$ , however, is yet to be developed due to the lack of vision modeling in quantifying video quality. The parameters listed in Table 2.1 are experimental and we leave a more rigorous HVS-based characterization to future work.

As indicated in the block diagram of the segmentation stage in Figure 2.6, two threshold units control the switch to selectively update blocks through the low-delay flow. The low-delay blocks are then inverse transformed back to the spatial (pixel) domain and are put into the low-delay image plane for compression. An

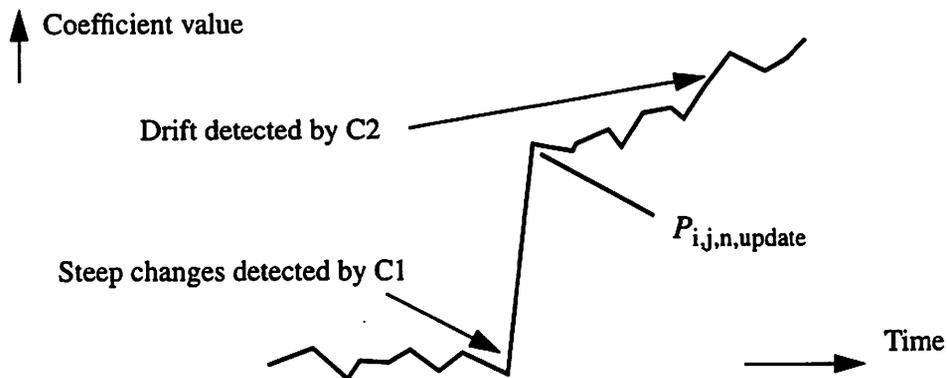


Figure 2.7 An illustration of the two detection conditions; the curve of a time-varying coefficient value is shown.

example is shown in Figure 2.8, where three frames of the Suzie test sequence are shown with low- and high-delay images. Areas in black have no pixel values. The low- and high-delay images can be added to obtain the original. It is not surprising to see most of the background resides in the high-delay portion. It is also worth noting that although the person in picture is moving, some of the high-textured regions, such as hair, are segmented to the high-delay flow as well.

For head-and-shoulder scenes appeared in videoconferencing applications, the low-delay image plane typically consists of 10 to 20 percent of the total blocks. The percentage is observed to vary significantly depending on the motion content of the sequence. In the case of scene changes, the whole frame is sent to the low-delay flow. The corresponding high-delay image plane consists of blocks going to the high-delay flow. The generation of the high-delay plane is related to the compression stage, which will be described next.

### **2.3.2 Compression Stage: Motion Estimation with DCT**

The compression stage as shown at the right frame of Figure 2.6, removes spatial and temporal redundancies from the segmented video flows. Since redundancy removal creates data dependency, compressing the low-delay flow with reference to data in the high-delay flow is not allowed. Should the data dependency constraint be violated, low-delay data cannot be decompressed ahead of high-delay data, leading to the synchronous reconstruction. The compression of the high-delay flow, however, can reference data in the low-delay flow. The current encoder is chosen to keep the compression of two flows independent because we did not

found significant compression gain to justify the additional complexity of establishing cross-flow references.

The compression stage complies with the dependency constraint by using two separate encoding loops. Our design adopts the motion estimation (ME) with discrete cosine transform (DCT) architecture [18][35][50]. The upper (lower) ME loop in the diagram corresponds to the compression of the high- (low-) delay flow. The reconstructed low-delay image plane in the ME loop is fed back to the segmentation stage. The feedback is stored as the latest update for the second segmentation condition described previously. The high-delay image plane is obtained by subtracting the reconstructed low-delay image plane from the input video frame.



Figure 2.8 From top to down are the 45th, 46th, and 47th frames of Suzie sequence; From left to right are the original, low-delay, and high-delay images.

This allows quantization errors in the low-delay ME loop to be passed as a part of the input to the high-delay ME loop.

In the following, the key aspects of motion estimation, transform coding and quantization are briefly reviewed. These techniques are well known in video coding research. We then describe area filling and rate control functions, which were developed for DCVC. Area filling was used to improve the efficiency of motion estimation. Rate control was used to control the coded quality of compressed flows.

### **2.3.3 Motion Estimation, Transform Coding and Quantization**

Motion-compensated video coding forms the basic architecture of all existing and some in-developing video compression standards, which include MPEG1, MPEG2 [50], MPEG4, H.261, and H.263 [2]. Rather than treating video as a three-dimensional signal, it is often found more effective to describe scenes in successive video frames with simple transformations such as translation, zooms and pans. While the first video frame is compressed using image coding techniques, its following frames can often be deduced by indicating how image regions move. The presence of motion structures suggests ways to achieve high compression by using the motion estimation and compensation.

To apply motion estimation, the compression algorithm first divides a frame into  $N$  by  $N$  blocks (typically  $N = 8$  or  $16$ ). It then performs a search for the best match from the previous reconstructed frame. The difference block (prediction error) between the current block and its best match is compressed using transform coding. Over the years, numerous work has been done to speed up the motion search and increase matching precision. For example, to speed up searching, a hierarchical three-step matching was proposed to find the motion gradient first and then narrow the search range. To increase matching precision, a half-pixel interpolation is often performed on the reconstructed frame first. Motion characterization then extends the precision from pixel to half pixel. Depending on targeted applica-

tions, the aforementioned video compression standards define different modes of motion estimation, such as forward, backward, and bi-directional prediction. Different code tables are applied to compress the motion vector, which records the relative location of the best matched block. The decoder uses the motion vector to find the same prediction and add it to the decoded prediction error. In a way, such a scheme can be viewed as a two-dimensional DPCM (differential pulse coded modulation).

Transform coding of the difference block often applies discrete cosine transform (DCT) for its near optimality in compacting energy. The DCT coefficients are then quantized using a uniform quantizer with a zero zone to minimize the number of nonzero coefficients. Quantized coefficients are then encoded by run level encoding, which records for each nonzero coefficient, the number of zeros before it and its quantized value.

In our implementation, we applied the coding tables of the H.263 standard for motion vectors and run level coding. This is because videoconferencing is one of our target applications and some of the test sequences we acquired were H.263 test clips. With the same quantization step sizes applied to both flows, the compression efficiency is listed in Table 2. Four test sequences of head-and-shoulder scenes were encoded, all of which except the Carphone sequence have still backgrounds. The low-delay flow is approximately 50 to 80 percent of the total compressed traffic although it carries less than 20 percent of the blocks. This is mainly because the compression of the high delay flow is far more effective. We expect further optimization in compression to reduce the total bit rate as well as the low-delay portion.

Table 2.2 Average bits per pixel for the low- and high-delay flows; raw video is captured in 12-bit resolution or YUV4:2:0 format. As a comparison, H.263 outputs are listed at the last column.

Video sequence	Low-delay flow	High-delay flow	H. 263
Suzie	0.063	0.032	0.073
Salesman	0.035	0.018	0.043

Table 2.2 Average bits per pixel for the low- and high-delay flows; raw video is captured in 12-bit resolution or YUV4:2:0 format. As a comparison, H.263 outputs are listed at the last column.

Carphone	0.132	0.031	0.153
Miss America	0.028	0.024	0.034

### 2.3.4 Area Filling

A direct compression on the image planes as shown in Figure 2.8 does not turn in competitive compression efficiency because artificial block boundaries create many high frequency residues in prediction errors. An effective solution, as we found, is to fill the empty regions (indicated by black areas) with the same blocks from the reference frame in the ME loop. Pixel values are copied from the reference frame to the no-value regions. Since the first frame is always intra-coded, area filling guarantees all successive frames have no empty regions. Area filling operations are performed ahead of compression, as indicated by the AF blocks in Figure 2.8.

Figure 2.9 illustrates area filling using a simple frame with only nine blocks. Suppose the reconstructed first frame contains blocks from 'A' to 'I'. The reconstructed frame is used as the reference frame in the ME loop. The image at the center of the first row contains new blocks from 'J' to 'N' and four gray blocks. Area filling copies the blocks 'A', 'D', 'H' and 'I' of the reference frame to the gray blocks at corresponding locations. The resultant frame after area filling is shown on the right. To encode the third frame, the reconstructed second frame is used as the reference. In the example shown, blocks 'A', 'J' and 'K' are copied to replace the gray blocks of the image at the second row.

Area filling improves compression efficiency by preserving the shapes of image objects and smoothing artificial block boundaries. Blocks used in filling the empty area are not compressed for they are copied from the reference. A one-bit indicator per block is used to inform the decoder if the coded block belongs to the low-delay flow. At the decoder, it has the same video history as the encoder does and thus area filling can be performed without any additional overhead. For the

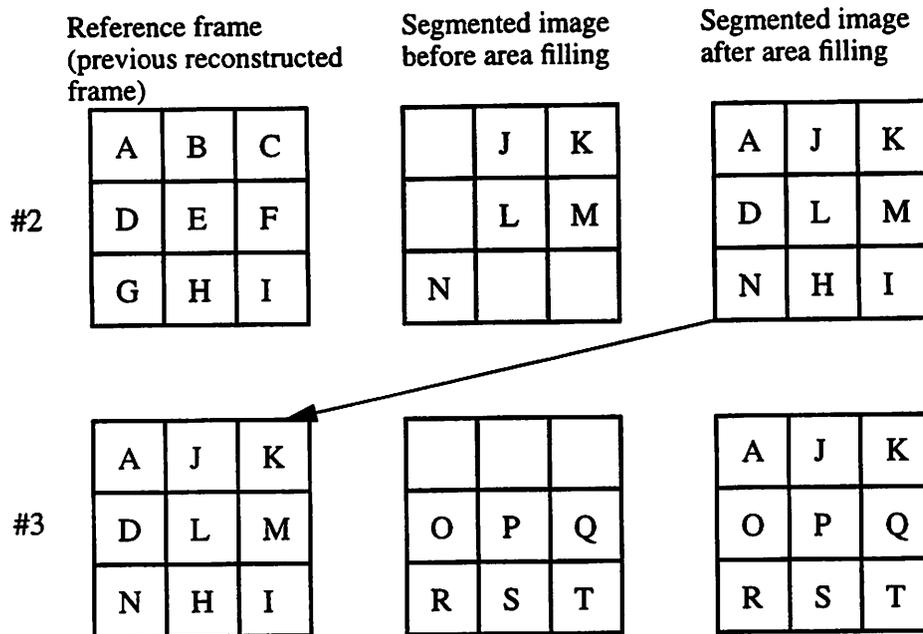


Figure 2.9 An illustration of the area filling operation; blocks are denoted by letters and the gray blocks have no pixel values.

compression of the high-delay flow, the one-bit indicator per block is not required because all blocks have high-delay components, some of which are quantization errors from the low-delay ME loop and some of which are normal video blocks. The addressing overhead before entropy coding is one bit for every block of 64 pixels, or 0.016 bpp.

### 2.3.5 Rate Control

At the outputs of both ME loops, rate control modules monitor output rates to ensure compliance with the QoS contract. As demonstrated in the chapter discussing DCVC applications, the quantization steps of the two ME loops need not be the same and this flexibility can be exploited to further improve video quality. The rate control module in the low-delay ME loop performs adjustments of quantization steps in the conventional way. A more interesting opportunity is to make use of the module in the high-delay ME loop to control the number of high-delay blocks in a frame. With a fixed bit rate, decreasing the number of blocks to be compressed increases their coded quality. For the high-delay flow, blocks are selected

based on their past history with the assumption of *delay inertia*. We presume those blocks that are less frequently updated through the low-delay flow tend to stay that way. These blocks stay on the receivers screen for a long time and thus need a higher visual quality. The rate control module keeps an age record of blocks and prioritizes the selection in the descendent order of block ages. The age of a block, incremented at each frame, is reset to zero when it is either updated as a low-delay block or selected for the high-delay flow. Applying rate control increases addressing overhead for the high-delay flow because the flow must also carry its own segmentation map. As Table 2.2 indicates the high-delay flow only contributes a small portion of the total traffic, the additional overhead is not very significant.

The rate controlled encoding of blocks can be viewed as the video extension of progressive image transmission. A block is updated through the low-delay flow to establish a coarse initial representation. It is then replenished through the high-delay flow with a finer version. Unlike image coding, video does not allow progressive coding for the whole frame. DCVC segments the video frame to regions with different updating frequencies. For high texture, slow varying regions, the rate controlled compression works well.

## 2.4 Decoding Algorithm

The DCVC decoder block diagram shown in Figure 2.10 is divided into two stages: the independent decompression of flows and their video composition. We do not elaborate further on the decompression stage, since it differs little from standards and textbook examples. Instead we focus on describing how the two video flows are combined and composite for final rendering.

The DCVC decoder follows a simple set of rules to display received blocks. Compressed bit streams from both flows are tagged with temporal references (frame numbers). The decoder maintains one temporal reference table for each flow, in which each entry stores the frame number of the received block at the coordinates. The tables are initiated to zero and blocks from earlier frames are

## Box 2.B Block Annihilation

It is conceivable that further traffic capacity gain may be obtained by instructing the network to stop forwarding those blocks in the high-delay flow, which are now obsolete. The annihilation of the blocks is, however, not possible in our current implementation using differential coding. If they are dropped in the network, the motion compensation loop of the high-delay flow at the decoder will lose synchronization with the loop at the encoder. Annihilation can be made possible by removing the dependency and using non-differential coding for compression. Special packetization and application-aware network switches must be deployed to take advantage of block preemption.

replaced with those from later frames. By comparing  $TR_{n,L}$ , temporal reference of the  $n$ th block from the low-delay flow, and  $TR_{n,H}$ , temporal reference of the  $n$ th block from the high-delay flow, the decoder makes the following decision:

1.  $TR_{n,L} > TR_{n,H}$ , display the block from the low-delay flow;
2.  $TR_{n,L} = TR_{n,H}$ , display the sum of two blocks;
3.  $TR_{n,L} < TR_{n,H}$ , display the block from the high-delay flow.

As an example, Figure 2.11 illustrates temporal reference tables and their composite frame. With the exception of identical frame numbers, the displayed block always comes from the latest frame, following Rule 1 and 3. It is easy to see the compositely video on the right is obtained by taking the maximal of each entry. These rules lead to asynchronous display of blocks. We assume when a block has its coarse representation in the low-delay flow and added details in the high-delay flow, the delay experienced by the low-delay traffic is always equal to or less than that of the high-delay traffic.

Due to a nonzero delay offset between the two flows, it is possible that a block from a later frame may be transmitted through the low-delay flow but it arrives at the decoder earlier than its precedents in the high-delay flow. By the decoding rules, this block preempts its precedents and it is rendered upon arrival. The occurrence of preemption is due to significant changes of some spatial fre-

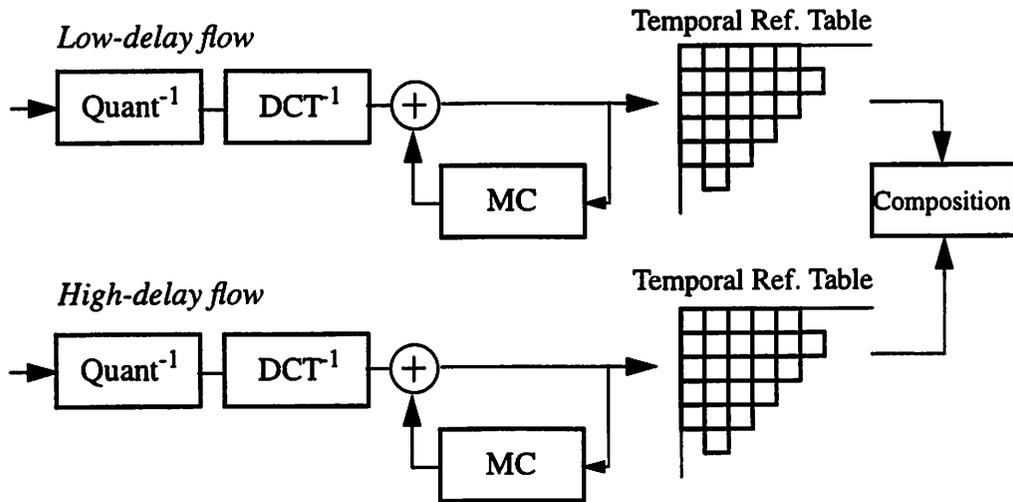


Figure 2.10 Schematic block diagram of the DCVC decoder.

quency components, as the segmentation is designed to detect. Movements of objects and scene changes typically cause those changes. Block preemptions can, under certain conditions, be used to further improve the efficiency of the network by annihilating obsolete blocks. Please refer to Box 2.B on page 38 for further discussion.

## 2.5 Summary

In this chapter, we first stated the goals of delay cognizant video coding, which are minimizing the bit rate of the lowest delay flow and maximizing delay tolerance of higher delay flows. The success of DCVC very much depends on delay segmentation to extract the most visually significant information for preferential delay treatment. We then described several prior segmentation methods and pointed out their shortfalls. Learning from those experiences, we developed the current scheme of estimating spatio-temporal block variation. The current method provides a good tradeoff between segmentation granularity and addressing overhead of segmented block locations. The compression of segmented video flows are performed by motion estimation and DCT to remove spatial and temporal redundancies. Techniques like area filling and rate control are developed to allow more efficient compression and better quality control. On the decoder side, a set of rules

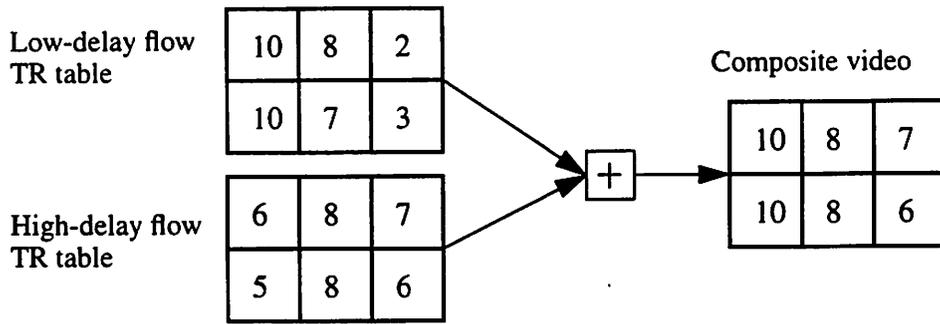


Figure 2.11 An illustration of video composition at the decoder.

are developed to determine the composition of frames for asynchronous video rendering. Conceivably, a post-processing stage can be added after decoding to reduce visual artifacts arising in asynchronous rendering. But we leave it to future work.

DCVC raises many possibilities in developing new applications which use differential delay flows to improve video quality and increase capacity. These new applications cannot be achieved with conventional video coding. In the next chapter, we move up one layer in the chapter front diagram to address DCVC applications.

## 2.6 Appendix A: Optimal Flow Rate Assignments

In this appendix, we explore basic properties of the cost and quality functions to gain more insight into the optimal condition of bit rate allocations of delay flows. First, we restate the optimization criterion and its constraint. A DCVC algorithm attempts to optimize the following cost function while satisfying the minimal quality constraint.

$$\min_{R_1, R_2, d} C(R_1, R_2, d) \quad (\text{Eq 2.3})$$

$$\text{subject to } Q(R_1, R_2, d) > q_0 \quad (\text{Eq 2.4})$$

$R_1$  and  $R_2$  are the average bit rates of the low-delay and high-delay flows, respectively.  $d$  is the delay offset between the two flows.  $q_0$  is the minimal acceptable

quality. As stated in (Eq 2.5), the perceptual quality function  $Q$  marginally increases with  $R_1$  and  $R_2$ , and marginally decreases with  $d$ . It is known that the rate-distortion curve is a convex function [7]. If the quality is characterized by the negation of distortion, say  $Q = \text{constant} - D$ , in the range of normal video quality, we can conclude the quality function  $Q$  is concave as in (Eq 2.6). The concavity of  $Q$  means the marginal quality improvement decreases as the rate increases.

$$\frac{\partial Q}{\partial R_1} \geq 0, \frac{\partial Q}{\partial R_2} \geq 0, \frac{\partial Q}{\partial d} \leq 0 \quad (\text{Eq 2.5})$$

$$\frac{\partial^2 Q}{\partial R_1^2} \leq 0, \frac{\partial^2 Q}{\partial R_2^2} \leq 0, \frac{\partial^2 Q}{\partial R_1 \partial R_2} \leq 0 \quad (\text{Eq 2.6})$$

**Theorem 2.1** For given  $d = d_0$ ,  $Q$  is a concave function of  $R_1$  and  $R_2$ . That is,

$$\frac{Q(R_1, R_2) + Q(R_1 + \Delta R_1, R_2 + \Delta R_2)}{2} \leq Q\left(R_1 + \frac{\Delta R_1}{2}, R_2 + \frac{\Delta R_2}{2}\right) \quad (\text{Eq 2.7})$$

Proof of Theorem 2.1:

We first describe a sketch of the proof. Consider the case when  $\Delta R_1$  and  $\Delta R_2$  are small. Take the two-variable Taylor series expansion on both sides of (Eq 2.7). When the second order derivatives are non-positive, the lefthand side of (Eq 2.7) is smaller than its righthand side. For small  $\Delta R_1$  and  $\Delta R_2$ , one can ignore the contribution of higher order terms. Thus (Eq 2.7) holds. Once we established the inequality holds for two neighboring points, the proof can be generalized to the case of two distant points.

Assume  $\Delta R_1$  and  $\Delta R_2$  are small. Left side of (Eq 2.7) after the Taylor expansion at  $(R_1, R_2)$ :

$$\frac{Q(R_1, R_2) + Q(R_1 + \Delta R_1, R_2 + \Delta R_2)}{2} = Q(R_1, R_2) + \frac{1}{2} \cdot \frac{\partial Q}{\partial R_1} \cdot \Delta R_1 + \dots$$

$$\begin{aligned}
& \frac{1}{2} \cdot \frac{\partial Q}{\partial R_2} \cdot \Delta R_2 + \frac{1}{2} \cdot \frac{\partial^2 Q}{\partial R_1 \partial R_2} \cdot \Delta R_1 \cdot \Delta R_2 + \frac{1}{4} \cdot \frac{\partial^2 Q}{\partial R_1^2} \cdot \Delta R_1^2 + \frac{1}{4} \cdot \frac{\partial^2 Q}{\partial R_2^2} \cdot \Delta R_2^2 + \dots \\
& + o(\Delta R_1^3) + o(\Delta R_2^3) + o(\Delta R_1^2 \Delta R_2) + o(\Delta R_1 \Delta R_2^2) \quad (\text{Eq 2.8})
\end{aligned}$$

Right side of (Eq 2.7) after the Taylor expansion at  $(R_1, R_2)$ :

$$\begin{aligned}
& Q\left(R_1 + \frac{\Delta R_1}{2}, R_2 + \frac{\Delta R_2}{2}\right) = Q(R_1, R_2) + \frac{1}{2} \cdot \frac{\partial Q}{\partial R_1} \cdot \Delta R_1 + \dots \\
& \frac{1}{2} \cdot \frac{\partial Q}{\partial R_2} \cdot \Delta R_2 + \frac{1}{4} \cdot \frac{\partial^2 Q}{\partial R_1 \partial R_2} \cdot \Delta R_1 \cdot \Delta R_2 + \frac{1}{8} \cdot \frac{\partial^2 Q}{\partial R_1^2} \cdot \Delta R_1^2 + \frac{1}{8} \cdot \frac{\partial^2 Q}{\partial R_2^2} \cdot \Delta R_2^2 + \dots \\
& + o(\Delta R_1^3) + o(\Delta R_2^3) + o(\Delta R_1^2 \Delta R_2) + o(\Delta R_1 \Delta R_2^2) \quad (\text{Eq 2.9})
\end{aligned}$$

With small  $\Delta R_1$  and  $\Delta R_2$ , the contribution of higher order terms can be ignored. Since all the second order derivatives are non-positive as stated in (Eq 2.6), comparing (Eq 2.8) and (Eq 2.9) concludes (Eq 2.7) holds.

Next consider that  $\Delta R_1$  and  $\Delta R_2$  are large and higher order term contributions cannot be ignored. Pick a large integer number  $N$  and divide  $\Delta R_1$  and  $\Delta R_2$  by  $2^N$  such that any triplet  $\left(i \frac{\Delta R_1}{2^N} + R_1, i \frac{\Delta R_2}{2^N} + R_2\right)$ ,  $\left((i+1) \frac{\Delta R_1}{2^N} + R_1, (i+1) \frac{\Delta R_2}{2^N} + R_2\right)$  and  $\left((i+2) \frac{\Delta R_1}{2^N} + R_1, (i+2) \frac{\Delta R_2}{2^N} + R_2\right)$  satisfy (Eq 2.7).  $i \in (0, 2^N - 2)$  is an integer.

Define the following symbols to simplify the use of lengthy notations:

$$A = Q\left(i \frac{\Delta R_1}{2^N} + R_1, i \frac{\Delta R_2}{2^N} + R_2\right) \quad (\text{Eq 2.10})$$

$$B = Q\left((i+1)\frac{\Delta R_1}{2^N} + R_1, (i+1)\frac{\Delta R_2}{2^N} + R_2\right) \quad (\text{Eq 2.11})$$

$$C = Q\left((i+2)\frac{\Delta R_1}{2^N} + R_1, (i+2)\frac{\Delta R_2}{2^N} + R_2\right) \quad (\text{Eq 2.12})$$

$$D = Q\left((i+3)\frac{\Delta R_1}{2^N} + R_1, (i+3)\frac{\Delta R_2}{2^N} + R_2\right) \quad (\text{Eq 2.13})$$

$$E = Q\left((i+4)\frac{\Delta R_1}{2^N} + R_1, (i+4)\frac{\Delta R_2}{2^N} + R_2\right) \quad (\text{Eq 2.14})$$

From (Eq 2.7), the following three inequalities hold:

$$A + C \leq 2B; B + D \leq 2C; C + E \leq 2D$$

Combine the above three inequalities to show:

$$A + E \leq 2C \quad (\text{Eq 2.15})$$

Notice the distance between the coordinates of  $A$  and  $C$  is now twice as large as the distance between the coordinates of  $A$  and  $B$ . We can thus rewrite (Eq 2.10) to (Eq 2.14) with  $N$  replaced by  $N-1$ . The same derivation applies and (Eq 2.15) can be shown to hold.  $N-1$  can then be replaced by  $N-2$ . The iteration continues till  $N$  reaches 1, which is (Eq 2.7). ■

From Theorem 2.1, the *equi-quality* (EQ) points of  $\{(R_1, R_2) | Q(R_1, R_2, d_0) = q_0\}$  thus must form a decreasing, convex curve like the one shown in Figure 2.12 on page 44. Its convexity is assured because it is the inverse of an increasing concave function.

**Example: Coding two independent Gaussian sources**

It is easy to verify that EQ curves of coding two independent, discrete time, continuous amplitude Gaussian sources are convex. It is known that the rate-distortion function of a Gaussian source is:

$$D(R) = 2^{-2R} \cdot \text{variance of Gaussian} \quad (\text{Eq 2.16})$$

Let  $Q(R_1, R_2) = \text{constant} - D(R_1) - D(R_2)$  be the total quality of the two independent sources. Then,  $\{(R_1, R_2) | Q(R_1, R_2) = q_0\}$  can be shown to satisfy the following condition.  $(R_1, R_2)$  forms a convex function.

$$2^{-2R_1} + 2^{-2R_2} = \text{constant}$$

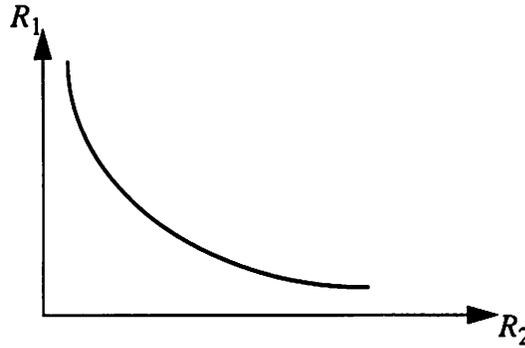


Figure 2.12 An equi-quality function that satisfies  $Q(R_1, R_2, d_0) = q_0$ .

Similar to the quality function, the network cost function  $C$  marginally increases with  $R_1$  and  $R_2$  and marginally decreases with delay offset,  $d$ . As stated in (Eq 2.17), the marginal cost of  $R_1$  is always greater than that of  $R_2$  because of higher connection cost to deliver low delay traffic.

$$\frac{\partial C}{\partial R_1} \geq \frac{\partial C}{\partial R_2} \geq 0, \frac{\partial C}{\partial d} \leq 0 \quad (\text{Eq 2.17})$$

The second-order structure of the cost function is less clear. Nevertheless, one can conjecture the shape of the *equi-cost* (EC) curves. With finite delay bounds,  $R_1$  and

$R_2$  should no longer be measured as long-term averages. Instead they should be estimated as the *effective bit rate* of individual flows (See Box 2.A on page 20). Published work on effective rate addressed separately the scenario in which traffic sources are correlated with the same delay requirements, and the scenario in which independent sources with differential delay requirements in prioritized queues. In the two-flow DCVC, their output traffic, with differential delay requirements, may be correlated. Since this scenario is yet to be studied, we cannot reference published results to justify the following claims on the properties of EC curves. Instead, we make reasonable assumptions.

When the two flows are negatively correlated in traffic, meaning (loosely speaking) an increase in bit rate of one flow corresponds to the decrease of the other, and both have the same delay requirements, it is known that EB of both flows is smaller than the sum of EB of individual flows. To see a simple example, consider the case of two on-off traffic sources with their phases perfectly out of synchronization. When a source is in the on state, it transmits at rate  $R_p$ . When it is in the off state, it stops transmission. The EB of each source is greater than  $R_p/2$  and therefore their sum is greater than  $R_p$ . However, since they are negatively correlated and out of sync, EB of the total traffic is just  $R_p$ . The same argument applies to positively correlated traffic sources.

The two-flow DCVC case has differential delay requirements and thus the above observations on EB cannot be directly applied. Nevertheless, we believe in most occasions network carriers can take advantage of the negative correlation to reduce cost. We assume the EC curves of negatively correlated flows are concave

and the EC curves of positively correlated flows are convex. Figure 2.13 illustrates the two cases.

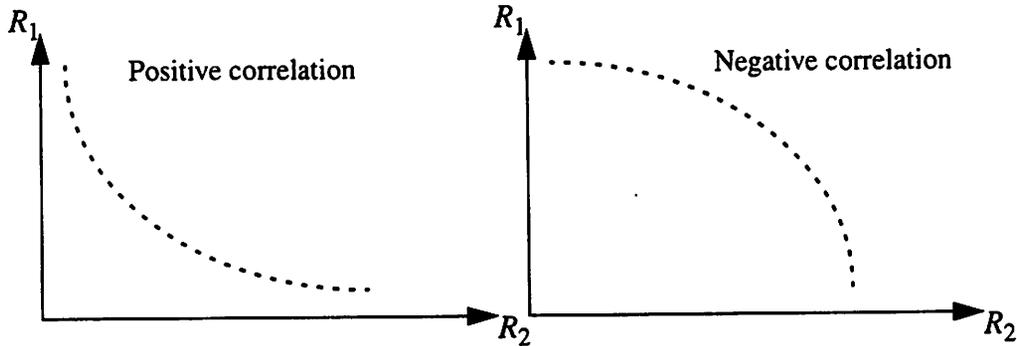


Figure 2.13 An equi-cost function that satisfies  $C(R_1, R_2, d_0) = c_0$ .

The optimization problem may be solved by the Lagrange multiplier method. When a positive multiplier exists, (Eq 2.18) states the sufficient condition for obtaining the optimal values of  $R_1$ ,  $R_2$ , and  $d$ .

$$\frac{\frac{\partial Q}{\partial R_1}}{\frac{\partial C}{\partial R_1}} = \frac{\frac{\partial Q}{\partial R_2}}{\frac{\partial C}{\partial R_2}} = \frac{\frac{\partial Q}{\partial d}}{\frac{\partial C}{\partial d}} \quad (\text{Eq 2.18})$$

From (Eq 2.17) and (Eq 2.18), it is concluded that the marginal quality with respect to  $R_1$  is greater than the marginal quality with  $R_2$ .

$$\frac{\partial Q}{\partial R_1} \geq \frac{\partial Q}{\partial R_2} \quad (\text{Eq 2.19})$$

Since  $Q$  is concave in rates, the optimal rate allocations for  $R_1$  and  $R_2$  must be unique and minimized. Furthermore, (Eq 2.19) indicates the optimal value is biased against  $R_1$ . When  $Q$  is a separable and identical function in  $R_1$  and  $R_2$ , the optimal  $R_1$  is always smaller.

Further understanding of the optimal rate allocations can be approached by analyzing EC and EQ. As shown in Figure 2.14 on page 47, the tangent points of a

given EQ and its minimal EC may happen at the middle of the curve or at one of its terminals. The three cases are summarized and discussed in below.

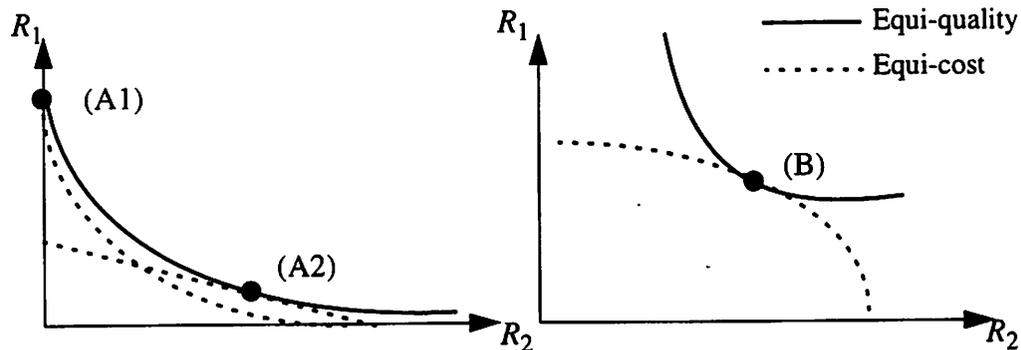


Figure 2.14 Use equi-cost (dotted line) and equi-quality (solid line) curves to find the optimal rate allocations. The three possible optimal operating points are marked with black dots.

*Case (A1):* Both EC and EQ are convex. The curvature of EC is greater than that of EQ. The optimum is at a terminal point.

*Case (A2):* Both EC and EQ are also convex. The curvature of EC is less than that of EQ. The optimum is at a tangential point in the middle of the curve.

*Case (B):* Both EC and EQ are concave. The optimum is always at the middle of the curve.

While (A2) and (B) encourage the addition of the second flow to reduce cost, Case (A1) gives its preference to the single flow model. In (A1), the two flows are positively correlated and the marginal quality gain introduced by the second flow is less than the marginal cost increase. Hence, the use of the second flow is not justified.

When both flows are applied, the optimization settles to a point that minimizes the bit rate allocation to the low-delay flow. Figure 2.15 on page 48 illustrates that in the portion of the line  $R_1+R_2$  above the EQ curve, the optimum has

the smallest  $R_1$ . Unless both flows have the same marginal costs, the optimum typically does not coincide with the value obtained by minimizing  $R_1+R_2$  directly.

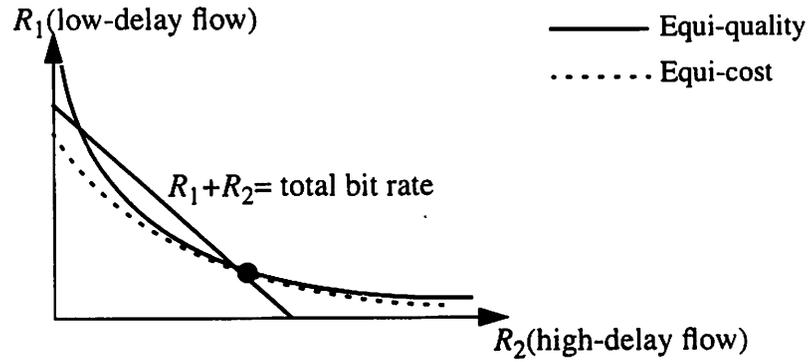


Figure 2.15 The minimization of the total bit rate and its low-delay portion. The marked point has the lowest bit rate for the low-delay flow.

**Example:**

Suppose the cost and quality functions at  $d = d_0$  are expressed in (Eq 2.20) and (Eq 2.21), respectively.

$$C(R_1, R_2, d_0) = 4R_1 + 2R_2 \quad (\text{Eq 2.20})$$

$$Q(R_1, R_2, d_0) = 2\log R_1 + 2\log(R_2 - 3) \quad (\text{Eq 2.21})$$

In this example, both functions are additive of components in  $R_1$  and  $R_2$ . They are separately plotted in Figure 2.16 on page 49. Notice that in this quality expression, in order to reach the same perceptual quality,  $R_2$  needs to be greater than  $R_1$  in order to compensate for the degradation caused by longer delay.

Using (Eq 2.18), we can derive the minimum total cost when both flows are present, expressed in the function of  $q_0$ .

$$C_{min}(R_1, R_2, d_0) = 4\sqrt{2}e^{q_0/4} + 6 \quad (\text{Eq 2.22})$$

There are limiting cases where applying a single flow can result in lower cost. We shall also consider those cases. Their minimal costs can be shown as

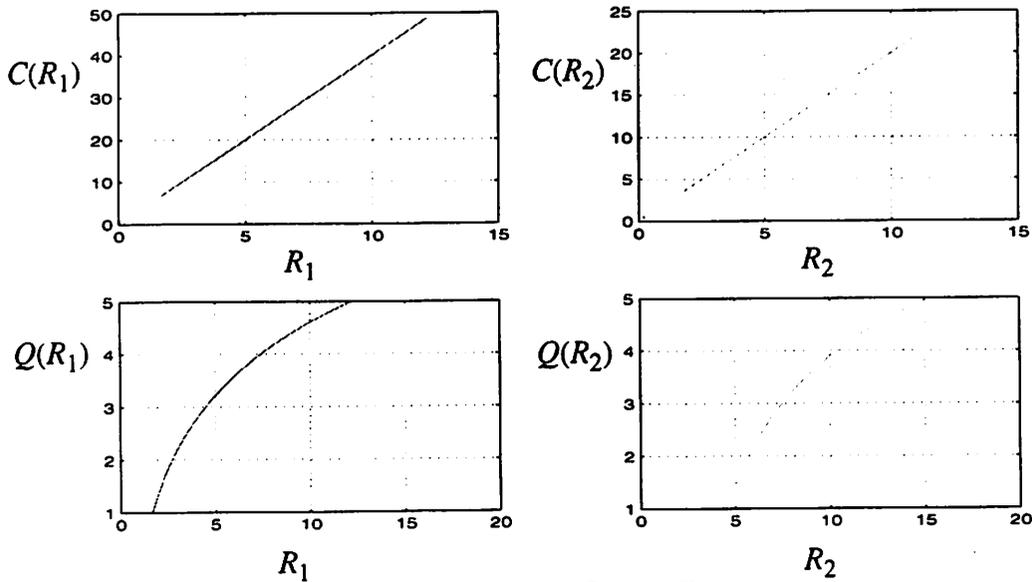


Figure 2.16 The cost and quality functions of  $R_1$  and  $R_2$ .

$$C_{min}(R_1, R_2=0, d_0) = 4e^{q_0/2} \quad (\text{Eq 2.23})$$

$$C_{min}(R_1=0, R_2, d_0) = 2e^{q_0/2} + 6 \quad (\text{Eq 2.24})$$

In Figure 2.17 on page 50, the above three equations are plotted in the range of  $q_0 \in [1, 5]$ . The minimum of the three curves is plotted in Figure 2.18 on page 50. Due to the special structure of the cost function, it is not always cost efficient to apply both flows. In this case, only when  $q_0 > 4.2$ , both flows should be used.

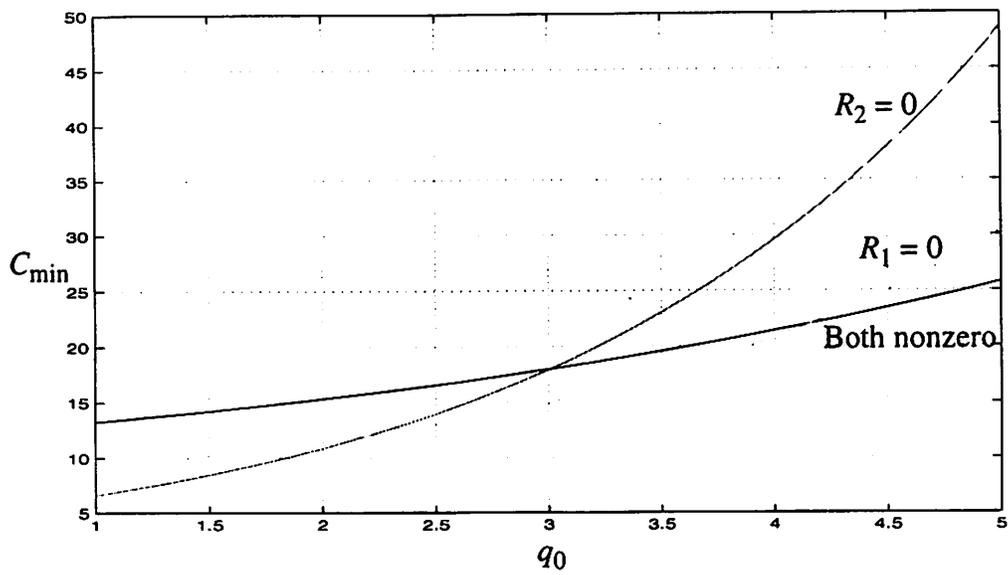


Figure 2.17 The plots of (Eq 2.22)( $R_2 = 0$ ), (Eq 2.23)( $R_1 = 0$ ), and (Eq 2.24) (Both nonzero).

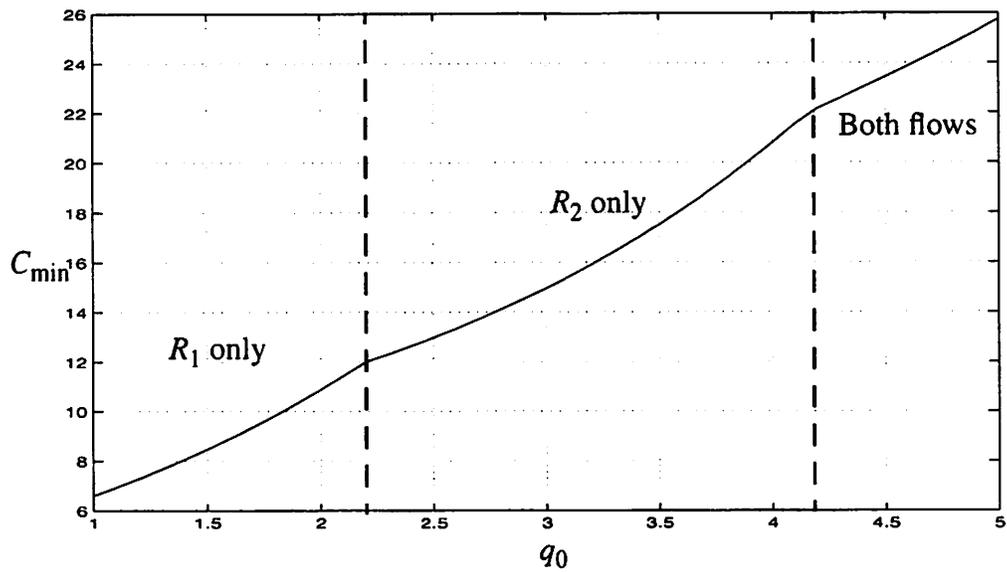
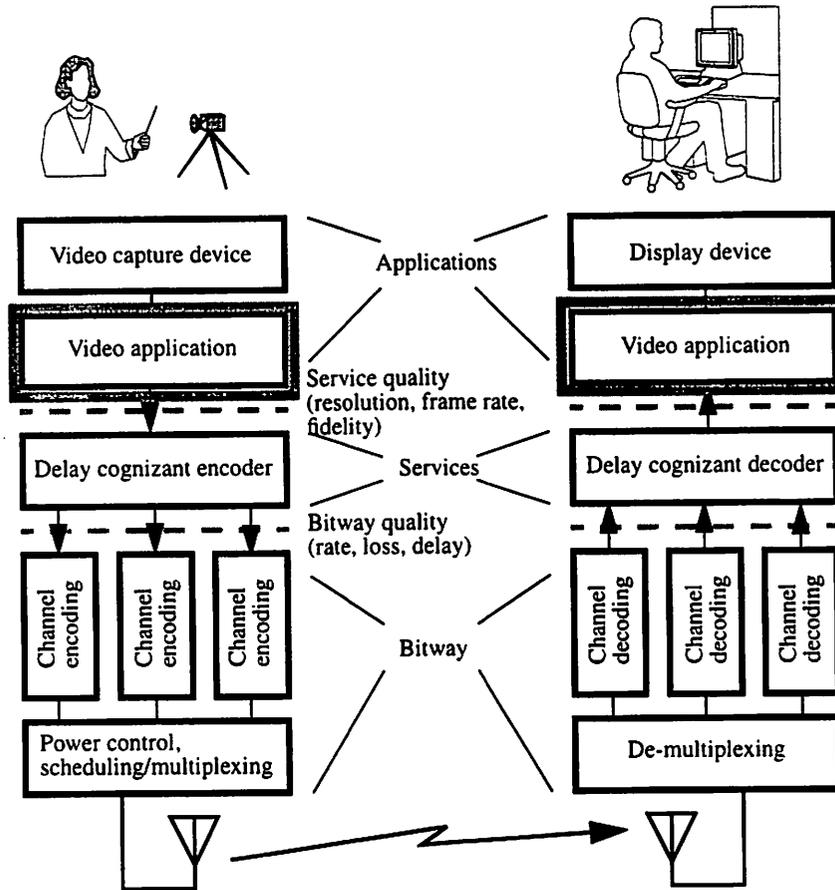


Figure 2.18 The minimum of (Eq 2.22), (Eq 2.23), and (Eq 2.24) with three regions marked.

# 3

## DCVC Applications



Having introduced the codec architecture of delay cognizant video coding in the previous chapter, and promised a number of improvements over conventional coding, in this chapter, we move the theme up to the applications layer to describe how these promises are realized by employing DCVC in various network environments. The realizations are all involved with multi-differential delay flows of DCVC, which carries the most visually significant information in the low-delay

flow and less significant information in the high-delay flow. Because the mechanism of provisioning delay flows varies in different network environments, we divide the chapter into four sections, each of which defines and describes its network and the way DCVC may be applied.

The four network environments discussed are broadband ISDN (B-ISDN) [37], delay/cost differentiated networks, the Internet, and wireless networks. These only represent a small portion of potential candidate networks where DCVC innovates. Nevertheless, they have covered a wide range of in-use and under-developing networks. In terms of service quality provisioning, B-ISDN is at one end of the spectrum and has the most control on quality of service for user applications. Today's Internet is at the other end of the spectrum and offers only best effort services. A new paradigm of delay/cost differentiated networks (See Box 1.B on page 5 about voice over IP) is emerging as an alternative of the above two to provide a limited class of services at different costs. Furthermore, we are seeing wireless network services to migrate from pure voice telephony to multimedia data. Wireless networks, with the time varying channel and the high bit error rate, offer one of the most challenging environment to video coding.

It is important to keep in mind that although the four networks are discussed separately, DCVC changes neither its segmentation method nor its compression algorithm to tailor for specific networking characteristics. Should the changes be made, the loosely coupled joint source-channel coding paradigm would be violated and one would have to design as many 'versions' of DCVC as the number of networks. As we have argued in Section 1.3 and in [32], this tightly coupled paradigm does not scale to heterogeneous network environments. This chapter demonstrates how differential delay flows can be mapped to different network channels/priorities in order to efficiently exploit network resources. When flows come through network boundaries, flow structures are restored and passed to the next link, which may utilize flows in a totally different fashion.

We first introduce video on B-ISDN, where we demonstrate that DCVC utilizes the residual bit rate to improve quality. Delay/cost differentiated networks are described next. We show significant cost savings can be attained with DCVC. We then demonstrate how DCVC can be applied to Internet video to increase its error resiliency. Finally, we show DCVC adapts to time-varying, variable link speed wireless networks.

### **3.1 Video on Broadband ISDN**

Broadband integrated services digital networks (B-ISDN) [37] was designed to provide subscribers with high-speed digital channels that make possible a slew of new services, including digital TV, hi-fi audio, multimedia communications and retrieval. Asynchronous transfer mode (ATM) is the recommended networking transport protocol of B-ISDN. ATM divides all information into short, fixed length cells of 53 bytes, which allows simple and fast hardware switching. One motivation to integrate all services onto a single platform is to increase traffic efficiency. One aspect of efficiency is statistical multiplexing: multiple bursty traffic sources are multiplexed in such a way that their peaks do not coincide. Since the traffic peaks of some sources may fit in the troughs of other sources, the total capacity needed will be less than the sum of the peaks, which implies bit rate reservations can be made smaller.

While the potential gains of statistical multiplexing can be large, a traffic characterization is needed to estimate this effective rate for the purpose of admission control, since peak rates can no longer be applied. In the networking research community, this problem has long been recognized and numerous work has been done to estimate the effective bit rate (or effective bandwidth) of stochastic, bursty traffic sources, most prominently variable bit rate (VBR) video [21][28][36][68]. The commonly used queuing model is shown in Figure 3.1, where incoming traffic waits in a shared buffer. The switch reads out ATM cells out of the buffer at a constant rate and passes them to the switch fabric. At a switch, the effective rate of a stochastic source characterizes the bit rate that must be reserved to guarantee a

small, nonzero packet loss probability (typically  $10^{-5}$  or lower). By allowing a small loss probability, more video connections can be statistically multiplexed to the same link, thereby improving traffic efficiency. Due to the bursty nature of video, this traffic capacity gain can be significant. Effective bit rate is a function of loss probability, the size of the switching buffer, and the stochastic model of the traffic source. At a fixed link rate, the size of the switching buffer poses an upper bound on maximum queuing delay.

Prior networking research has proposed a number of formulations to characterize effective rate. Regardless of modeling details, these proposals all share some basic properties. First, the effective rate of a bursty source is greater than its average rate and less than its peak rate. Second, the effective rate converges asymptotically to the average rate when the buffer size grows to infinity. Third, it converges asymptotically to the peak rate when the buffer size reduces to zero. Finally, effective rate is additive across multiple sources and to satisfy the constraint on loss, the sum of all incoming traffic must be less than the link rate.

While the bit rate reservation is made at the estimated effective rate, the actual transmitted video traffic over time is still at the average rate. The residual difference (residual bit rate) between the effective and the average is nonzero, and is wasted if not used. Conventional video coding cannot make use of this residual because the residual availability depends on traffic activities of other sources, which are unknown to the encoder. An example is illustrated in Figure 3.2, where two on-off bursty sources and the sum of their traffic are shown. The residual avail-

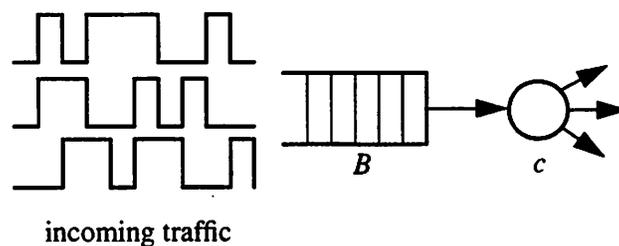


Figure 3.1 The queuing model of a network switch consists of a single, shared buffer of size  $B$  and a fixed rate server  $c$ , followed by the switch fabric.

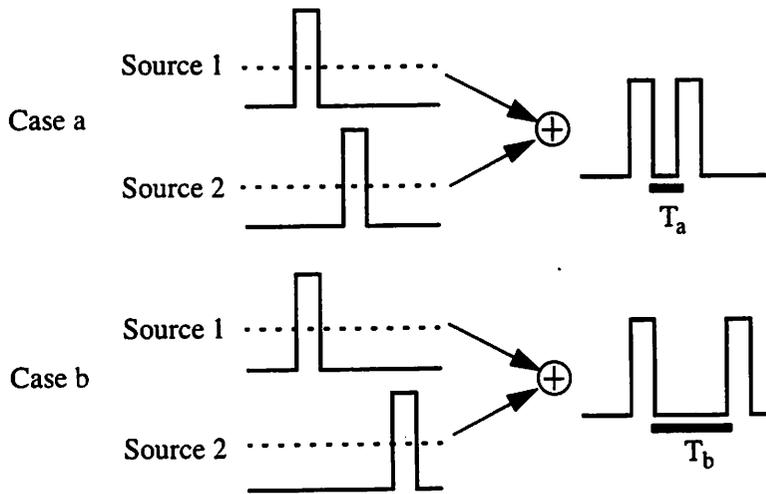


Figure 3.2 An illustration to show the availability of the residual rate depends on traffic behavior of other sources. Solid lines represent actual traffic arrivals and dot lines represent their effective bit rates. Residual rate becomes available when neither of the two sources are sending bursts. As the two cases illustrate, Source 1 cannot predict in advance the length of the available period.

able periods of in case a and b,  $T_a$  and  $T_b$ , vary depending on the arrival times of the bursts. Since the encoder at Source 1 is not aware of the traffic from Source 2 and does not know the lengths of available periods, it cannot apply techniques like closed loop rate control to make use of the residual. Because conventional video coding cannot make use of residual rate, it is not surprising that prior networking research does not address this issue, either.

### 3.1.1 Improving Video Quality

This DCVC application, first described in [15], was motivated by the observation that DCVC can make use of residual rate to improve video quality. The key idea is to leverage the relaxed delay requirement of the high delay flow to fit the residual availability. The DCVC encoder does not need to know the exact moments nor the lengths of available periods,  $T_a$  or  $T_b$ . Instead, the network switch can serve packets in high-delay flows when the low-delay traffic of incoming sources is temporally inactive. The switch functions like a prioritized two-class, single server queue, with the high priority assigned to the low-delay traffic and the

low priority assigned to the high-delay traffic. A schematic diagram of the two-class prioritized switch is shown in Figure 3.3. The effective bandwidth of a DCVC connection is reserved solely based on traffic statistics of the low-delay flow. The use of residual rate does not affect the formulation nor the outcome of statistical multiplexing analysis. It is worth emphasizing that the video quality improvement comes at no extra transport cost, since the residual rate is reserved as a part of the effective rate. The improvement is achieved by carrying the delay critical information on the low-delay flow to establish an initial image and by carrying high-quality but delay-tolerant information on the high-delay flow to progressively improve quality.

To demonstrate the quality improvement with DCVC, we encoded a 15-second video sequence and simulated its transmission through a network switch. The sequence contains 450 frames and is a concatenation of three short clips with 150 frames each. Although the average bit rate of the low-delay flow is 30 Kbps, its peak rate is almost 17 times more than the average due to intra-coding at scene changes. Rate control was applied to the high-delay flow to reduce the number of blocks encoded at each frame and to improve the quality of encoded blocks. We consider a network switch with a capacity of 1.5 Mbps. The maximum queuing delay of the low-delay flow is set to be 400 ms, which is equivalent to a buffer size of 600 Kbits. The packet loss probability must be  $10^{-6}$  or lower.

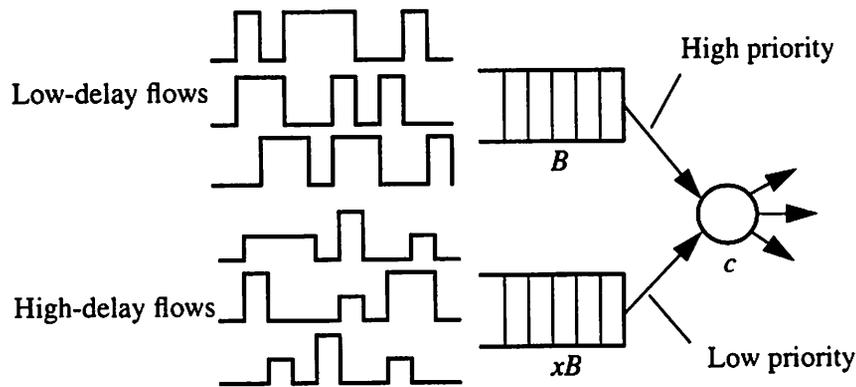


Figure 3.3 The queuing model of a two-priority-class network switch consists of two buffers, one for each flow, and a fixed rate server.

The low-delay flow of the compressed video sequence is approximated by the two-state Markov model described in Section 3.6 on page 69. Parameters used in computing its effective bit rate are:  $\mu_1 = 26.5$  Kbps;  $\mu_2 = 512$  Kbps;  $p_{12} = 1/149$ ;  $p_{21} = 1$ ;  $\delta = 2.3 \cdot 10^{-5}$ . The effective rate of the low-delay flow is 161 Kbps, so less than ten of which can be admitted to the network switch simultaneously. With random starting points through the duration of video, we first simulated ten such sequences with the low-delay flow only and observed no violations of the given loss probability. We then added the high-delay flow traffic to our simulation to observe its maximum queuing delay. Although the high-delay flow has a lower transmission priority, in the several hundred simulations the maximum waiting time in this lower priority queue never exceeded 90 msec.

We compared the perceived video quality of a sequence with the low-delay flow only and with both flows. Although the actual delay experienced by the high-delay traffic may be time-varying, we used the worst case in which all packets in the high-delay flow lag their counterparts in the low-delay flow by 99 msec. An informal subjective evaluation by graduate students favored the two-flow DCVC-

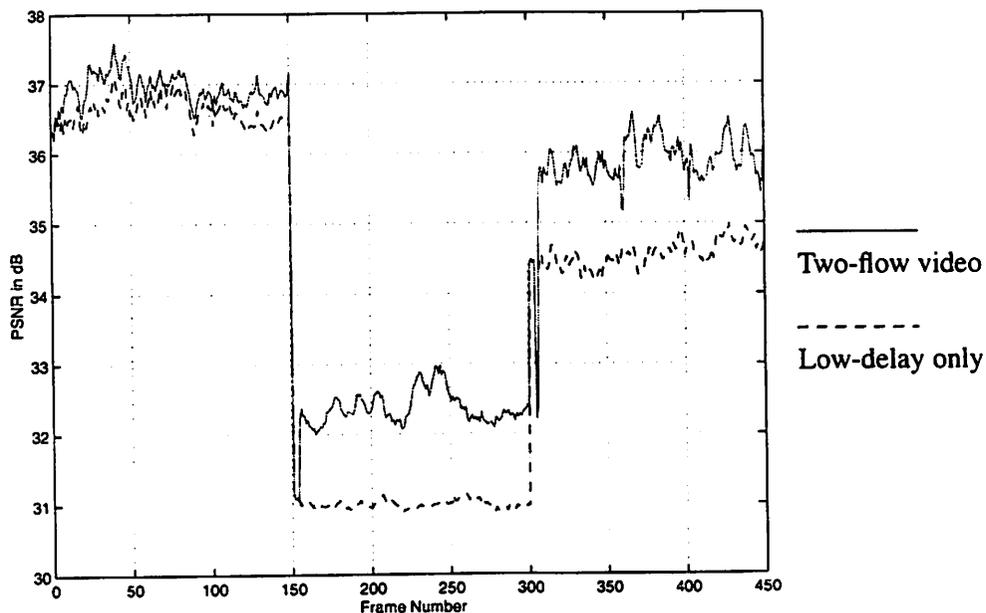


Figure 3.4 PSNR comparison in video quality of the low-delay-flow-only vs. DCVC two-flow video sequences.

coded video. The peak signal-to-noise ratio (PSNR) of both conditions, when compared with the original, as shown in Figure 3.4, was always positive, and sometimes exceeded 2dB.

### 3.1.2 Increasing Video Capacity

This application demonstrates how DCVC can deliver equal subjective quality at a lower effective rate, thereby increasing the traffic capacity of the network. Recall that in the previous application, as much as 2dB increase in PSNR, can be achieved through the addition of the delay-tolerant high-delay flow. Our approach to increase capacity is to convert the 2dB quality gain into bit rate savings. Its queuing paradigm for DCVC is the same as the first application's: a two-class prioritized queue with the high priority assigned to the low-delay flow. The high-delay flow is sent in the residual rate of the low-delay flow. Comparisons are made against single flow conventional coding. With everything else being equal, while a single flow encoder compresses the video at the quantization step size  $\Delta$ , the two-flow DCVC encoder assigns a step size larger than  $\Delta$  to the low-delay flow and a step size less than  $\Delta$  to the high-delay flow. The larger the quantization step size, the poorer the quality and the lower the bit rate. A DCVC encoder adjusts both step sizes to deliver the same quality as the single-flow case. The effective bandwidth of DCVC video can be shown in the following example to be 30% less than the effective bandwidth of conventional video. Therefore, every two conventional video connections carried in the network can be replaced by three DCVC connections, for an increase in traffic capacity of 50%.

We used a H.263 coder and the DCVC coder to encode the 15-second Salesman sequence. The quantization step size  $\Delta$  of the H.263 coder was set to 16 (for uniformly quantized DCT coefficients in inter-coded blocks). The step size of the low-delay flow of DCVC was set to 20 while that of the high-delay flow was set to 10. Rate control of the high-delay flow made at most 10% of the total blocks encoded in each frame. Rate control is necessary to adjust the bit rate of the high-delay flow to be less than the residual bit rate of the low-delay flow. The PSNR

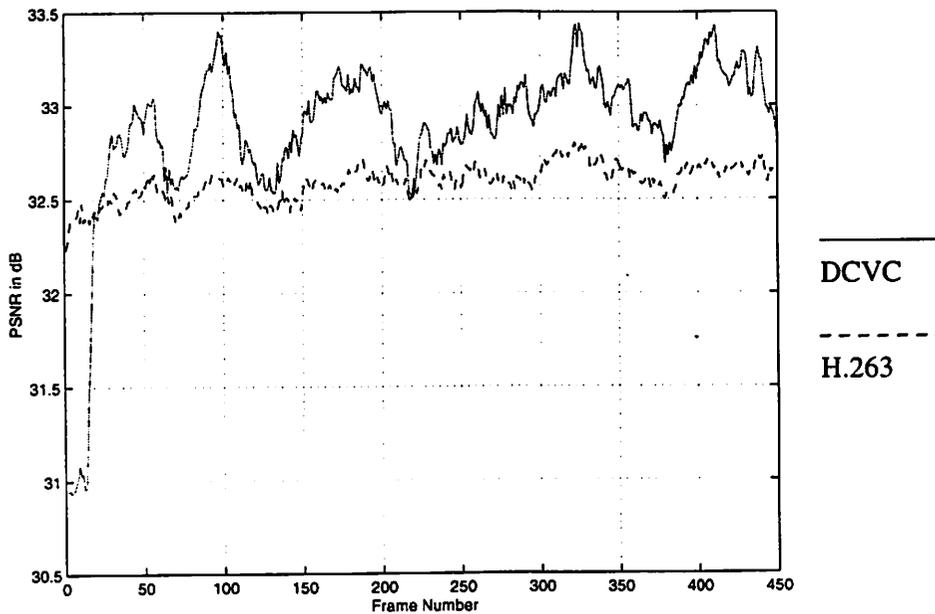


Figure 3.5 PSNR plots of H.263 and DCVC video to show comparable quality with 30% less bandwidth saving for DCVC.

measured quality of the H.263 stream and the quality of DCVC video with a 10-frame delay offset is plotted in Figure 3.5. DCVC video has a higher PSNR most of the time, except for the first 20 frames, where the low-delay flow establishes an initial, coarse representation and waits for the high-delay flow to gradually improve the quality.

We again applied the two-state Markovian traffic model described in Section 3.6. The effective bit rate of the H.263 stream is estimated to be 158 Kbps. The effective rate of the DCVC low-delay flow is 110 Kbps, which has sufficient residual bit rate to carry the high-delay flow. As shown from these numbers, DCVC requires 30% less bandwidth to deliver the same quality.

### 3.2 Video on Delay/Cost Differentiated Network

Differentiated service networks (DSN) are emerging amid their simplicity in system management and control. While multiple services may be still be carried by the same physical link, they are logically separated by either fixed bandwidth allocations or priority assignments. Constraining the number of service classes and

their QoS requirements simplifies demand estimation and bandwidth reservation, both of which are more difficult issues in the general context of integrated service networks (ISN). As many companies have announced their plans to offer voice over IP services, (see Box 1.B on page 5) this new DSN paradigm may be realized sooner.

Consider the two-tier IP networks, voice and data, as the simplest DSN. Connection costs associated with the two services differ in the orders of magnitude. Voice over IP, having low loss and low latency, is charged by the minute. IP data access, on the other hand, is charged by a flat monthly rate with unlimited usage. IP data packets, however, have 30% chance of being dropped during high usage hours. This delay/cost differentiated network is ideal for delay adaptive applications like DCVC.

DCVC can exploit the delay/cost differentiated network to minimize connection cost while maintaining a similar video quality. One of the objectives of DCVC, minimizing the low-delay traffic, fits nicely to the cost minimization criterion. Provided that the total cost of connections is defined as the cost sum of usage rates of individual flows and costs grow linearly with usage, the total cost  $C_T$  can then be expressed as follows.

$$C_T = P_l \cdot R_l + P_h \cdot R_h \quad (\text{Eq 3.1})$$

$P_{l(h)}$  is the unit rate cost of the low-(high-) delay flow.  $R_{l(h)}$  is the usage rate of the low-(high-) delay flow. In practice, the usage rate may be defined as the effective bit rate or the average rate. Were the effective rate applied, Section 3.1.2 showed the effective rate could be reduced by as much as 30% for the sequence tested, a 30% cost saving. Were the average rate applied, Table 2.2 indicated 10-15% savings are achievable.

### 3.3 Internet Video

Today's Internet adopts no admission control and in most cases, any network connected device can send data packets without preauthorization. These data packets, however, are not guaranteed to reach their destinations. When a link on the route becomes congested because of heavy traffic, the switch flushes its buffer and packets waiting in the queue are dropped. A number of studies showed packet loss can be as severe as 30% as during high usage hours (8 am to 6 pm) on weekdays and less than 1% on weekends [41]. Most network applications employ end-to-end transport protocols like Transmission Control Protocol (TCP) to monitor and control the transmission rate. When a packet is lost (judged by time out mechanisms), TCP uses the back channel to inform the sender to retransmit the packet.

While TCP or similar reliable transport protocols are necessary for computer data communications, it incurs much higher latency, due to the roundtrip latency of retransmissions, which is not suitable for interactive applications. Since continuous media such as video has a relaxed demand on reliability, various error recovery and concealment techniques for video have been proposed to increase its robustness to packet losses [23][33][75][81]. TCP is replaced by User Datagram Protocol (UDP), since UDP does not perform retransmission and any correctly received packet has only one-way latency. For lost packets, video coding algorithms must then conceal the error to minimize visual artifacts.

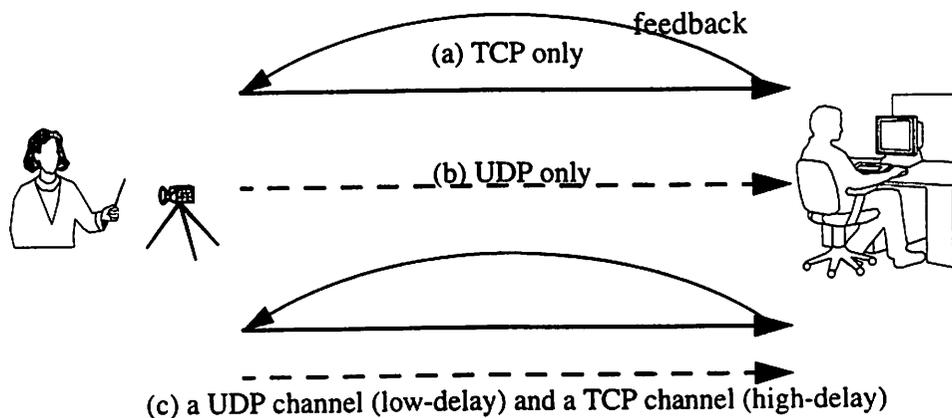


Figure 3.6 Internet video can be carried by (a) TCP only; (b) UDP only; (c) both UDP and TCP with different latencies.

Current Internet video applications apply either TCP or UDP to video transmissions. As shown in Figure 3.6, conventional video coding is constrained to the first two options: using TCP for reliable delivery and suffering from long delay, or using UDP for low perceptual delay and suffering from visual artifacts. A solution that incorporates the advantages of both TCP and UDP is needed.

DCVC presents a potential solution to the dilemma: Internet video with low perceptual delay and few artifacts. Since either TCP or UDP has its own unique advantages, DCVC applies them both. This is illustrated as the third option in Figure 3.6, where the UDP channel is matched to the low-delay flow and the TCP channel is matched to the high-delay flow. Information carried by the low-delay flow tends to stay on the screen for a shorter time and it is often quickly replaced by new scenes. Therefore, the low-delay flow can be sent over an unreliable channel like UDP. On the other hand, information carried by the high-delay flow tends to stay longer. Visual artifacts, if any, would also stay longer. It is thus necessary for the high-delay flow to have a reliable transport channel like TCP. Since the flow tolerates longer delay, extra latency caused by TCP retransmissions is tolerable.

Figure 3.7 illustrates an example comparing the UDP only and UDP/TCP solutions. To emphasize the visual artifacts due to packet loss, we chose an example of introducing the loss in the middle of scene changes. In this figure, the speaker is being switched from the lady in the first image to the man behind the desk in the second image. Due to the packet loss at the UDP channel, the lower half of the new image is lost and the corresponding area from the lady image remains on screen. Because of the motion compensation technique applied in compression, corrupted image areas propagated from one frame to the next. On the third row of the figure, images from the 40th frame after the packet loss are shown and visual artifacts are obvious in both solutions.

There is an important difference between the second and the third images on the third row, however. In the second image, both the suitcase and the telephone

on the man's desk were recovered while in the third image, none of them appeared. The second image, which is generated by the UDP/TCP solution, was a composite of video frames from the low- and high-delay flows. The suitcase as well as the telephone are backgrounds and they are sent through the reliable TCP channel. Their images are thus not affected by packet loss and the ensuing error propagation.

This simple example demonstrates DCVC, which was motivated by differential delay networks, can be applied to address the error robustness issue effectively. Although the example showed the UDP/TCP solution was more resilient to packet loss, the low-delay flow was not immune to the loss. From the figure, we



Figure 3.7 Packet loss at scene changes can be disastrous, as illustrated by the above images. From left to right are images from the outputs of uncompressed, DCVC with UDP/TCP, and UDP only. Images in the top row are received before the packet loss. Images in the middle row are from one frame after the loss and those in the bottom row are from 40 frames after the loss.

found the high motion area was affected most by error propagation. For differentially coded video, error propagation is unavoidable unless higher levels of redundancy are introduced in the coded bitstream. For motion-compensated video, redundancy introduction can be more effective if the motion information is utilized.

The author, along with Dr. Marc Willebeek-LeMair and Dr. Zon-Yin Shae of IBM Research, proposed an effective redundancy introduction algorithm named intelligent macroblock update (IMU) in [83]. IMU evaluates the relative importance of a macroblock by calculating how much image area in future frames depends on it. For example, in Figure 3.8, the block *A* in frame *N* is referenced partially by two blocks in frame *N*+1, four blocks in *N*+2, and four blocks in *N*+3. If block *A* were corrupted, visual artifacts would incur in those referencing blocks. To prevent the error propagation, the block is intra-coded, meaning it is compressed without using motion compensation. At the expense of a higher bit rate, any earlier packet loss caused propagation stops at block *A*. Referencing blocks in later frames will thus be error free, provided that no more packets are lost. IMU was shown to work well at the same bit rate, comparing against static intrablock coding schemes. The main drawback of IMU, however, is that the algorithm must accumulate a number of frames before it can decide what blocks to be intracoded. Frame accumulation creates additional delay, which is fine for Internet streaming video but discouraged by interactive applications.

### **3.4 Wireless Video**

The wireless multiaccess network, due to its limited bandwidth and high bit error rate (BER), is likely to be the bottleneck link in an end-to-end connection. With over 100 million digital cellular subscribers worldwide, applications with richer functionality and higher bandwidth are in demand. Video over wireless is a challenging research topic, which attracted much attention. [52] described the Wireless Andrew project at Carnegie Mellon University and a video codec that

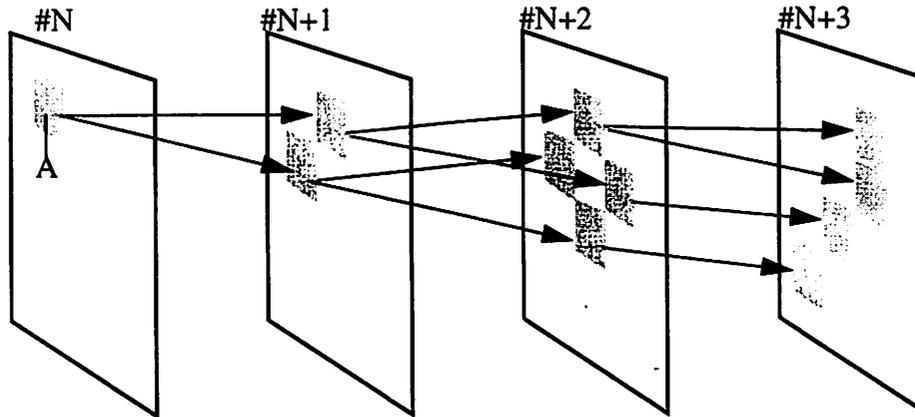


Figure 3.8 An illustration of block dependency due to motion compensation and differential coding.

adjusted the compression ratio according to the wireless link throughput. [49] proposed a pyramid vector quantization scheme, which applied a fixed rate quantizer to reduce the susceptibility to bit error propagation. [86] approached the problems of sending MPEG2 video over wireless by special framing and packetization coupled with system and video layer concealment. Most of the published work assumed a fixed BER of the link and applied coding techniques tailored for the targeted BER. However, the assumption was inadequate and lead to inefficient utilization of the channel, because link dynamics are ignored.

Unlike a wired link, whose channel condition and transmission rate typically do not change, a wireless link may vary in a wide range of channel conditions in a short time. Its BER is affected by many factors, including the speeds of receivers and transmitters, the motion of ambient objects which may block or attenuate radio signals, and the surrounding area. Propagation measurements reported as much as 50 dB drop in signal strength was observed in fading periods [3][31]. Figure 3.9 illustrates a simple fading phenomenon by plotting the electromagnetic fields of a standing wave pattern, which is formed by adding two out-of-phase waves. As a mobile receiver moves through the field, the amplitude of the received signal fluctuates in a period of every half wavelength. The time spacing between fades depends on the carrier frequency and the receiver moving speed. For example, a cellular carrier frequency of 900 MHz and a mobile user walking speed of 2 m/sec result in an inter-fading time of 160 msec. While techniques such as power

control and error correction codes can be applied to relieve the severity of the channel fading, available bit rates for data transmissions may be reduced, as shown in the figure.

The time varying nature of available rate suggested a more efficient utilization of the channel can be achieved by making the video coding algorithm adapt to available rate variations. [46] reported such a codec design that measures the short-term channel throughput and controls its output rate to match the throughput. This approach works well when the measurement is feedback in time for coding adjustment, which is feasible for the uplink transmission. The video server for the downlink transmission, however, may not be located near the wireless link to respond to the fast changing channel condition.

DCVC is suitable for wireless channel adaptation by sending the high-delay flow to make use of the fluctuation of available rate. Since the low-delay flow (or other single flow video) is constrained by its maximum delay requirement, it

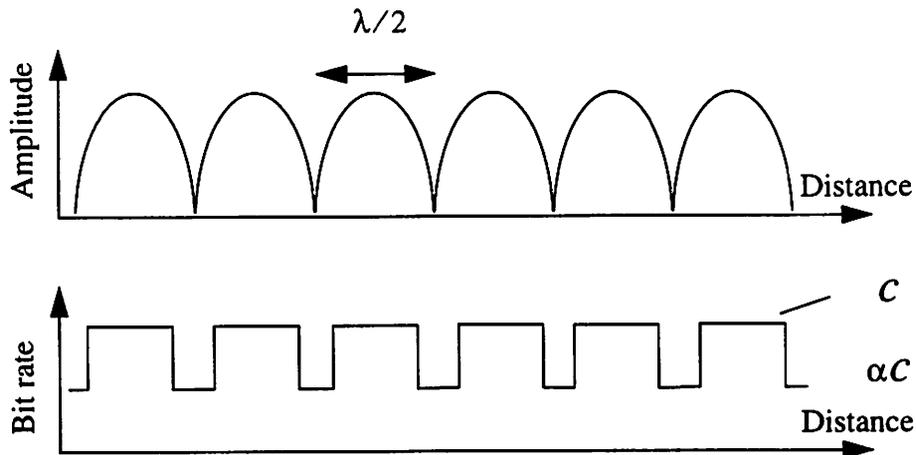


Figure 3.9 As a mobile travels through the standing wave pattern, it will experience fades once every half wavelength. The standing wave pattern shown at the top is a simple addition of two waves with equal amplitude but 180 degrees out of phase. During the fading periods, weaker received signals require stronger error protection and thus the available bit rate is smaller.  $C$  is the available rate outside of fading and  $\alpha C$  is the rate in fading.  $\alpha$  is less than one.

cannot completely take advantage of the fluctuation. Given the maximum delay  $D$ , the peak arrival rate  $R_p$ , of the low-delay flow is bounded as follows.

$$R_p(D) = \inf \int_t^{t+D} R_{av}(u) du \quad t \in [0, \infty) \quad (\text{Eq 3.2})$$

The above equation states the peak arrival rate must be smaller than any window sum of available rates. Should the upper bound be exceeded, the maximum delay requirement could be violated. It is easy to show the following relationships.

$$D_1 < D_2 \Rightarrow R_p(D_1) \cdot \frac{D_2}{D_1} < R_p(D_2) \quad (\text{Eq 3.3})$$

$$R_A = \lim_{D \rightarrow \infty} \frac{R_p(D)}{D} \quad (\text{Eq 3.4})$$

The allowable burst size increases as a longer delay is tolerated. When there is no delay requirement, the time average approaches the long term average rate,  $R_A$ .

Suppose the delay requirements of the low- and high-delay flows are  $D_1$  and  $D_2$ , respectively. The peak rate allowed for the low-delay flow,  $R_p(D_1)$  can be estimated by (Eq 3.2). The peak rate allowed for the high-delay flow is  $R_p(D_2) - R_p(D_1) \cdot \frac{D_2}{D_1}$ , which is the difference of the maximum available of  $D_2$  and the maximum expected arrival of  $D_1$ . The queuing model of the wireless link is

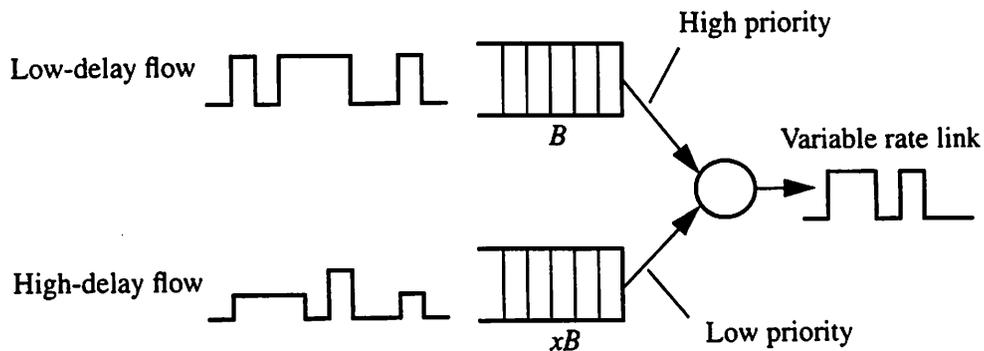


Figure 3.10 The queuing model of two-priority-class wireless link, which has a time varying link speed.

shown in Figure 3.10. It differs from Figure 3.3 on page 56 in that the service rate is variable. Video quality improvement using the high-delay flow can be shown to be similar to the conclusion reached in Section 3.1.1.

### **3.5 Summary**

In this chapter, we demonstrated DCVC applications in four networking environments: B-ISDN, delay/cost differentiated networks, Internet, and wireless networks. DCVC can be applied to B-ISDN to make use of residual rate for improving video quality and increasing capacity. It can be used to reduce connection costs in the emerging delay/cost differentiated networks. Internet video can benefit from DCVC to increase error resiliency while still maintaining low latency. DCVC can also be applied to time varying wireless links to make use of available rate to improve video quality. Although it appears applications in these networks are different, fundamentally they are very similar. All applications exploit the longer delay channels of networks: residual rate in B-ISDN, low cost/high delay network, TCP connection, or available rate in wireless. We envision for communication infrastructures in the future, as long as they have differential cost/reliability/delay channels, DCVC can be applied.

While the visual quality metric shown in the figures of this chapter is the commonly used peak signal to noise ratio (PSNR), we are aware of its mismatch with results obtained by quality evaluation participated by human subjects. DCVC quality is much concerned because there were no prior studies on asynchronously rendered video. In fact, very little is known about conventional, synchronously rendered video and a good computation model that mimics human vision systems is yet to be developed. Recognizing the need to verify DCVC quality, in the next chapter, we focus on discussing related issues about human vision and presenting the results of subjective quality evaluation.

### 3.6 Appendix A: Effective Rate

In the following, one effective rate formulation is briefly described without the complete derivation. Significant amount of research on effective bit rate (or effective bandwidth) in recent years leads to a number of different formulations and proposals [21][28][36][68]. While these works differ in the stochastic models for traffic streams, they are essentially based on large deviation estimates of the loss probability in the asymptotic regime of large buffers. As the buffer size increases, the loss probability approaches zero at an exponential rate. As one might have expected, for a fixed buffer size, the effective bandwidth of a source approaches its peak rate when this probability decreases to zero.

Activities of compressed video traffic can often be fairly accurately modeled by a Markov-modulated fluid with a discrete number of states. Each state is assigned a bit rate and when the traffic source is at state  $i$ , it transmits at the bit rate  $\mu_i$ . The state transitions of the Markov chain reflect the bit rate changes of the output traffic. Let the required loss probability be expressed as  $e^{-\delta B}$ , where  $B$  is the buffer size. It is shown in [36] that the effective bandwidth of the model can be expressed as  $\Lambda(\delta)/\delta$ , where  $\Lambda(\delta)$  is the log spectral radius function of the matrix  $\begin{bmatrix} P_{ij} \cdot e^{\delta \mu_i} \end{bmatrix}$ .  $P_{ij}$  is the transition probability from state  $i$  to state  $j$  and  $\mu_i$  is the bit rate at state  $i$ .

While a model of many states can be developed to characterize compressed video, we found the compressed video streams used in our experiments could be adequately approximated by a Markov model with only two states. The above stochastic matrix can be shown to be:

$$\left[ P_{ij} \cdot e^{\delta\mu_i} \right] = \begin{bmatrix} (1-p_{12})e^{\delta\mu_1} & p_{12}e^{\delta\mu_1} \\ p_{21}e^{\delta\mu_2} & (1-p_{21})e^{\delta\mu_2} \end{bmatrix} \quad (\text{Eq 3.5})$$

Its log spectral radius function is the logarithm of the largest positive eigenvalue, which has a simple closed form solution for this 2x2 matrix.

$$\Lambda(\delta) = \log \frac{b(\delta) + \sqrt{b^2(\delta) - 4a(\delta)}}{2} \quad (\text{Eq 3.6})$$

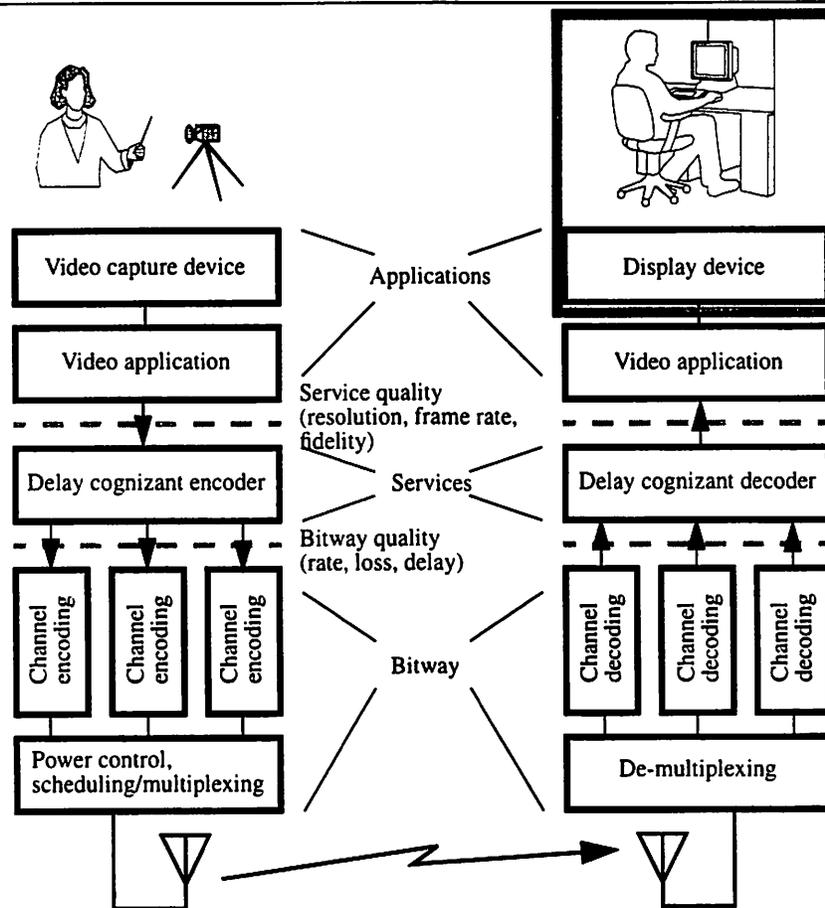
$$a(\delta) = (1-p_{12}-p_{21})e^{\delta(\mu_1+\mu_2)} \quad (\text{Eq 3.7})$$

$$b(\delta) = (1-p_{12})e^{\delta\mu_1} + (1-p_{21})e^{\delta\mu_2} \quad (\text{Eq 3.8})$$

The above set of equations are used in calculating effective rates mentioned in the main text.

# 4

## Quality Evaluation



Delay cognizant video coding (DCVC) applies asynchronous video rendering to composite display images from differential delay flows. Asynchronous rendering causes video regions originated from the same camera capturing moment to appear at the receiver's display at different times. Although we have demonstrated potential advantages of DCVC applications in various networking environments in the previous chapter, we need to ensure that quality degradation caused by asyn-

chronous rendering, if there is any, is acceptable even with a large delay offset. In the previous chapter, we applied the commonly used peak signal-to-noise ratio (PSNR) as a measure of video quality. The PSNR measure is known to be largely correlated with subjective quality but significant discrepancies do exist [26]. Moving up one layer in the chapter-front diagram, the theme of this chapter is to examine the quality of DCVC video by both formal subjective evaluation and computational methods.

We first briefly review relevant work about human vision modeling, subjective testing methods, and computational tools. We then discuss the fidelity characterization of DCVC video. The focus of the chapter is its quality characterization using both psychophysical and computational evaluations. Psychophysical studies rely on the participation of human subjects, who were shown video clips and were asked to judge their quality. For computational modeling, we used the traditional PSNR measure as well as a video quality metric developed by Lambrecht [42][43]. The analysis of subjective test data revealed an interesting finding - DCVC sometimes improves quality. We examined the natural scene content that lead to this conclusion and constructed simple artificial stimuli to mimic those scenes. The simple stimuli can help researchers build better vision models and improve their compression algorithms.

## **4.1 Background**

Understanding visual perception is crucial for developing measures of image and video quality. No matter how sophisticated mathematically a video coding algorithm becomes, the final recipient of the video is the human eye and the ultimate performance metric of a video coder is subjective quality. It is therefore natural to incorporate some key aspects of human vision systems (HVS) into the coding algorithm for two purposes. First, a good HVS model provides guidelines on the allocation of bits to the output components that contribute most to visual quality. Second, HVS model can be used to develop reliable metrics to predict subjective quality, which may obviate the need for extensive subjective testing.

Despite the wide recognition and advocacy of such methodologies, these attempts achieved limited success. HVS research has made much progress in understanding the processing of still images but not motion pictures. As a result, recent video compression standardization activities such as MPEG2, MPEG4, and H.263 still conduct formal subjective quality evaluations.

Early HVS studies established fundamental principles that still are widely applied today. The discoveries include contrast sensitivity functions of spatial and temporal frequencies, contrast sensitivity variations in the presence of masking, critical flicker frequency that determines the CRT refresh rate, and the subsampling of chrominance space. Knowledge about HVS helped image and video processing in almost every area. For example, the DCVC algorithm applies higher detection thresholds for high spatial frequencies because of their lower sensitivity; the quantization of DCT coefficients favor low spatial frequencies; and the luminance component of video information is stored in higher resolution than the chrominance components. While HVS research made significant progress in the past fifty years, the understanding of higher cognition functions is limited and unsatisfactory. Modern computers can perform in a second the same amount of calculations a human does in a year. However, knowledge in computer vision lags behind for simple tasks such as pointing out a dog in an image. As MPEG4 includes object coding as a part of the standard, unsupervised object recognition and extraction are expected to promote more research efforts.

The advance of digital imaging in the last decade has prompted the development of computational models for measuring and predicting image fidelity. Prior work by Watson [82], Daly [20], Lubin [48], and Lambrecht [42][43] took into account contrast sensitivities across spatio-temporal frequencies as well as associated localized masking. Because there is a substantial agreement on the basic functional response of physiological mechanisms in the visual pathways of the brain, these proposed vision models share a similar basic architecture. Figure 4.1 shows the high level flow diagram of this architecture. The model takes two sampled

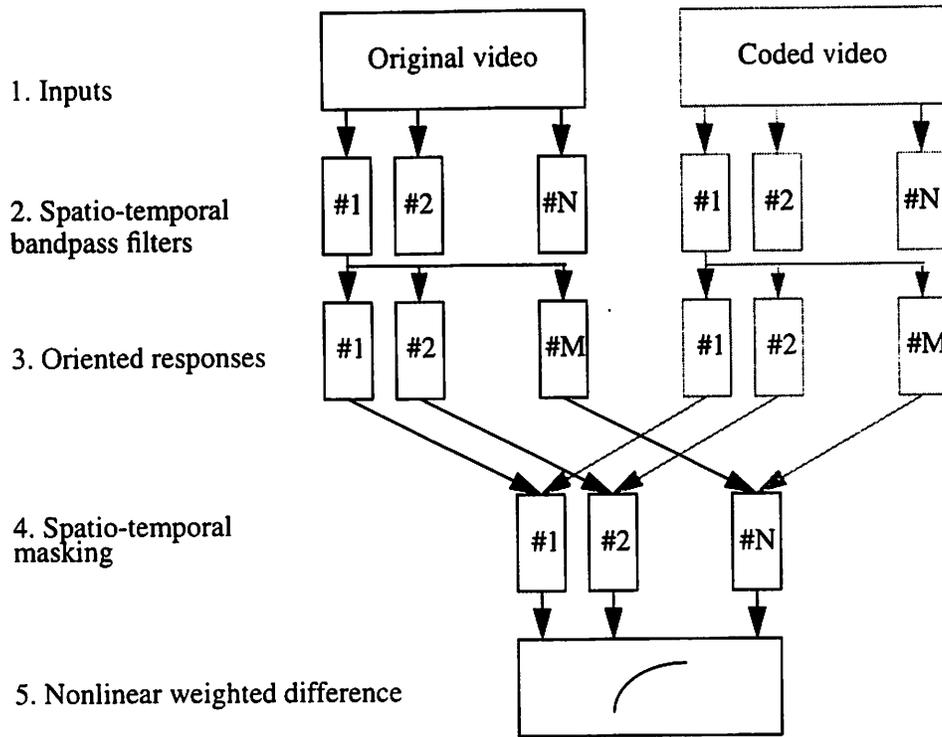


Figure 4.1 Flow diagram of a computational visual fidelity metric.

inputs: original, uncoded video as the reference, and coded video as the test target. Typically, at this stage, several additional observer parameters are entered, including the distance of the observer from the display, the frame rate, and the size of the images expressed in degrees of visual angle. The second stage, labeled as 'spatio-temporal bandpass filters' in the figure, performs filtering operations on the inputs at different spatial and temporal locations. Typically, a single, circular spatio-temporal filter is applied. At the third stage, filtered outputs are convolved with eight or more spatially oriented filters to get the oriented responses. In some designs, the second and third stages may be implemented by a single set of filters. Outputs of the third stage consists of channel responses of different frequencies and orientations at distance-adjusted spatial and temporal locations. The fourth stage, labeled as 'spatio-temporal masking', computes the strength of visible noise in the presence of background signals. Masking functions and their parameters applied in prior work differ significantly. The simplest uses contrast sensitivity only and more sophisticated models take into account phase coherence and the learning effect

[20]. The final stage pools all outputs and generates one (scalar) or more (vector) parameters to characterize the differences between the two input video in designer-defined metric.

In Section 4.3.2, we apply Lambrecht's motion picture quality metric to measure the quality of delayed segmented video. Strictly speaking, his and the aforementioned objective measures are *fidelity* metrics rather than *quality* metrics. The basic system architecture in Figure 4.1 takes two inputs and characterizes how similar they are. In most applications, viewers do not have the reference input to judge picture quality. As a simple example, the quality judgement of an image of a tree does not require the physical presence of the tree, partly because we know what a tree should look like based on our experience. An ideal quality metric shall thus take just one input. Unfortunately, this involves the aforementioned central level processing which is less understood. Another good example involving central processing is eye tracking. HVS studies found contrast sensitivity of a moving object depends on whether the viewer is following the object or not. Should eyes be tracking the object, temporal frequency and temporal masking might have little effect. On the other hand, if eyes were not tracking the motion, temporal masking could be very significant. Eye tracking is not incorporated in most models for it also involves central level processing. Recognizing that there are impairments which are not easily measured yet are obvious to a human observer, we believe subjective testing is still necessary.

Formal subjective testing has been used for many years and several commonly used methods are described in CCIR Recommendation 500 'Methodology for the subjective assessment of the quality of television picture.' In brief, a number of experts and non-expert observers are selected, tested for their visual capabilities, shown a series of test scenes in a controlled environment, and asked to score the quality of the scenes in one of a variety of manners. Although the results of subjective testing reflect true video quality, it is seldom used for the following reasons:

- A wide variety of possible methods and test element parameters must be considered;
- Many observers must be selected and screened;
- Meticulous setup and control are required;
- The complexity makes it very time consuming and costly.

In the following, we briefly summarize several commonly used measurement methods, the first two of which are applied in our experiments.

1. **Stimulus Comparison Method:** scene pairs are shown together for comparison. The differences may be scored using a numerical rating scale or simple ranking in adjective terms: better, the same, and worse. This method is used in our quality ranking experiment.
2. **Single Stimulus Continuous Quality Evaluation:** a single sequence is shown and the subject evaluates its quality in a continuous scale. Data is taken every few seconds to estimate the time varying quality of the image sequence. This method is used in our quality rating experiment.
3. **Double Stimulus Impairment Scale:** observers are shown multiple reference scene, degraded scene pairs. The reference scene is always first. Scoring is on an overall impression scale of impairment, with five being imperceptible and one being very annoying.
4. **Double Stimulus Continuous Quality Scale:** it is similar to the Double Stimulus Impairment Scale method but a continuous quality scale is rated in reference to the other scene of the pair.

Procedures used in our subjective testing experiments as well as the generation of test video sequences are detailed in Section 4.6: Appendix A.

## 4.2 Video Fidelity Characterization

In these experiments, the objective is to determine if sequences with non-zero delay offset between the two flows can be visually discriminated from the original, jitter free video rendering. The video sequences used in the experiments were the luminance components of standard H.263 test clips: Suzie, Salesman, and Mother-daughter (see “Appendix A: Subjective Evaluation Procedure” on page 88 for details). Both compressed and uncompressed sequences were used. A standard self-paced psychophysical method of constant stimuli was used with three delay offsets (0, 1, and 2 frames) for the high delay flow to determine the discriminator threshold for delay offset. Each run consisted of 100 to 150 trials with correct response feedback provided after each trial.

We found judgements of video fidelity to be unanimous: for both uncompressed and compressed video sequences, artifacts from asynchronous rendering are perceptible at the shortest delay. This is not surprising since aggressive segmentation ought to create discriminable visual differences, as we shall elaborate in Section 4.4. While conducting the experiments, we also observed a learning effect. After a subject watched the same sequence 20 to 30 times, they learned to focus on specific details in the sequence for making the discrimination and ignored the rest of the image. A videoconferencing participant is not likely to watch the video more than once so this type of learning is unlikely to be a factor in practical application of DCVC. Therefore, we are less concerned about fidelity than perceived quality.

Another interesting observation was that while delay offset was perceptible, the video quality of compressed sequences did not necessarily degrade. In fact, for some compressed sequences and most observers, the quality appeared to improve with relatively long delays. This effect appears to be related to a reduction of mosquito noise<sup>1</sup> when delay is introduced. This observation prompted us to examine video quality in more detail.

## 4.3 Video Quality Characterization

Unlike the fidelity experiments, this set of experiments focused on evaluating the quality of compressed video. In particular, we are interested in the compression-introduced masking effect on delay segmentation artifacts. From the fidelity experiments, we know that nonzero delay offset and its accompanied asynchronous reconstruction introduce perceptual differences in uncompressed sequences. It is also known that lossy compression generates quantization noise. Furthermore, we observed that the noise contributed by compression seems to be stronger and dominates the perception of the overall video quality. Our goal in these experiments is to characterize the effect of delay on compressed video quality. The procedure of the experiments and the preparation of test sequences are described in Section 4.6.

### 4.3.1 Subjective Quality Evaluation

Subjective testing consists of two parts: ranking and rating. Ranking experiments applied the Stimulus Comparison Method while rating experiments applied the Single Stimulus Continuous Quality Evaluation method.

**Video quality ranking:** The results from ranking the four simultaneously presented stimuli, steps 1 and 3 from Section 4.6, are summarized in Table 4.1. The numbers in the table represent the frequencies of stimuli being ranked as the best, 2nd best, 3rd best and worst video quality when the four stimulus conditions are compared to each other. The table notations for the four stimulus conditions; high compression with no delay, high compression with twelve frame delay, low compression with no delay and low compression with twelve frame delay, are  $H_0$ ,  $H_{12}$ ,  $L_0$  and  $L_{12}$ , respectively. The table frequencies represent the aggregated rankings

---

1. In compressed video, distortion sometimes seen around the edges of moving objects, and characterized by moving artifacts around edges and/or by blotchy noise patterns superimposed over the objects, resembling a mosquito flying around a person's head and shoulders. (Quoted from Federal Standard 1037C, Glossary of Telecommunication Terms, 1996)

given by the eleven subjects for the seven video sequences. Because every subject ranked each sequence four times in an experiment, the total number of data points per condition is 308.

Table 4.1 Aggregated ranking results from 11 subjects and their choices on 7 test sequences

Votes	L <sub>0</sub>	L <sub>12</sub>	H <sub>0</sub>	H <sub>12</sub>
Best	125	<u>164</u>	4	15
2nd Best	<u>134</u>	106	25	43
3rd Best	39	32	106	<u>131</u>
Worst	10	6	<u>173</u>	119

Larger numbers in this table are distributed in the upper-left and lower-right quadrants. As expected, the difference in compression level between first two columns and the second two columns had a significant impact on video quality. The impact of delaying part of the video stream on video quality is more subtle but nevertheless significant. The difference between the high compression conditions, with and without delay (H<sub>0</sub> & H<sub>12</sub>) was significant ( $\chi^2 = 23.8; p < 0.01$ ). Similarly, the difference between the low compression conditions, with and without delay (L<sub>0</sub> & L<sub>12</sub>) was also significant ( $\chi^2 = 10.2; p < 0.05$ ).

Consider the high compression conditions, the response distribution in the H<sub>12</sub> condition column is shifted up relative to the distribution in the H<sub>0</sub> column which indicates the delayed video was *favored* over synchronous video. Similarly, for the video sequences with less compression, the response distribution in the L<sub>12</sub> column is shifted up relative to the distribution in the L<sub>0</sub> column which indicates the delayed video had higher quality than the normal or non-delayed video. This result is most surprising, delaying part of the video stream *improved* video quality for compressed H.263 video sequences. DCVC can improve network performance and improve video quality at the same time, a finding that has important implications for future low bandwidth video coding.

**Video Quality Ratings:** The video quality ranking results were unexpected. If delay segmentation improves video quality for a fixed perceptual delay in side by side comparisons, would it still be observable when sequential video quality assessments are made? Two sets of quality ratings using the same stimuli were gathered about 30 minutes apart for each subject. Test-retest analysis [79] of the two data sets indicates they can be safely combined into one data set.

Since video content might have impact on video quality for DCVC sequences, we evaluated the effect of DCVC separately for each sequence. A three-way repeated measures analysis of variance (treatments-by-treatments-by-subjects design) was performed on each of the seven video sequences. (See “Appendix B: Three-way Repeated Measures Analysis of Variance” on page 91 for details) The results for each video sequence are shown in Table 4.2. The first three columns contain the F-ratios for the main effect factors: compression, delay, and subject respectively. Subject is incorporated in the analysis because observers applied different ranges of feedback scores. Some subjects like to use one to five while others like to use one to ten. The differences create subject dependent means and variances, which are accounted for in our analysis.

Factors that had a significant effect on video quality are indicated by an asterisk. The last four columns contain the mean video quality rating for the four conditions. For all video sequences, the compression level has a significant effect on video quality. However, the delay factor was significant ( $p < 0.05$ ) for two sequences, the Carphone and Salesman sequences. For the Carphone sequence, eleven-frame DCVC delay improved image quality. However, for the Salesman sequence, the same delay degraded the image quality. For the rest of the sequences, delay had no significant effect. In general, long delay offset has limited influence on perceived video quality, either positively or negatively. Lastly, the subject factor

is always significant as we expected since different observers applied different rating ranges.

Table 4.2 Results from three-way effects analysis of variance on the ratings aggregated across 11 subjects (\* marks  $p < 0.05$ )

Sequence	F <sub>rate</sub>	F <sub>delay</sub>	F <sub>subject</sub>	L <sub>0</sub>	L <sub>12</sub>	H <sub>0</sub>	H <sub>12</sub>
Carphone	205.60*	4.38*	8.25*	4.60	4.93	2.48	2.77
Foreman	309.52*	2.61	10.04*	5.55	5.11	2.92	2.91
Salesman	76.99*	3.96*	4.17*	5.00	4.52	3.48	3.34
Suzie	249.17*	0.36	15.19*	5.10	5.00	2.85	2.78
Mother	130.58*	0.71	13.68*	4.40	4.31	2.41	2.76
Claire	190.86*	0.24	12.51*	5.15	5.23	3.24	3.30
Miss Am	87.32*	1.90	11.27*	4.33	4.61	3.07	3.18

Results from a quality ranking evaluation using simultaneous presentation of all four conditions indicated that delayed video looked the same or better than traditional, synchronous video. When observers were asked to make quality ratings for the same sequences presented one at a time, the improvement with delay disappeared for all but one video sequence. Video quality was generally not effected by the large twelve frame delay. The lack of having a direct comparison stimulus and having to rely on memory probably accounts for the disappearance of the improvement with delay effect in the sequential testing conditions.

How can delay improve video quality, even by a small amount? Figure 4.2 schematizes an example where delay should improve video quality by reducing dynamic noise. The figure illustrates the condition when an original, uncompressed block (8 by 8 pixels) is varying slowly in time. Under high compression, the compressed block contains quantization noise. Upon rendering the video sequence, the block closely follows the luminance variation of the original block. However, the quantization noise changes from frame to frame as shown in the second row of Figure 4.2. This noise is often referred to as mosquito noise and is very annoying. For delayed video, however, the encoder sends the second to the fourth block to the high delay flow, which will arrive at the receiver after 400 msec. In the mean time, the decoder simply keeps showing the first block received as shown in the third row of Figure 4.2. The quantization noise seen by our subjects is thus

static. The static noise seems to be preferred to the dynamic noise. The static noise might even be attributed to the original image but dynamic noise is clearly not a part of the original scene. Presumably, this observation can be applied to improve MPEG and other conventional compression algorithms, because skipping the blocks that cause dynamic noise improves quality and reduces bit rate at the same time. We explore this surprising finding further in Section 4.4.

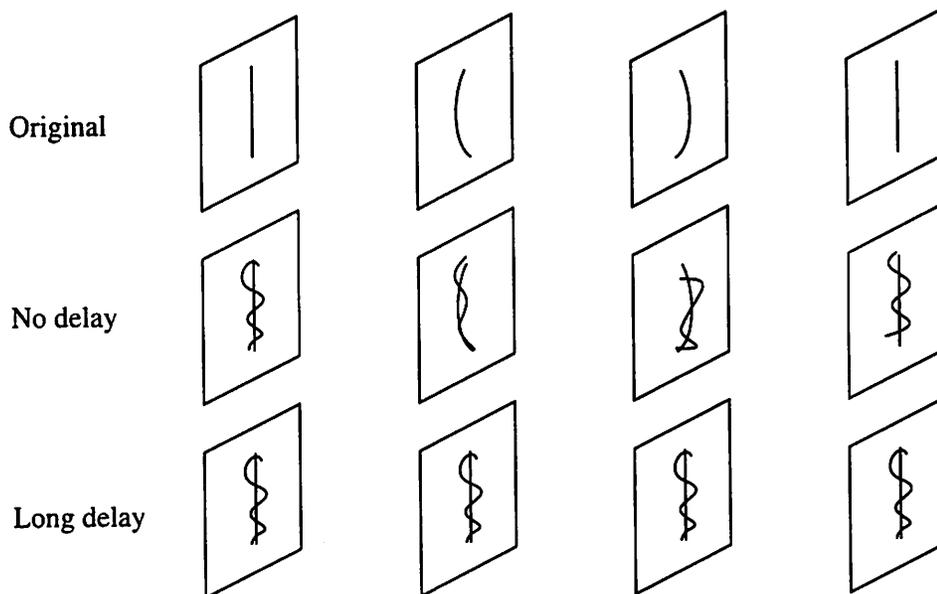


Figure 4.2 A qualitative illustration of the condition when delayed video looks better.

### 4.3.2 Computational modeling methods

We employed the popular PSNR (Peak Signal to Noise Ratio<sup>1</sup>) measure as well as a computational vision model developed by Lambrecht [42][43] to quantify video quality. PSNR is the commonly used metric valued for its simplicity and universal mean squared error formulation. The Lambrecht model, named Moving Pictures Quality Metric (MPQM), applies the basic architecture illustrated in Figure 4.1. The model includes multi-scaled arrays of Gabor shaped spatial filters

---

1. PSNR is calculated by taking the ratio of peak signal energy,  $255^2$ , and noise signal energy,  $(\text{uncompressed} - \text{compressed})^2$ .

at several orientations, intra-channel masking and a Minkowski summation stage. The model also includes an extension to the time domain by adding sustained and transient temporal filters to evaluate video sequences. Video inputs to both metrics were adjusted to reflect the CRT luminance nonlinearity (gamma function) to approximate the luminance actually presented on the display screen.

Table 4.3 Peak Signal-to-Noise Ratio per frame of the Mother-daughter sequence

PSNR(dB)	$L_0$	$L_{12}$	$H_0$	$H_{12}$
Average	32.54	32.45	31.72	31.66
Minimum	32.17	32.08	31.29	31.30
Maximum	33.03	32.93	32.20	32.19

Table 4.3 listed the average, minimum and maximum of the per-frame PSNR of the four stimulus conditions in the Mother-daughter sequence. Similar results apply to other sequences. The PSNR measure predicted nonzero delay off-set sequences had a lower quality by a relatively small amount of difference. The difference is much bigger when PSNR is calculated locally based on blocks, which probably reflects human perceived quality better. The maximum differential PSNR PSNR calculated on blocks is often greater than the mean differential PSNR across all blocks. Viewing experiences show severe local quality degradation has stronger impact than mild, universal quality degradation. The per-block PSNR difference can be as much as 1.46 dB as shown Table 4.4, which again predicts the quality degrades in asynchronous reconstruction. As a reference, Table 4.5 lists the per-block PSNR differences of  $H_0$  and  $L_0$ , which have different compression levels.

Table 4.4 Maximum PSNR difference (DPSNR) between  $H_0$  and  $H_{12}$  calculated per-block

Sequence	Mother-daught	Miss America	Suzie	Foreman
DPSNR(dB)	1.46	1.16	0.7	1.04

Table 4.5 Maximum PSNR difference between  $L_0$  and  $H_0$  calculated per-block

Sequence	Mother-daught	Miss America	Suzie	Foreman
DPSNR(dB)	2.12	2.62	2.97	2.96

The outputs of MPQM are quantified in just-noticeable-distortion (JND). Table 4.6 listed the average, minimum and maximum of the JND values using MPQM metric units for the Mother-daughter sequence. Like the PSNR metric, MPQM also did not predict that delayed video would look better. The results showed that the effect of delay contributed approximately 30 percent of degradation relative to the change of compression levels.

Table 4.6 MPQM outputs of the Mother-daughter sequence; the higher the noise, the lower the quality

JND	$L_0$	$L_{12}$	$H_0$	$H_{12}$
Average	0.207	0.216	0.236	0.245
Minimum	0.187	0.194	0.210	0.218
Maximum	0.230	0.240	0.262	0.269

Although we hoped to replace time-consuming psychophysical experiments with computational metrics, the above results suggested this is not accurate enough for the types of artifacts introduced with delay segmentation. Neither the commonly used PSNR nor the HVS based MPQM adequately captured the effect of differential delay. Further enhancement of the HVS modeling methods should be addressed before they are considered as suitable replacements for subjective evaluations.

#### 4.4 Dynamic Noise Incurred Quality Degradation

Although the conclusion of subjective quality tests indicated delay segmented video sometimes looked better, we took this surprising finding cautiously. The main focus of DCVC is not to develop a better compression algorithm but to explore the delay aspect of video. Ideally, delaying any part of an ‘optimally’ compressed video will cause quality degradation, because an optimal compressor should have kept only visually significant information. Any additional changes, either spatially or temporally, should cause a noticeable degradation. The compression we used was not optimal in the visual quality sense and thus our findings suggest there might be ways to improve quality and save bits at the same time. After the compression algorithm is modified to take into account this finding, we expect

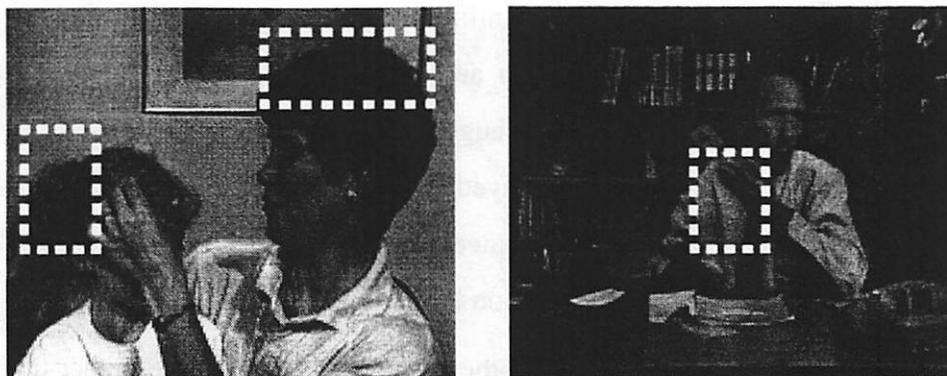


Figure 4.3 Image regions that incur strong flickering noise under heavy compression are marked by dotted boxes.

delay segmented video to always be somewhat degraded when it is compared with synchronous video.

Our goal is to isolate the causes for DCVC based quality augmentation to improve the compression algorithm. As illustrated in Figure 4.2, small variations ‘modulate’ accompanied quantization noise, which often causes flicker. Dynamic, flickering noise attracts human viewers’ attention and leads to lower quality ranking. On the other hand, if the compressor ignores those small variations, quantization noise becomes static and is often overlooked if it does not occur in key areas. After studying the test sequences, we found attention-attracting flickering occurred in mostly static, medium to high textured areas. Examples are marked in Figure 4.3.

As clearly described by Klein [39], natural images are much harder to analyze than simple lines, edges and gratings used in basic vision research. Natural scenes in video are even harder to analyze because of the temporal dimension. Marked areas in Figure 4.3 do not always flicker and localizing its occurrence both spatially and temporally is rather difficult. Our approach is to design artificial stimuli which mimic these scenes in creating dynamic noise. Simple artificial stimuli are easier to study in order to develop new guidelines for compression. We recognize, however, flickering noise does depend on video content. For example, flicker-

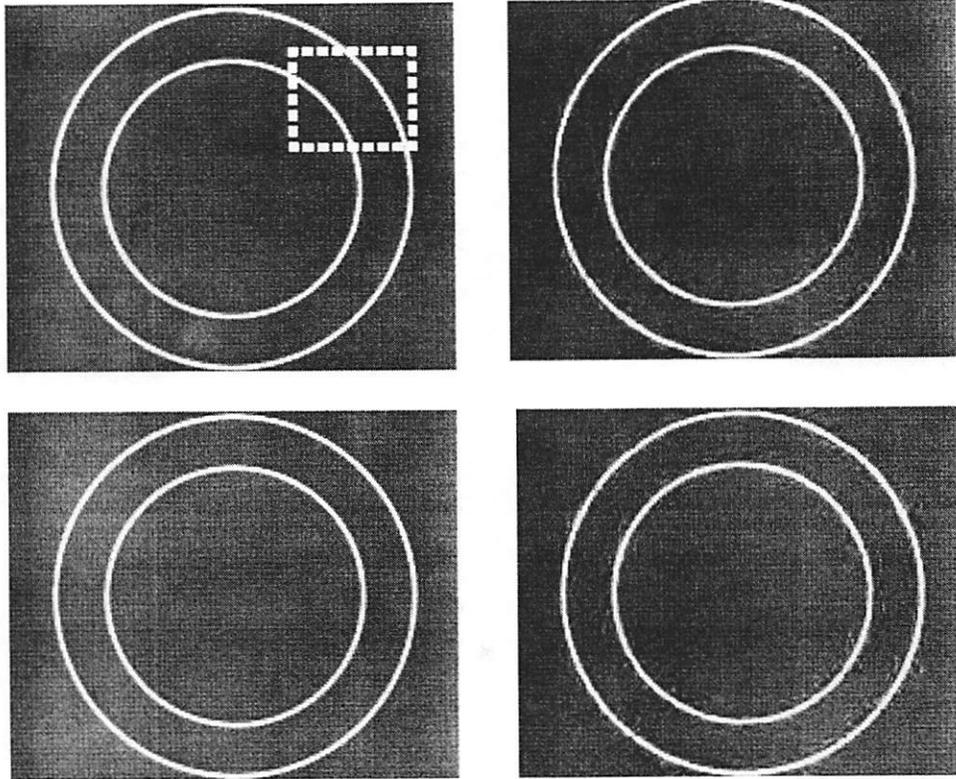


Figure 4.4 Uncompressed (left) and compressed (right) video frames of the double-ring pulsing circle stimulus; the dotted box is magnified in 3D shown in Figure 4.5.

ing of densely leaved trees are hard to identify as artifacts. A weakness of artificial stimuli is that they do not take into account context dependency.

Besides the amount of quantization noise introduced, periodicity also has strong impact on perceived flickering. Take the pattern of a shifting, periodic grid for example. Flickering noise is stronger if the size (area) of the grid is smaller. Flickering is less noticeable if the grid is bigger. In this case, spatial periodicity seems to reduce perceived flickering. It is empirically observed that temporal periodicity also reduces flickering. A periodic temporal movement seems to cause a learning effect on observers. And once noise activities are anticipated, they no longer cause strong perception of flicker. We found for the same set of compressed video frames, reordering to make them aperiodic causes stronger perceived flickering.

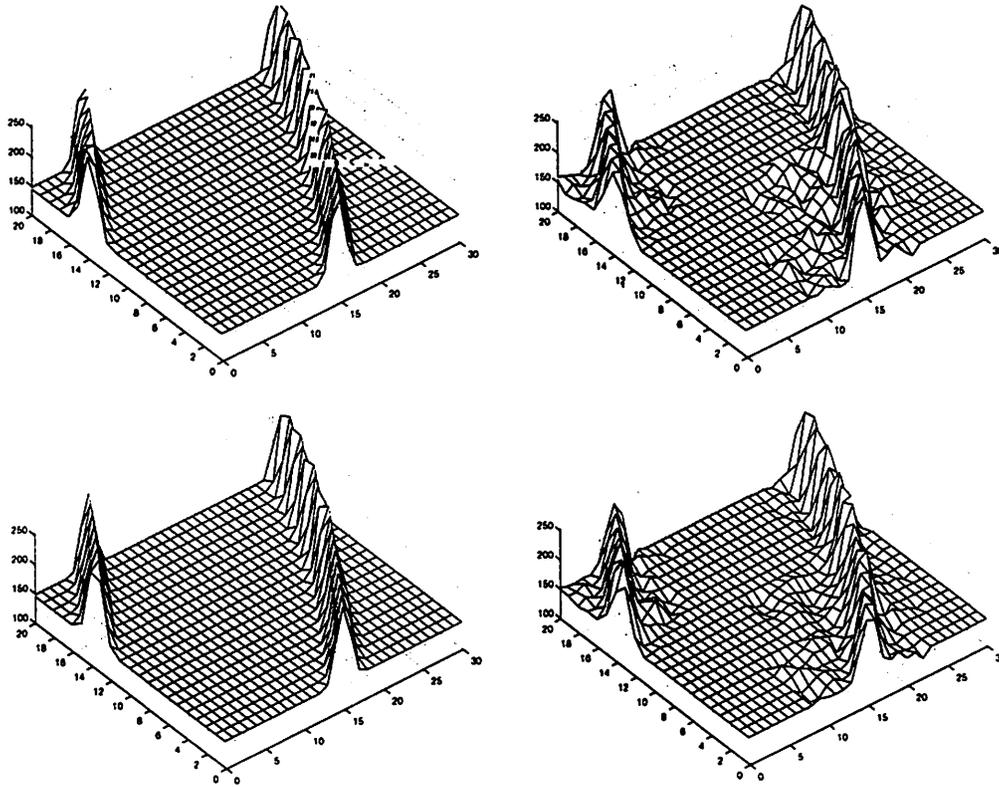


Figure 4.5 A magnified view of the dotted box in Figure 4.4; temporal variation of quantization noise causes flickers.

One artificial stimulus we constructed was a double-ring pulsing circle (change in diameter). While it is difficult to show flickering, which involves temporal changes, on paper, two consecutive frames of the uncompressed and compressed images are plotted in Figure 4.4 for reference. Pseudo codes for generating the stimulus is reported in Section 4.8: Appendix C. The two rings are almost perceptually static on our CRT display, while in fact the rings move synchronously at a moving distance less than half a pixel. Under heavy compression, the small movement is visually amplified by quantization noise around the rings. When the compressed stimulus is played back, flickering artifacts are very noticeable around both rings. Figure 4.5 plots a small segment of the rings in three dimensions, where z-axis represents pixel intensity. The change of pixel values in neighboring areas is significant and annoying.

The above stimulus is one example which looks static in uncompressed video but draws viewers attention after compression because of flickering. The strength of the flickering noise of this stimulus also depends on quantization levels and temporal periodicity. A more complete characterization of several other stimuli can be found in [11].

## **4.5 Summary**

In this chapter, we answered an important question about delay cognizant video coding - how does it impact video quality? The formal subjective testing indicated delay segmented video even looked better than synchronous video in certain occasions under both low and high levels of compression. We identified some of the natural scenes in which delayed video looked better and constructed artificial stimuli to mimic these scenes. We hope these simpler stimuli will help video coding research to develop better coding control rules and improve compression algorithms. Since our findings will only affect the compression end, decompression algorithms and bit stream formats need not change. Output streams of the improved compressor will be compatible with existing standards.

Up to this chapter, we have described the DCVC algorithm, its applications and quality evaluation. The themes are circled around the service and application layers. DCVC assumes its delay flows can be carried by a differential QoS network. In the next chapter, the theme is shifted down to the bitway layer and we examine how to implement differential QoS in wireless networks.

## **4.6 Appendix A: Subjective Evaluation Procedure**

Eleven paid volunteer subjects from the UC Berkeley campus participated in the experiments in January, 1998.

The seven raw video sequences used in the experiments are standard H.263 test clips: Carphone, Claire, Foreman, Miss America, Mother-daughter, Salesman, and Suzie. The test sequences are available from ITU and its members. They are

stored in the 4:2:0 QCIF format (176 by 144 pixels). For both the fidelity and quality experiments, only the luminance component of the video was used. Each sequence was 2.5 seconds long (75 frames) and was presented on a Sony Trinitron monitor at 60 Hz (two scans per frame). MATLAB with the PC-MatVis [10] psychophysical testing and stimulus presentation extension were used to present the stimuli and gather the rating data. Among the encoded sequences, the number of low-delay blocks was between 10 to 20 percent of the total. The actual percentage is content dependent. For nonzero delay offset sequences, we applied the same amount of delay uniformly, in the units of frame display time (1/30th of a second), to the video data in the high-delay flow.

The test sequences were generated with two independent variables, delay and compression. Each variable had two levels for a total of four stimulus conditions per sequence. We investigated not only the effects of delay offset but also the effects of compression-introduced masking.

- **Compression level:** The first frame (the only I frame) of all four stimulus conditions was compressed at the same quantization level and thus contained identical information. The amount of compression-introduced noise in subsequent frames is controlled by the quantization level (QL). The quantization step size in inter-coding is twice of the quantization level. All 64 DCT coefficients of

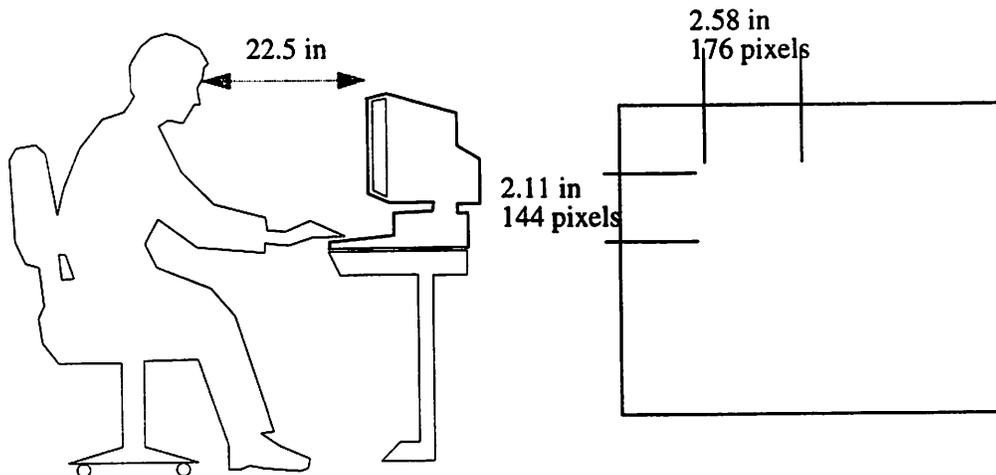


Figure 4.6 Experiment settings; the right shows four test images (176 by 144) displayed on the screen. The pixel size is 2.24 min wide.

inter-coded blocks are quantized with the same level. Increasing the quantization level decreases the video quality and vice versa. In stimulus conditions 1 and 2, QL was set to 10 for all seven sequences. In stimulus conditions 3 and 4, QL was set to 12 to compress Salesman, Mother-daughter, and Miss America while the other four were compressed with QL equal to 13. Depending on the video content, a decrease of QL from level 12 to 10 increases the compressed bit rate by 20 to 50 percent.

- Delay level: Synchronous, zero-delay-offset video reconstruction was applied to conditions 1 and 3. A delay offset of 12 frames (~400 milliseconds) between the low- and high-delay flows was applied to stimulus conditions 2 and 4. Non-zero delay offsets lead to asynchronous video reconstruction.

The procedure of evaluating video sequence quality involves the following three steps. The experiment settings are illustrated in Figure 4.6.

1. Simultaneous Presentation Quality Ranking: All four stimulus conditions (video clips) were presented simultaneously, two across and two down on the screen as shown in Figure 4.6. Stimulus locations were chosen randomly. The 2.5-sec long presentation was repeated ten times (additional viewing time was available as desired by the subject). The subjects were asked to rank order the four stimuli using their own subjective criteria for quality. The quality ranking step applied the Stimulus Comparison Method described in Section 4.1.
2. Successive Presentation Quality Rating: Each of the four stimulus conditions was presented individually in random order for a total of 20 trials, 5 for each condition. Each stimulus presentation lasted 5 seconds (two repeats). After each stimulus presentation, the subject was asked to rate the image quality on a scale of 0 to 9. Subjects were not told that only the four stimulus conditions seen earlier were being presented again. They were told that the four stimuli that appeared in step 1 bracketed the range of quality levels to be presented in this

step of the experiment. The quality rating step applied the Single Stimulus Continuous Quality Evaluation method described in Section 4.1.

3. Repeat: Finally, step 1 above was repeated using the same stimulus conditions. Subjects were not informed that the stimulus conditions in steps 1 and 3 were in the same screen locations.

To evaluate consistency of the subject responses, the three steps above were performed for all seven sequences and then repeated. It took each subject about an hour to finish the experiment.

The most often received comment from our subjects was the difficulty in rating video quality in step 2. With highly compressed sequences, different patterns of noise appeared in different parts of the image and were varying over time. In preliminary studies, when step 2 was performed alone subject rating criteria for video quality appeared to shift over time generating ‘inconsistent’ results. We found that step 1 helped in reducing the inconsistency by presenting four stimuli simultaneously. The longer viewing time gave subjects an opportunity to study the stimuli and establish stable criteria.

## 4.7 Appendix B: Three-way Repeated Measures Analysis of Variance

The three-way fixed effects model involves three experimental factors, each of which is represented by a fixed effect on the observations. The linear model for the three-way fixed effects analysis of variance is given by

$$Y_{ijkl} = \mu + A_j + B_k + C_l + D_{jk} + E_{kl} + F_{jl} + e_{ijkl} \quad (\text{Eq 4.1})$$

where  $Y_{ijkl}$  is the  $i$ th observation in the  $j$ th level of the first factor, the  $k$ th level of the second factor, and the  $l$ th level of the third factor;  $\mu$  is the grand mean;  $A_j$  is the treatment effect associated with the first factor;  $B_k$  is the treatment effect associated with the second factor;  $C_l$  is the treatment effect associated with the third

factor;  $D_{jk}$  is the treatment effect associated with the interaction effect of the first and second factors;  $E_{kl}$  is the treatment effect associated with the interaction effect of the second and third factors;  $F_{jl}$  is the treatment effect associated with the interaction effect of the first and third factors;  $e_{ijkl}$  is the random variable associated with error.

We assume  $e_{ijkl}$  is a Gaussian random variable and the errors in any observations are independent. The null hypotheses of the main effects (or the first order effects) for this model are given by:

$H_0: \forall j, A_j = 0$ , the first factor has no effects on the observations.

$H_0: \forall k, B_k = 0$ , the second factor has no effects on the observations.

$H_0: \forall l, C_l = 0$ , the third factor has no effects on the observations.

Define  $MSE$ ,  $MSA$ ,  $MSB$ , and  $MSC$  as follows.

$$MSE = \frac{\sum_{j=1}^J \sum_{k=1}^K \sum_{l=1}^L \sum_{i=1}^I (Y_{ijkl} - \text{mean}_{jkl})^2}{JKL(I-1)} \quad (\text{Eq 4.2})$$

$$MSA = \frac{\sum_{j=1}^J KLI(\text{mean}_j - \text{mean})^2}{J-1} \quad (\text{Eq 4.3})$$

$$MSB = \frac{\sum_{k=1}^K JLI(\text{mean}_k - \text{mean})^2}{K-1} \quad (\text{Eq 4.4})$$

$$MSC = \frac{\sum_{l=1}^L JKI(\text{mean}_l - \text{mean})^2}{L-1} \quad (\text{Eq 4.5})$$

where  $mean_{subscript}$  is the mean value of the samples subject to the subscript. When there is no subscript, it is the mean value of all samples.

The  $F$ -ratios of the three main effects are calculated by the following equations.

$$\text{The fixed effect of A: } F_{J-1, JKL(I-1)} \sim \frac{MSA}{MSE} \quad (\text{Eq 4.6})$$

$$\text{The fixed effect of B: } F_{K-1, JKL(I-1)} \sim \frac{MSB}{MSE} \quad (\text{Eq 4.7})$$

$$\text{The fixed effect of C: } F_{L-1, JKL(I-1)} \sim \frac{MSC}{MSE} \quad (\text{Eq 4.8})$$

The  $F$  distribution can be found in lookup tables of many Statistics textbooks or can be computed using an incomplete Beta function.

A null hypothesis is rejected for a given tail probability  $p$  if its  $F$ -ratio is greater than the lookup value of the  $F$  distribution. In the analysis applied in this chapter,  $p$  is set to 0.05, which means when the hypothesis is rejected, the probability of the hypothesis being true is less than 0.05. On the other hand, if the  $F$ -ratio is less than the lookup value of the  $F$  distribution, there is no sufficient statistical support to reject the hypothesis.

In the analysis applied in Section 4.3.1,  $A_j$  represents the main compression factor of two levels;  $B_k$  represents the main delay factor of two levels; and  $C_l$  represents the main subject factor of 11 levels.

## 4.8 Appendix C: Pseudo Codes for Generating the Double-ring Pulsing Circle Stimulus

The following codes generate a sequence of 64 QCIF-sized frames (176 by 144). The standard H.263 compression algorithm is then applied with quantization levels set to 15. The playback rate is 30 frames per second using PC-MatVis [10]. Pixel values are gamma corrected before display to ensure luminance linearity.

```

int pattern[64] = {           // the matrix controls the temporal
    0, 0, 0, 1, 0, 1, 1, 0, // pulsing pattern of the stimulus;
    0, 0, 0, 1, 0, 0, 0, 1, // radius of the circle depends on
    0, 0, 0, 1, 0, 1, 0, 0, // this pattern
    0, 0, 0, 0, 0, 0, 0, 1,
    1, 1, 0, 0, 0, 0, 1, 0,
    0, 0, 0, 0, 0, 0, 0, 1,
    0, 1, 0, 1, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 1, 0};
radius = 70;                // radius of the outer circle
scaling = 200;              // the width of thin circles has a
spread = 0.3;               // Gaussian shape; scaling and spread
                             // determine the shape

PI = 3.1416;

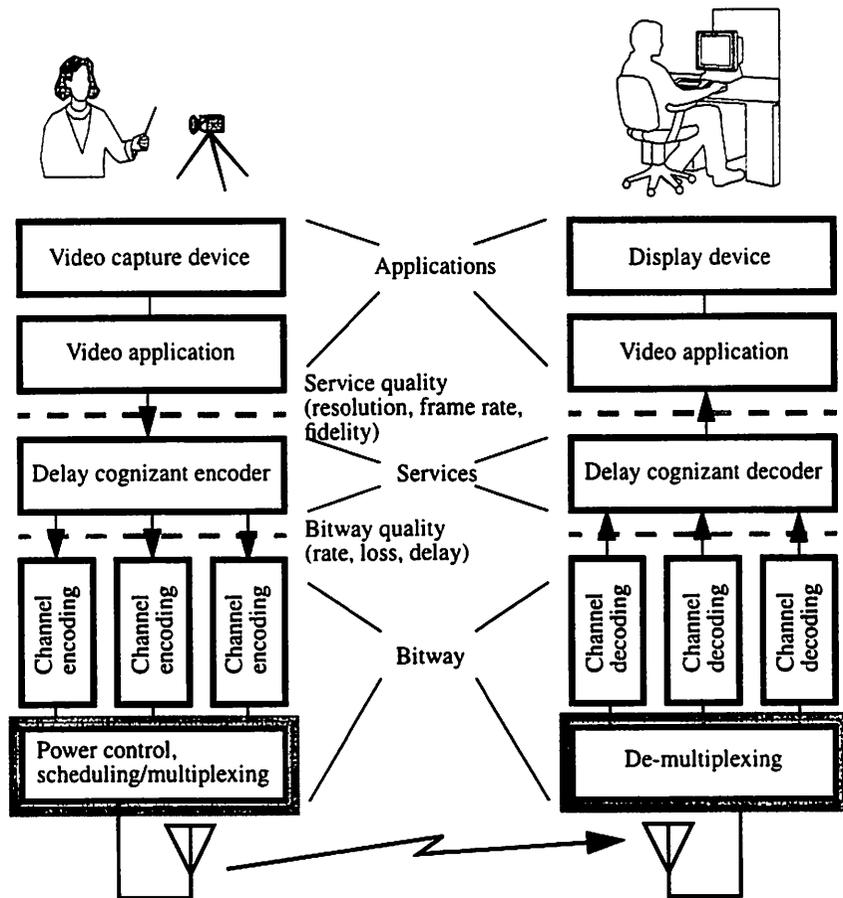
for (frame = 0; frame < 64; frame++)
{
    if (pattern[frame])
    {
        r = radius;         // the variation of radius is 0.2 pixels
    }
    else
    {
        r = radius+0.2;
    }

    for (i = 0; i < 176; i++)
    for (j = 0; j < 144; j++)
    {
        {
            dist1 = sqrt((i-88)^2+(j-72)^2)-r;
            dist2 = sqrt((i-88)^2+(j-72)^2)-r+20;
            pixelvalue[i][j] = scaling*exp(-dist1/2/spread)/
sqrt(2*PI*spread) + scaling*exp(-dist2/2/spread)/sqrt(2*PI*spread);
        }
    }
    write pixel values
}

```

# 5

## Power Control and Scheduling on CDMA Wireless



In the previous three chapters, we introduced the delay cognizant video coding algorithm and the two layers above, its applications and quality evaluation. One premise of DCVC is a networking infrastructure that supports differential delay flows, which hold promises to improve traffic efficiency. Since wireless network, with scarce bandwidth and hostile channel environment, is likely to benefit from the improvement most, we chose wireless Code Division Multiple Access

(CDMA) networks as the subject of study at the bitway layer. CDMA is a spread spectrum technique which spreads information or coded symbols onto a much larger bandwidth via modulation by a *signature* or *spreading sequence* [27][57][73][74]. Code-spread information from different users can coexist in the same channel with limited interference to each other. The interference increases as the number of users increases. Besides mostly known advantages such as resilience to multipath fading, from the link layer perspective, CDMA is well suited to support statistical multiplexing (SM) and thus provides a good architecture for bit rate scalability. CDMA is suited for SM for its graceful link quality degradation when concurrent transmissions occur. Users see an increase of bit error rate (BER), which may not necessarily lead to packet loss. In contrast, another commonly used technique, Time Division Multiple Access (TDMA), requires explicit scheduling to avoid packet collisions, which is an all or none situation. The SM compatible feature makes CDMA a favorable choice to provide multimedia services.

The focus of the chapter is to examine issues of using power control and scheduling techniques in CDMA to control QoS received by each flow. Adjusting the transmitting (or receiving) signal power of a mobile relative to other users' signal provides both a solution to the near-far problem and an effective approach to control link reliability reflected by BER. The idea of using power control to provide differential reliability to flows was proposed in [76]-[78]. The key results developed in those papers will be quoted and used in this chapter. Our emphasis is the step before power control, the scheduling of packet transmissions. While explicit scheduling is not required for CDMA as demonstrated in digital cellular standard IS-95, it is required for provisioning QoS guarantees. The scheduling algorithm selects packets to be transmitted to ensure that both reliability and delay guarantees are not violated. As we will show, the joint power control and scheduling problem is a computationally hard problem with no polynomial time solutions scaled to the number of users.

We first give a brief review on the development of cellular mobile service, which is into the third generation using wideband CDMA technology. The system model is then introduced and used as a reference in the following discussions. We then describe the scheduling problem of CDMA networks with fixed-rate links, which correspond to the second (the current) generation cellular. Recognizing bursty multimedia data cannot be efficiently carried by fixed-rate links, cellular standard committees initiated the development of multi-rate services, which will be part of the third generation cellular. The scheduling and network capacity issues of multi-rate links are then discussed under various power and reliability constraints.

## **5.1 Background**

The first generation of cellular mobile phone service started in the early 1980's by using analog frequency modulation (FM) technology. (A historical review of cellular can be found in [71].) Systems like Nordic Mobile Telephony System (NTMS) in Scandinavia and Advanced Mobile Phone System (AMPS) in the US used 25 to 30 KHz per channel. The service area is often divided into a large number of cells with a segment of frequency spectrum allocated to each. Neighboring cells get different frequency segments and cells using the same segment are spatially separated to minimize interference between users of the same channel. Before long, the strong market demand exposed the inefficiency and limitations of these analog systems and a digital cellular phone solution was sought.

The second generation cellular was developed primarily to address the capacity limitation in dense subscriber areas. An AMPS-enhanced TDMA standard IS-54 (later IS-136) tripled the capacity with the same bandwidth. Separately Global Mobile System (GSM), also a TDMA standard, was deployed in Europe and later to many other countries. Some GSM manufacturers claimed the new standard increases the capacity by seven times. In Japan, Personal Digital Cellular (PDC) was developed and is the current market dominator. In 1992, Telecommunications Industry Association (TIA) in the US approved a second digital standard

IS-95 developed by Qualcomm. Unlike previously mentioned standards, IS-95 applies direct sequence (DS) CDMA technology, which was initially claimed to increase the analog AMPS capacity by 20 times. Although later market deployments proved its capacity gain to be over-estimated, the advantages of CDMA were gradually accepted and recognized. It formed the basis of the third generation cellular. Today an estimated 100 million subscribers worldwide signed up digital cellular service.

While the second generation cellular successfully increased the system capacity by employing digital compression, modulation, equalization and error correction, the data rate of each channel remains below 20 Kbps, which is just enough for voice communications. The fixed low-speed data rate, however, will not satisfy the increasing demand of wireless Internet access and multimedia applications. These new applications transmit bursty traffic and in the case of Web access, service quality is judged by the response time. Therefore, even in the long term, the same amount of data is downloaded through the link. A link capable of accommodating high peak rates with short downloading time is preferred than a slow, fixed-rate link with long download. Without increasing the total bandwidth, flexible bit rate allocation among users thus is crucial to the provision of multimedia wireless.

Recognizing the importance of flexible bit rate allocation, the in-progress third generation cellular standards lists multi-rate services as one primary design objective. Air interface proposals are being finalized for the project International Mobile Telecommunications in the year 2000 (IMT-2000), coordinated by International Telecommunications Union (ITU) [55]. IMT-2000 targets at offering data rates at least 144 kbps for high-mobility users with wide area coverage and 2 Mbps for low-mobility users with local coverage. Several US and Europe/Japan proposals are based on wideband CDMA with some technical differences. It has been agreed upon that the third generation CDMA should have the following new capabilities over IS-95: wider bandwidth and chip rate, provision of multi-rate services,

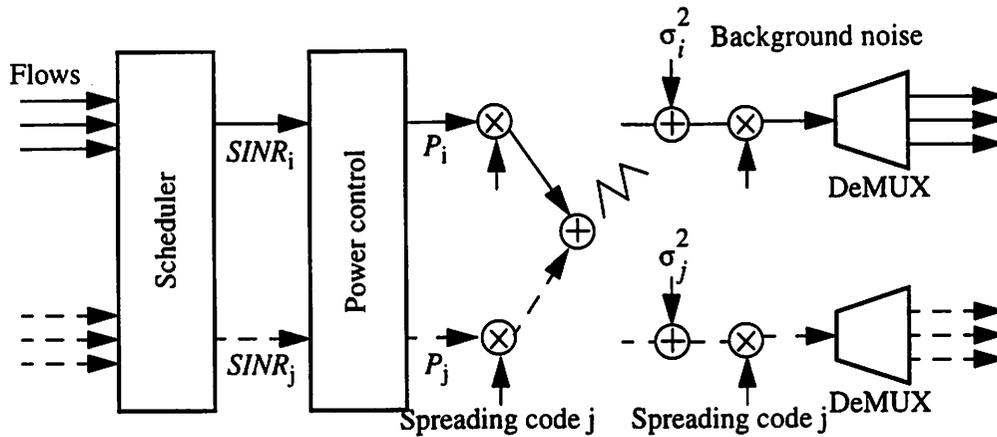


Figure 5.1 The system schematic diagram of the CDMA downlink transmission.

packet data, coherent uplink detection, fast power control in the downlink, and optional multi-user detection.

In this chapter, the joint power control and scheduling problem of CDMA is examined in the context of fixed-rate links, modeled after the second generation cellular, and in the context of multi-rate links, modeled after the third generation cellular. The system schematic diagram of the CDMA downlink is shown in Figure 5.1. We are interested in the downlink because multimedia access is mostly asymmetric with heavy downloads. The problem, nevertheless, can be formulated for the uplink and most conclusions will still apply. In the diagram, flows arriving at the cellular base station are tagged with their delay and reliability requirements. The reliability requirement is translated into the minimum signal to interference and noise ratio (SINR). The scheduler then base on these requirements to pick the flows to be transmitted. Flow packets are then passed to the power control algorithm to compute their transmitting power assignments. Each mobile's data is multiplied by its assigned power and the corresponding spreading codes. At the mobile, the received data stream with added background noise is multiplied with its spreading code to recover the information. Since a user may have more than one flows, a demultiplexer is required to restore the flow structure.

This system diagram applies to both fixed-rate and multi-rate links. In the former, each mobile is assigned a single spreading code with fixed large spreading

gain, which mimics IS-95. In the latter, a mobile may acquire multiple spreading codes with variable spreading gain, which mimics the upcoming wideband CDMA. Rather than formulating the problem with detailed physical layer attributes such as modulation, waveforms, equalization or fading patterns, we chose to study the issues from the link layer perspective. This methodology differs from most prior work. In recent years, a number of proposals on multimedia CDMA wireless networks have been published [30][34][56][85]. While some work focused on a single type of networking configurations, others tried to address a broader performance issue by comparing and contrasting different designs. However, the approaches taken by most work were often targeted at the evaluation of the whole system consisting access protocols, link layer control, physical layer implementations, and even traffic sources. With a lot of elements influencing the outcome of a performance evaluation, it is difficult to pinpoint the factor that has the most significant impact. Our assessment approach is narrower in scope in the hope of bringing more insights into the problem.

## **5.2 Scheduling Problem of Fixed-Rate Links**

The fixed-rate link design was taken by the second generation cellular for it was sufficient for voice communications. IS-95, the first CDMA standard, specifies a chip rate of 1.2288 Mcps (chips per second) and a data rate of 9.6 Kbps which has enough capacity for a low-bit rate voice codec [74]. IS-95 applies matched filter receivers for their simplicity in implementations. We state the power control constraint for matched filter receivers next.

### **5.2.1 Power Control Feasibility Test**

Conventional matched filter receivers and the associated power control problem have been extensively studied in recent years. The formulation applied in the following analysis can be found in [76][78] and other CDMA textbooks [27][57][73][74]. Assume the spreading codes are pseudonoise (PN) sequences. The signal-to-interference-and-noise ratio (SINR) of the received signal for flow  $i$ ,

expressed as  $\beta_i$ , can be expressed as a function of  $P_i$ , the receiving power of each symbol,  $N$ , the spreading gain, and  $\sigma_i^2$ , the background noise depending on receiver's location. Specifically,

$$\beta_i = \frac{P_i}{\frac{1}{N} \sum_{j \neq i} P_j + \frac{1}{N} \sigma_i^2} \quad (\text{Eq 5.1})$$

The above expression points out a key feature of CDMA: SINR is bounded even when receiving power increases to infinity. In the limiting case, the background noise is relatively small and can be ignored. It is easy to see SINR is upper bounded by the ratio of spreading gain to the number of active interferers. This presents one perspective of the interference-limited nature of CDMA link capacity.

For an arbitrary set of  $\beta_i$ , there may not exist  $P_i$ 's such that (Eq 5.1) is satisfied. To see the feasible condition, rewrite the equation as follows:

$$P_i = \left( \frac{\beta_i}{N + \beta_i} \right) \left( \sum_j P_j + \sigma_i^2 \right) \quad (\text{Eq 5.2})$$

Sum both sides over all  $i$ 's.

$$\sum_j P_j = \left( \sum_j \frac{\beta_j}{N + \beta_j} \right) \left( \sum_j P_j \right) + \sum_j \sigma_j^2 \frac{\beta_j}{N + \beta_j} \quad (\text{Eq 5.3})$$

$$\sum_j P_j = \frac{\sum_j \sigma_j^2 \frac{\beta_j}{N + \beta_j}}{1 - \sum_j \frac{\beta_j}{N + \beta_j}} \quad (\text{Eq 5.4})$$

For the summation of all receiving power to be positive, the denominator of (Eq 5.4) must be positive. The power control feasibility test of matched filter receivers thus can be expressed as an inequality shown in (Eq 5.5).

$$\sum_j \frac{\beta_j}{N + \beta_j} < 1 \quad (\text{Eq 5.5})$$

We define  $\gamma_i$  as the power index of flow  $i$  expressed in (Eq 5.6), which has the reliability requirement expressed in SINR  $\beta_i$ .

$$\gamma_i = \frac{\beta_i}{N + \beta_i} \quad (\text{Eq 5.6})$$

The power control constraint states that in order to satisfy SINR requirements of all active flows, the sum of their power indices must be less than 1. Another perspective of the interference limited capacity is when a flow requests very high SINR, its power index approaches one. In that case, this flow needs to be transmitted alone.

## 5.2.2 Rate Admissible Region

Simultaneous flow transmissions must follow the power control feasibility constraint stated in (Eq 5.5). The queueing model of fixed-rate links thus cannot be abstracted as an MISO (multi-input/single output) queue like an IP network switch shown in Figure 3.1 on page 54. Instead, the MIMO (multi-input/multi-output) model in Figure 5.2 should be used. While flows belonged to the same user still share the same spreading code, flows from different users now compete for the

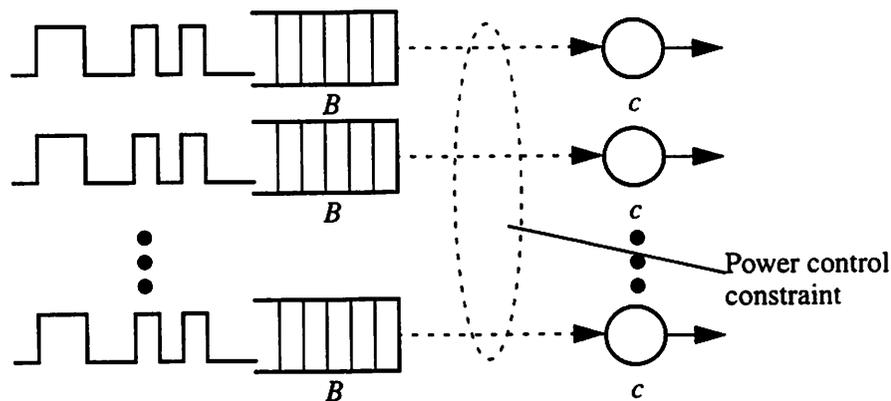


Figure 5.2 The queueing model of fixed-rate links is expressed as MIMO (multi-input/multi-output) governed by the power control feasibility test.

power resource. The function of the scheduling algorithm is to select the flows to be transmitted while observing the feasibility test.

One way to look at the difference between MIMO and MISO is to compare scheduling constraints. In MISO, the constraint can be expressed as:

$$\sum_j n_j \gamma_j < 1 \quad (\text{Eq 5.7})$$

$\gamma_j$  is defined in (Eq 5.6) and  $n_j$  is an integer representing the number of packets from flow  $j$  being transmitted. The system can transmit more than one packet from the same flow. In MIMO,  $n_j$  is set to one and (Eq 5.7) is reduced to (Eq 5.5). This reflects the fact that at each scheduling slot, at most one packet can be transmitted from each flow even if the system still has spare capacity.

The MIMO model has a significant impact on optimal scheduling and admission control. An immediate change from MISO is that the rate admissible (throughput) region is no longer characterized by a single number, the maximum link speed. Since there are as many links as the number of users (mobiles), the rate admissible region is a multi-dimensional polyhedron. The region's dimension is the same as the number of users, when the flows of a user require the same SINR.

**Example 5.1:** There are three users in a service cell. Each of them has one flow with the same power index  $\gamma = 0.49$ . Given the feasibility test that the sum of power indices must be less than one, at each time slot, only two of them can be served simultaneously. Therefore, the maximal assignment vectors, which represent the combinations of maximal number of active links, are  $\{(1,1,0), (0,1,1), (1,0,1)\}$ . Average arrival rates of links  $(\lambda_1, \lambda_2, \lambda_3)$  constitute a vector residing in the convex hull formed by assignment vectors. Note that the arrival rates are normalized in time with at most one packet transmitted each slot. The three-dimensional rate admissible region is shown in Figure 5.3 with maximal assignment vectors marked. By the definition of convex hull, any arrival rate vectors inside the region

can be decomposed as a linear combination of maximal assignment vectors. Arrivals can be scheduled using a simple time division multiplexing policy by applying the set of maximal assignment vectors, should the decomposition be computed.

### 5.2.3 Scheduling Arbitrary Arrivals

The above rate admissible region creates new scheduling problems which do not exist in MISO queues. First and foremost, a scheduling algorithm ought to be able to stabilize the MIMO system when the average rate vector is inside the region. A queueing system is stable when the total queue length remains bounded for finite rate inputs. In an MISO queue, the system stability problem can be easily verified by ensuring that the sum of the average rates of incoming flows is less than the link speed. This is the result of the *work conservation* law [40]. Any scheduling algorithm that is work conserving stabilizes the queue. In an MIMO system, however, no such relations exist. We give an example next to show an intuitively sound scheduler does not always stabilize the system.

**Example 5.2:** There is a scheduling policy that maximizes the number of concurrent transmissions, i.e. it tries to find an assignment vector with most number of 1's. We follow the setting of Example 5.1. The three maximal assignment vectors are  $\{(1,1,0), (0,1,1), (1,0,1)\}$ . The time sequence of arrival and departure events are

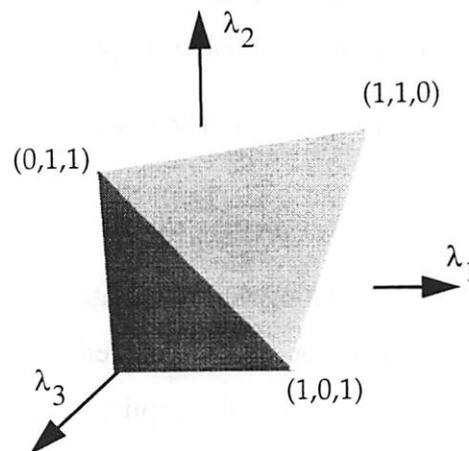


Figure 5.3 The rate admissible region of Example 5.1. The average arrival rate of the  $i$ th user is  $\lambda_i$ .

shown in Table 5.1. The last column shows the number of packets left in the system at the end of the time slot. The elements in the triplets correspond to the user queues. In the first time slot, each queue receives one packet and the first and second queues are served. This leaves one packet in the third queue. In the second slot, the same arrival and service patterns repeat. This leaves two packets in the third queue. There is no arrival in the third slot but the scheduler, constrained by fixed-rate links, can at most serve one packet each queue. Therefore, at the end of the third slot, there is one packet left in the third queue. Repeat the three-slot cycle and at the end of the sixth slot, packets in the third queue grow from one to two. Clearly, this aggressive scheduling algorithm does not stabilize this arrival pattern, although the rate vector is inside the admissible region.

Table 5.1 Time sequence of arrival/departure of Example 5.2.

Time Slot	Arrivals	Departures	Queue Lengths
1	(1,1,1)	(1,1,0)	(0,0,1)
2	(1,1,1)	(1,1,0)	(0,0,2)
3	(0,0,0)	(0,0,1)	<u>(0,0,1)</u>
4	(1,1,1)	(1,1,0)	(0,0,2)
5	(1,1,1)	(1,1,0)	(0,0,3)
6	(0,0,0)	(0,0,1)	<u>(0,0,2)</u>

This simple example demonstrates the challenges of proposing a scheduling algorithm for MIMO queues, especially for arbitrary arrivals. In order for a scheduler to stabilize the queues, it must orient the service vector to the same direction of the arrival vector. In the above example, the average arrival vector is  $\left(\frac{2}{3}, \frac{2}{3}, \frac{2}{3}\right)$  but the service vector is  $\left(\frac{2}{3}, \frac{2}{3}, \frac{1}{3}\right)$ . The system thus destabilizes in the long run.

There is one known scheduling policy to stabilize any arrivals in the admissible region. The algorithm was proposed and proved by Tassiulas and Ephremides [65] for a different problem but it is equally applicable here. The algorithm selects the assignment vector as follows:

$$\max(\vec{Q}^T \cdot \vec{\delta}) \text{ subject to } \vec{\gamma}^T \cdot \vec{\delta} < 1$$

where  $\vec{Q}$  is the queue length vector;  $\vec{\gamma}$  is the power index vector;  $\vec{\delta}$  is the assignment vector. The algorithm finds the assignments that maximizes the total queue length. It is easy to verify that the maximum total queue length policy stabilizes arrivals in Example 5.2.

The problem of finding the optimal assignment vector, however, is NP-complete.

**Theorem 5.1** The computational complexity of the maximum total queue length scheduling algorithm is NP-complete.

Proof of Theorem 5.1: The algorithm has the same formulation as the well-known Knapsack problem. See p. 247 of [25].

Moreover, we suspect the complexity of the admission control problem is also NP-complete. Admission control decides whether the rate vector is inside the admissible region. The suspicion arises because we found the admission control problem of multi-rate links is NP-complete. See Section 5.7 Appendix A for the proof.

#### 5.2.4 Scheduling i.i.d Bernoulli Arrivals

While the task of scheduling arbitrary arrivals is important for non-real-time data services, the provision of delay guaranteed services requires the knowledge of statistical behavior of flow traffic. We investigated the scheduling problem of i.i.d. (independent and identically distributed) Bernoulli arrivals, assuming arrivals between flows are also independent. We assume that all flows have identical reliability requirements. In this constrained case, we are able to prove that the longest queue first (LQF) scheduler minimizes the total number of waiting packets. The LQF policy is the degenerated form of the NP-complete maximum total queue length policy. With the same reliability requirements, which implies the power

indices are equal, the maximum length scheduler selects flows starting with those having the longest queue lengths. The optimality proof is detailed in Section 5.8 Appendix B. The computational complexity of the LQF scheduler is in the order of  $O(N \log N)$ , where  $N$  is the number of flows.

We performed queueing simulations in a videoconferencing setting using DCVC flows. The bursty video traffic, which has a peak-to-average rate ratio over 15, is transmitted over a fixed-rate link. We examined the mean waiting time of applying LQF scheduler in the MIMO model and as a comparison, the mean waiting time of applying a simple FIFO scheduler in the MISO model. In the latter, we assume a flow is no longer restricted to one packet per slot. A flow can access more power resources when other flows are idle. We discovered that the fixed-rate connection in the MIMO model causes long delay and large buffer queueing, even when the system load is light. This prompts us to look at more flexible rate allocation using multi-rate links, which are described next.

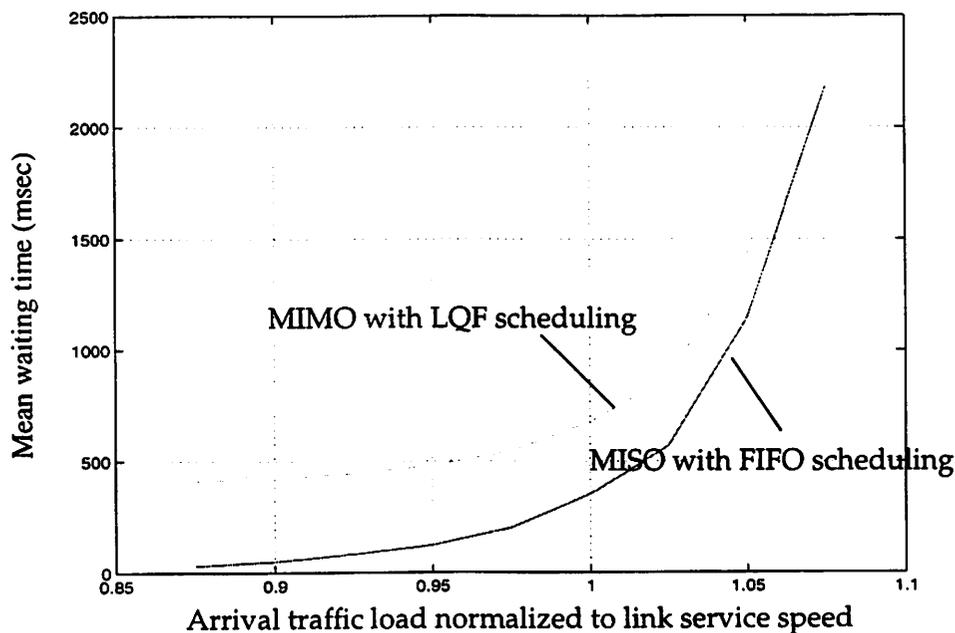


Figure 5.4 Queueing simulation results of compressed video traffic over the wireless link. Significant performance gaps in mean waiting time exist between MIMO and MISO queues.

### 5.3 Scheduling Problem of Multi-Rate Links

In light of recent proposals of wideband CDMA (WCDMA) in the third generation cellular to service multimedia data, there is a need to develop flexible architectures to facilitate more statistical multiplexing. In this and the next two sections, we discuss multi-rate links by comparing two configurations of WCDMA networks and their maximum cell throughputs. The two WCDMA configurations considered provide multiple data rates by different means. High Speed CDMA (HS-CDMA) assigns each user a single code with small spreading gain to enable a high transmission rate when it is needed. In contrast, Multi-Code CDMA (MC-CDMA) employs codes with a large spreading gain but permits a user to acquire more than one code for higher transmission rates. Key aspects of the schemes are listed in Table 5.2. To make a fair comparison, we assume the chip rates are equal. The difference in spreading gains hence results in different symbol (transmission) rates. In contrast with wireline networks, which can only perform SM in time, CDMA users' traffic is statistically multiplexed in power. When an outage occurs and the instantaneous arrival rate exceeds the link capacity, the link quality degrades and all users see an increase in bit error rate (BER). In the MC-CDMA configuration, there is also self-interference between spreading codes of the same user.

There is another configuration delivering the same performance as HS-CDMA, which is a hybrid of CDMA and TDMA. In this configuration, a single CDMA spreading code is shared by many users. The access of the spreading code is scheduled in a TDMA fashion. This hybrid configuration can be emulated by a HS-CDMA system through assigning the same TDM schedule to HS-users. To avoid repetitive discussions, we thus omitted the hybrid configuration in our comparison.

Comparisons in the following sections are made based on total cell throughputs, given different receiver structures and various combinations of power and reliability (characterized by signal-to-noise ratio) requirements. We assume

both systems are designed to be flexible and allow users to access the maximal bit rate, which is upper bounded by the cell throughput. Characterizing the throughput helps the evaluation of the two access models. We acknowledge that physical layer enhancements including adaptive equalization, directional antenna array, and space time coding, are not explicitly incorporated into the link layer problem formulation discussed below.

Table 5.2 A comparison of WCDMA network configurations. Assume their chip rates are the same.

Scheme	High speed CDMA (HS-CDMA)	Multicode CDMA (MC-CDMA)
Spreading code	one for each user	multiple codes for each user
Processing gain	fixed, low	fixed, high
Operation	same as baseline CDMA	packets are assigned to multiple spreading codes
Statistical multiplexing	power	power
Outage effect	low SNR increased BER	low SNR self interference increased BER

The next two sections are organized around two receiver structures: matched filter receivers and multiuser MMSE receivers. A section is devoted to each and is further divided into different operation scenarios, which represent a combination of power and SINR constraints. Matched filter receivers, which have been extensively studied and are easy to implement, treat signals from other CDMA users as white Gaussian noise. The white noise assumption is unjustified because spreading sequences have structures. With the advance of microelectronics, more complex receivers can be implemented to take advantage of the sequence structures. Recently the benefits of applying optimal linear multiuser receivers have been characterized analytically under certain conditions [69][70]. This advanced receiver structure delivers better spectral efficiency than the conventional matched filter does. As we will show, in some operation scenarios, the application of multiuser receivers eliminates the performance gap between the two configurations.

Operation scenarios may apply to either downlink or uplink constraints. Table 5.3 listed the scenarios and the sections they appear. Scenario 1, due to its total cell power constraint, mostly applies to downlink situations. Limiting the total transmitting power controls the intercell interference. Scenario 3 and 4 apply individual power constraints per active code. They represent the uplink transmission in which power or linearity of signal amplifiers on mobiles are the capacity bottleneck. Scenario 3 marks the SINR column as ARQ, which stands for automatic request for retransmission. Instead of the minimum SINR requirement, the system uses ARQ to provide guaranteed delivery. Lastly, Scenario 2 removes all the power constraints. It reflects that the capacity of CDMA is interference-limited, either in the downlink or uplink.

Table 5.3 A list of operation scenarios discussed in this paper.

Scenario	Total Power	Indiv. Power	SINR	Applicable Link
1 (Sec. 5.4.1 & 5.5.1)	V		V	Downlink
2 (Sec. 5.4.2 & 5.5.2)			V	Both
3 (Sec. 5.4.3)		V	ARQ	Uplink
4 (Sec. 5.4.4 & 5.5.3)		V	V	Uplink

## 5.4 Matched Filter Receivers

For CDMA matched filter receivers, the signal-to-interference-and-noise ratio (SINR) has been defined in (Eq 5.1). Consider the constrained case in which all flows have the same reliability requirements. Assume perfect power control, that is, all received signals have an equal power,  $P$ . (Eq 5.1) can be rewritten as a function of  $P$ , the number of interferers  $m-1$ , the spreading gain  $N_m$  (the subscript denotes the number of users in a cell), and the background noise  $\sigma^2$ . Specifically,

$$SINR = \frac{P \cdot N_m}{(m-1) \cdot P + \sigma^2} \quad (\text{Eq 5.8})$$

In the following discussions, we will apply the above equation with different power and SINR constraints to evaluate the throughput performance of HS-CDMA and MC-CDMA. Based on the combinations of these constraints, scenarios are created and discussed separately. A table in each scenario is presented at the beginning to summarize the key differences.

### 5.4.1 Scenario 1

Table 5.4 Scenario 1 assumptions

Constraints	Total power	Individual power	SINR requirement
Condition	Bounded	Limited by the total power	Yes

In this scenario, the cell has its total power bounded. Individual users must share the power and transmissions must meet the SINR requirements regardless the number of codes used. The following derivation assumes the simplest case when there is only one user in the cell. The same derivation technique can be applied to more than one users with more algebraic manipulations. Since their results do not differ, we apply the simpler derivation. In the case of HS-CDMA, the user is assigned with one spreading code and all the power is assigned exclusively to that code. In the case of MC-CDMA, the user applies  $m$  codes and the power is equally divided between them. The derivation will demonstrate that in order to combat the self-interference, MC-CDMA with  $m$  codes needs to increase its processing gain by more than  $m$  times of the code used in HS-CDMA. Since the increase in processing gain reflects to the decrease in transmission rate, the derivation concludes that HS-CDMA allows a higher throughput and thus performs better in this scenario. It is not difficult to show when there are more than one user, and the total cell power is bounded, HS-CDMA still fares better.

Let  $\beta$  be the target signal to interference and noise ratio;  $P_t$  is total cell power;  $N_m$  is the minimum processing gain with  $m$  spreading codes;  $\sigma^2$  denotes the background noise.

For a single spreading code in the case of HS-CDMA,

$$\frac{P_t \cdot N_1}{\sigma^2} \geq \beta \quad (\text{Eq 5.9})$$

When the user applies MC-CDMA with  $m$  codes, each code sequence is allocated with a power one- $m$ th of  $P_t$ . The following equation states the condition that the SINR requirement is satisfied.

$$\frac{\frac{P_t}{m} \cdot N_m}{(m-1) \cdot \frac{P_t}{m} + \sigma^2} \geq \beta$$

$$N_m \geq \left( \beta \cdot \left[ (m-1) \cdot \frac{P_t}{m} + \sigma^2 \right] \right) / \frac{P_t}{m} \quad (\text{Eq 5.10})$$

Combine (Eq 5.9) and (Eq 5.10) to show:

$$N_m \geq \left[ m + \frac{P_t}{\sigma^2} \cdot (m-1) \right] \cdot N_1 \geq m \cdot N_1 \quad (\text{Eq 5.11})$$

The processing gain needs to increase more than  $m$  times to satisfy the SINR constraint. Therefore, it is better to use less codes than more codes.

## 5.4.2 Scenario 2

Table 5.5 Scenario 2 assumptions

Constraints	Total power	Individual power	SINR requirement
Condition	Unbounded	Unbounded	Yes

Different from the previous scenario, this one removes all the power constraints. Neither total power nor individual power is bounded. In this case, the background noise is negligible and no longer affects the capacity. The cell throughput does not go to infinity, however, because of the interference. Recall the basic formulation from (Eq 5.8):

$$SINR = \frac{P \cdot N_m}{(m-1) \cdot P + \sigma^2}$$

The power control feasibility test derived in Section 5.2.1 on page 100 is rewritten as follows.

$$m \cdot \frac{\beta}{\beta + N_m} < 1 \quad (\text{Eq 5.12})$$

When the SINR requirements differ, the above feasibility condition has a more general form, which can be found in [5][76][78]. Notice that the ratio is a function of  $\beta$  and processing gain. As  $N_m$  decreases, the ratio  $\frac{\beta}{\beta + N_m}$  increases and thus reduces the number of users. However, the total cell thruput, which is defined as the sum of every user's thruput in a cell, increases. As shown in (Eq 5.13), HS-CDMA applies the processing gain  $N_1$ , which is one- $m$ th of the DS-CDMA processing gain. The  $m$ -fold transmission rate speed-up increases the total cell thruput. In this scenario with matched filter receivers, HS-CDMA again performs better.

$$N_m = m \cdot N_1 \quad (\text{Eq 5.13})$$

$$\text{Total cell thruput of HS-CDMA: } m \cdot \frac{\beta + N_1}{\beta} = \frac{m \cdot \beta + N_m}{\beta} \quad (\text{Eq 5.14})$$

$$\text{Total cell thruput of MC-CDMA: } 1 \cdot \frac{\beta + N_m}{\beta} = \frac{\beta + N_m}{\beta} \quad (\text{Eq 5.15})$$

An intuitive explanation about the favoring of HS-CDMA in Section 5.4.1 and Section 5.4.2 is that active spreading codes interfere with each other. Therefore, it is advantageous to minimize the number of codes used.

### 5.4.3 Scenario 3

Table 5.6 Scenario 3 assumptions

Constraints	Total power	Individual power	SINR requirement
Condition	Unbounded	Fixed	No; guaranteed delivery through retransmission (ARQ)

The third scenario is different from the previous two in that there is no SINR requirement but there is a power constraint per code. The setting is a CDMA uplink with perfect power control (equal receiving power) from all mobiles. Automatic retransmission is employed to provide guaranteed delivery for data services. This formulation was first introduced by Oh and Wasserman in [53], who showed the optimal processing gain for maximal cell throughput is linearly proportional to the number of active users (or codes) as in (Eq 5.16). In this equation,  $c$  is a constant depending on specific modulation and channel coding techniques.

$$\text{Optimal processing gain } N_{max}|_m = c \cdot (m - 1 + \eta) \quad (\text{Eq 5.16})$$

where  $\eta = \frac{L\sigma^2}{P_r}$  is the noise to receiving power ratio.  $P_r$  is the fixed receiving

power at the base station assuming optimal power control.  $L$  is the chip rate and  $\sigma^2$  is the background noise normalized to chip rate. The Oh and Wasserman result concluded that at the maximal throughput (by optimizing the processing gain), the probability of a successful packet transmission is *constant*, regardless of the number of active users.

Define  $R_m = \frac{L \cdot P_s}{N_{max}}$  as the effective transmission rate and  $\theta = R_m \cdot m$  as the total cell throughput, where  $P_s$  is the probability of a successful transmission. Under the throughput maximization condition,  $P_s$  is a constant. (Eq 5.16) can be rewritten as:

$$\theta + R_m \cdot (\eta - 1) = \frac{L \cdot P_s}{c} = \gamma, \text{ where } \gamma \text{ is a constant.} \quad (\text{Eq 5.17})$$

Depending on the value of  $\eta$ ,  $\theta$  may be greater or less than  $\gamma$ . Table 5.7 listed the conditions and their favorable access techniques. A graphical representation is shown in Figure 5.5.

It is worthwhile to explain the three different outcomes in cell throughput. One assumption of this scenario is that total power is unbounded, despite that individual receiving power is fixed. As  $\eta > 1$ , the background noise contributes more

Table 5.7 Changes in the total cell throughput at various conditions.

Condition	Relation	$m$	$R_m$	Thruput	Favoring
$\eta = 1$	$\theta = \gamma$	↗	↘	Unchanged	No difference
$\eta > 1$	$\theta + R_m \cdot (\eta - 1) = \gamma$	↗	↘	↗	MC-CDMA
$\eta < 1$	$\theta - R_m \cdot (1 - \eta) = \gamma$	↗	↘	↘	HS-CDMA

interference than a single interferer does. Therefore, with each additional code invoked in the cell, the contribution to throughput outweighs its contribution to interference. Systems operated in this region is considered power limited and MC-CDMA works better. The opposite condition happens when  $\eta < 1$ , in which interference per code is stronger than background noise. In this case, like the first two scenarios discussed before, the system capacity is interference limited and the less number of active codes, the larger the capacity. The third condition, in which interference per code is exactly the same as the background noise, the throughput is the same regardless of the access configuration.

#### 5.4.4 Scenario 4

Table 5.8 Scenario 4 assumptions

Constraints	Total power	Individual power	SINR requirement
Condition	Unbounded	Fixed	Yes

This last scenario has a similar set of constraints as Scenario 3 with the addition of SINR requirement. The constraints call for fixing the receiving power

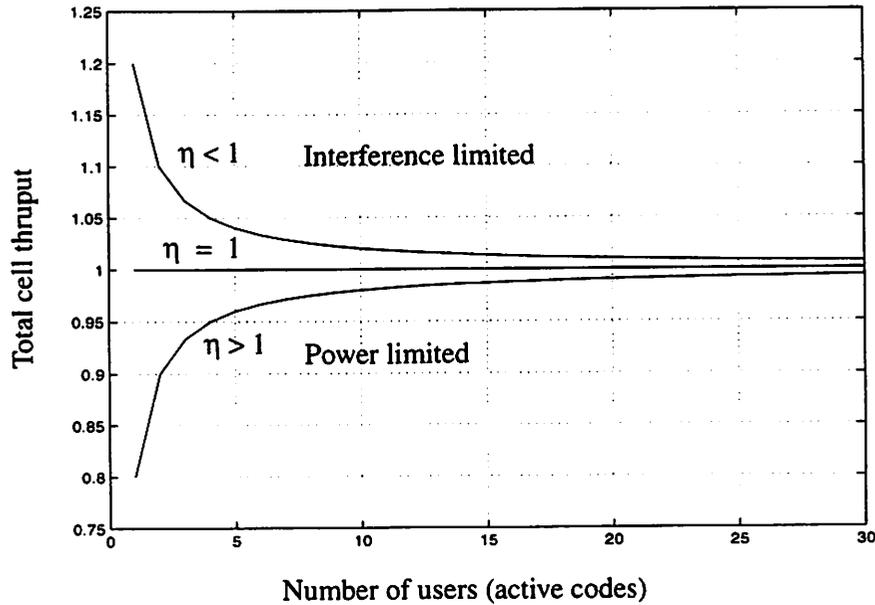


Figure 5.5 A graphical illustration on the changes in the total cell thruput at various conditions.

per code but not the number of codes a mobile can acquire. In practical systems, this could happen if the transmitting power/energy is limited by the linearity of the signal amplifier on a mobile. Without loss of generality, we assume there is only one user in a cell. The results can be applied well to the multiuser case. For a single spreading code in the case of HS-CDMA,

$$\frac{P_r \cdot N_1}{\sigma^2} \geq \beta \quad (\text{Eq 5.18})$$

When the user applies MC-CDMA with  $m$  codes, each code sequence is allocated with a fixed power  $P_r$ . The following equation states the condition that  $\beta$  is satisfied.

$$\frac{P_r \cdot N_m}{(m-1) \cdot P_r + \sigma^2} \geq \beta \quad (\text{Eq 5.19})$$

As the derivation demonstrates, the favorable choice between HS-CDMA and MC-CDMA also depends on the relative strength of the noise power. The

results listed in Table 5.9 are similar to the discussion in Scenario 3 and hence are not repeated here.

Table 5.9 Changes in the total cell throughput at various conditions.

Condition	Relation	Transmission Rate	Favoring
$P_r = \sigma^2$	$N_m = mN_1$	Unchanged	No difference
$P_r < \sigma^2$	$N_m < mN_1$	↗	MC-CDMA
$P_r > \sigma^2$	$N_m > mN_1$	↘	HS-CDMA

## 5.5 Multiuser MMSE Receiver

As we have shown in the previous section, matched filter receivers ignore the spreading code structure and treat the interference from other users as white noise. Therefore in most scenarios listed, this leads to the preference of HS-CDMA to minimize the number of active codes (interferers). In a CDMA cell, however, spreading sequences in use can be communicated to receivers at base stations and mobiles. They can then be used to jointly detect and remove unwanted signals. Adaptive interference suppression techniques including decorrelators, MMSE receivers and successive interference cancellation [51], thus have attracted a lot of attention.

Among multiuser linear detectors, the MMSE receiver has been shown to be the most effective one. While most research activities concentrate on physical layer issues, some new results in analyzing power control and link capacity have been emerging [69]-[72]. Unlike matched filter receivers, these new results typically depend on specific structures of spreading codes and power constraints. Since no known unified formulation is available, in the following discussion, we list the three analytical results and their associated capacity/throughput constraints.

The first result requires only a weak condition on the codes, the maximal rank condition. The condition requires any subset of spreading codes, which has a size  $N_m$  or smaller, are linearly independent.  $N_m$  is denoted as the spreading gain

of the system when there are  $m$  users in the cell. When  $m$  is smaller than  $N_m$ , the set of all active codes must be linearly independent. This result only applies to the case with no power constraints. It is valid for the non-limiting case, in which  $m$  and  $N_m$  are finite.

**Analytical Result 1 (AR1):** non-limiting case with maximal rank, linearly independent spreading codes

For  $m$  active users with SINR requirement  $\beta$  and processing gain  $N_m$ , a feasible transmitting power assignment exists if and only if the following inequality is satisfied:

$$\frac{m}{N_m} \cdot \frac{\beta}{\beta + 1} < 1 \quad (\text{Eq 5.20})$$

When there are power constraints, more constraints have to be imposed on the codes to get results.

The second result shown in (Eq 5.21) employs a different set of codes known as the *Welch Bound Equality* (WBE) codes. The quoted result again applies to the non-limiting case in which  $m$  and  $N_m$  are finite [72]. The power constraint is also incorporated in the formulation.  $P$  is the fixed receiving power for each active code at the base station assuming optimal power control.  $\sigma^2$  is the background noise. Note that as  $P$  goes to infinity, (Eq 5.21) turns to the same form as (Eq 5.20).

**Analytical Result 2 (AR2):** non-limiting case with WBE codes

A feasible power assignment exists if and only if the following inequality is satisfied:

$$\frac{m}{N_m} \cdot \frac{1}{\frac{\beta + 1}{\beta} - \frac{\sigma^2}{P}} < 1 \quad (\text{Eq 5.21})$$

The third result shown in (Eq 5.22) assumes general Pseudo-Noise (PN) signature sequences [69]. Because of the less imposed assumption on codes, the quoted result only applies to the limiting case in which the ratio of  $m$  and  $N_m$  is fixed but  $m$  asymptotically approaches infinity. (Eq 5.22) also turns to the same form as (Eq 5.20) as  $P$  goes to infinity.

**Analytical Result 3 (AR3):** limiting case with PN codes

Again, a feasible power assignment exists if and only if the following inequality is satisfied:

$$\frac{m}{N_m} \cdot \frac{\beta}{1 + \beta} \cdot \frac{1}{1 - \frac{\beta \cdot \sigma^2}{P}} < 1 \quad (\text{Eq 5.22})$$

In the previous section, we only assume the spreading codes for matched filter receivers are drawn from a pool of pseudorandom sequences. This corresponds to the third result (AR3). A fair comparison can only be made between (Eq 5.12) for matched filters and (Eq 5.22) for multiuser receivers. Furthermore, a separate study has found with WBE codes, both matched filter receivers and multiuser receivers deliver the same performance [72]. There is no known analytical results that correspond to AR1, for matched filters.

Careful readers may have noticed the three equations (Eq 5.20), (Eq 5.21) and (Eq 5.22) can all be written in the form  $\frac{m}{N_m} < \text{constant}$ , where the constant depends on the result. This is no coincidence. These constraints reflect the basic conservation law governing the trade off between the performance of one user and the others. (Eq 5.20) stems from the property that the sum of Mean Square Error (MSE) of all receivers is a constant. Decreasing the MSE of one user leads to the increase of some other's MSE. A further detailed discussion can be found at Section 3.6 in [70].

In the rest of the section, we apply the same methodology as in the previous section to evaluate the throughput performance of HS-CDMA and MC-CDMA in different scenarios. The three results are incorporated when applicable.

### 5.5.1 Scenario 1

Table 5.10 Scenario 1 assumptions

Constraints	Total power	Individual power	SINR requirement
Condition	Bounded	Limited by the total power	Yes

In this scenario, which has the same set of constraints as in Section 5.4.1 on page 111, the problem is formulated on a CDMA downlink with the total cell power bounded. Suppose there is only one user in the cell. In the case of HS-CDMA, the user is assigned with one spreading code. In the case of MC-CDMA, the user applies  $m$  codes. Transmissions must meet the SINR requirements regardless the number of codes. In Section 5.4.1, we found for matched filter receivers, HS-CDMA works better. For multiuser receivers, the following derivation will demonstrate the same conclusion that in order to combat the self-interference, MC-CDMA with  $m$  codes needs to increase its processing gain by **more** than  $m$  times of the code used in HS-CDMA. Since the increase in processing gain reflects to the decrease in transmission rate, the derivation concludes that HS-CDMA allows a higher transmission rate and thus performs better in this scenario.

Due to the power constraints, only the second and third results (AR2, AR3) are applicable. Let  $\beta$  be the target signal to interference and noise ratio;  $P_1$  is the total cell power;  $N_m$  is the minimum processing gain with  $m$  spreading codes;  $\sigma^2$  denotes the background noise.

#### **AR2: non-limiting case with WBE codes**

For a single spreading code in the case of HS-CDMA, from (Eq 5.21) we can derive the following constraint. In (Eq 5.23),  $P$  is replaced with  $P_1$  because the total cell power is devoted to this active code.

$$1 \cdot \frac{1}{N_1} \cdot \frac{1}{1 + \frac{1}{\beta} - \frac{\sigma^2}{P_t}} \leq 1 \quad (\text{Eq 5.23})$$

When the user applies MC-CDMA with  $m$  codes, each code sequence is allocated with a power one- $m$ th of  $P_t$ . The constraint has the following form.

$$m \cdot \frac{1}{N_m} \cdot \frac{1}{1 + \frac{1}{\beta} - \frac{m\sigma^2}{P_t}} \leq 1 \quad (\text{Eq 5.24})$$

Combine (Eq 5.23) and (Eq 5.24) to show:

$$N_m \geq \frac{m}{1 + \frac{1}{\beta} - \frac{m\sigma^2}{P_t}} \geq \frac{m}{1 + \frac{1}{\beta} - \frac{\sigma^2}{P_t}} \geq m \cdot N_1 \quad (\text{Eq 5.25})$$

The processing gain needs to increase more than  $m$  times to satisfy the SINR constraint. Therefore, it is better to use less codes than more codes.

### **AR3:** limiting case with PN codes

In this case, the analytical result was obtained by assuming a large spreading gain and a large number of users. Therefore, we approach the case by comparing  $N_m$ , a large spreading gain with  $m$  users, and  $N_{mn}$ , a even larger spreading gain with  $m*n$  users. (Eq 5.26) and (Eq 5.27) state the constraints when there are  $m$  and  $m*n$  users, respectively. They are derived from (Eq 5.22) by replacing the proper parameters. Combining these two equations, we show  $N_{mn} \geq n \cdot N_m$  in (Eq 5.28). Like the previous result, the processing gain needs to increase more than the multiple of codes to satisfy the SINR constraint. HS-CDMA is thus a better choice.

$$m \cdot \frac{1}{N_m} \cdot \frac{\beta}{1 + \beta} \cdot \frac{1}{1 - \frac{m \cdot \beta \cdot \sigma^2}{P_t}} \leq 1 \quad (\text{Eq 5.26})$$

$$m \cdot n \cdot \frac{1}{N_{mn}} \cdot \frac{\beta}{1 + \beta} \cdot \frac{1}{1 - \frac{m \cdot n \cdot \beta \cdot \sigma^2}{P_t}} \leq 1 \quad (\text{Eq 5.27})$$

$$N_{mn} \geq \frac{m \cdot n \cdot \frac{\beta}{1 + \beta}}{1 - \frac{m \cdot n \cdot \beta \cdot \sigma^2}{P_t}} \geq \frac{m \cdot n \cdot \frac{\beta}{1 + \beta}}{1 - \frac{m \cdot \beta \cdot \sigma^2}{P_t}} \geq n \cdot N_m \quad (\text{Eq 5.28})$$

## 5.5.2 Scenario 2

Table 5.11 Scenario 2 assumptions

Constraints	Total power	Individual power	SINR requirement
Condition	Unbounded	Unbounded	Yes

In this scenario, which has the same set of constraints as in Section 5.4.2 on page 112, there are no power constraints and the cell capacity is limited by interference. Given the SINR requirement, one can write down the power control feasibility test as shown in (Eq 5.29) for MMSE receivers.

$$\frac{m}{N_m} \cdot \frac{\beta}{\beta + 1} < 1 \quad (\text{Eq 5.29})$$

The test states that a feasible transmitting power assignment exists to satisfy all SINR requirements if the summation is less than one. (Eq 5.29) is identical to (Eq 5.20) in AR1, which is the most general result without imposing power constraints.

In (Eq 5.29), the ratio  $\frac{m}{N_m}$  is upper bounded by a constant. Changing the processing gain does not change the upper bound. As shown in (Eq 5.30), HS-CDMA applies the processing gain  $N_1$ , which is one- $m$ th of the DS-CDMA processing gain. The  $m$ -fold transmission rate speed-up has no impact on the total cell throughput as shown in (Eq 5.31) and (Eq 5.32).

$$N_m = m \cdot N_1 \quad (\text{Eq 5.30})$$

$$\text{Total cell thruput of HS-CDMA: } m \cdot N_1 \cdot \frac{\beta + 1}{\beta} = N_m \cdot \frac{\beta + 1}{\beta} \quad (\text{Eq 5.31})$$

$$\text{Total cell thput of MC-CDMA: } 1 \cdot N_m \cdot \frac{\beta + 1}{\beta} = N_m \cdot \frac{\beta + 1}{\beta} \quad (\text{Eq 5.32})$$

Unlike the conclusion reached in Section 5.4.2, applying MMSE receivers results in an equal performance of the two configurations. This conclusion reflects the aforementioned conservation law governing the trade off between the performance of one user and the others.

### 5.5.3 Scenario 3

Table 5.12 Scenario 3 assumptions

Constraints	Total power	Individual power	SINR requirement
Condition	Unbounded	Fixed	Yes

The last scenario to be discussed has the same set of constraints as in Section 5.4.4. The constraints call for fixing the receiving power per code but not the number of codes a mobile can acquire. In practical applications, the linearity of the signal amplifier on a mobile can cause this power/energy constraint. Without loss of generality, we assume there is only one user in a cell. Unlike the result in Section 5.4.4 for matched filter receivers, which concludes the optimal choice depends on the relative strength of power and noise, the following derivations for multiuser receivers indicate a fixed upper bound of the user to spreading gain ratio. The upper bound only depends on SINR, receiving power and background noise but not on spreading gain. Therefore, there is no advantage to use more or less codes. The performances of HS-CDMA and MC-CDMA are equivalent.

Due to the power constraints, analytical results exist for AR2 and AR3 only. Let  $\beta$  be the target signal to interference and noise ratio;  $P_r$  is the fixed receiving power;  $N_m$  is the minimum processing gain with  $m$  spreading codes;  $\sigma^2$  denotes the background noise.

#### **AR2: non-limiting case with WBE codes**

We first restate (Eq 5.21):

$$m \cdot \frac{1}{N_m} \cdot \frac{1}{1 + \frac{1}{\beta} - \frac{\sigma^2}{P_r}} \leq 1$$

Rearrange the above equation to see the upper bound of the user to spreading gain ratio is fixed as shown below.

$$\frac{m}{N_m} \leq 1 + \frac{1}{\beta} - \frac{\sigma^2}{P_r} \quad (\text{Eq 5.33})$$

**AR3:** *limiting case with PN codes*

(Eq 5.34) is a simple rearrangement of (Eq 5.22). Again, it shows the upper bound of the user to spreading gain ratio is fixed.

$$\frac{m}{N_m} \leq \left(1 + \frac{1}{\beta}\right) \cdot \left(1 - \frac{\beta \cdot \sigma^2}{P_r}\right) \quad (\text{Eq 5.34})$$

## 5.6 Summary and Future Work

In this chapter, we investigated the joint power control and scheduling problems of fixed-rate links and multi-rate links. Understanding fixed-rate links is important to expand the current service offering of digital cellular while the study of multi-rate links is useful for the next generation cellular. We compared the cell throughput performance of two WCDMA configurations: HS-CDMA and MC-CDMA. The results are summarized in Table 5.13 by receiver structures and problem constraints. We found when matched filter receivers are applied, HS-CDMA fares better in most occasions because of fewer interferers. Multiuser receivers, on the other hand, make use of the structure of the signature sequence to reduce inter-

ference most effectively. In two of the three scenarios, multiuser receivers eliminate the performance gap between HS- and MC-CDMA.

Table 5.13A summary of scenario constraints and the favored WCDMA configuration

Receiver Structure	Constraints			Link configuration	
	Total Power	Individual power	SINR	HS-CDMA	MC-CDMA
Matched filter receiver	V		V	V	
			V	V	
		V	ARQ	$P_r > \sigma^2$	$P_r < \sigma^2$
		V	V	$P_r > \sigma^2$	$P_r < \sigma^2$
Multiuser receiver	V		V	V	
			V	No difference	
		V	V	No difference	

Recently, a study reported that using more codes and larger spreading gain reduces the outage probability of violating individual power constraints [38]. The reported result applies to multi-user MMSE receivers with PN codes. While this study concluded MC-CDMA has a higher throughput, its conclusion does not contradict what we have described in this paper. While AR3 in our multiuser receiver discussion considers PN codes, the quoted analytical result is for the limiting case of infinite number of users. The study in [38] considered the case in which the number of users is finite. As the paper pointed out, applying shorter PN codes results in bigger SINR fluctuation, which causes the higher outage probability. It is acknowledged that HS-CDMA is inferior in this scenario.

Although we reported some progress in exploring scheduling related issues, the problem domain is too broad to be addressed in a thesis chapter and there are many open questions we do not have answers. While the MISO (multi-input/single-output) queueing model has been studied for years and plenty of literatures are available, the generic MIMO (multi-input/multi-output) model has yet to find its way in the mainstream research. In our experience of attempting to develop

a scheduler for MIMO, we encountered some fundamental differences which prevented us from invoking known theories developed for MISO. To name a few:

1. The work conservation law existing in MISO has no correspondence in MIMO. The optimality of many scheduling policies in MISO is established on the basis of conservation law. Therefore, these results cannot be extended to MIMO.
2. The criterion of comparing scheduling algorithms in the multi-dimensional admissible region is yet to be developed. For example, when the schedulable region of one algorithm has only partial overlap with the region of another algorithm, what criteria should be applied to choose the 'better' one?
3. While we have shown a couple of NP-complete problems, polynomial time heuristics need to be developed for practical applications. When the NP-complete algorithm is replaced by the heuristic, how can we ensure the basic properties such as stability still hold?

We believe the MIMO queueing model opens an interesting new direction in queueing theory and related disciplines. More research efforts are needed to answer the above questions.

## **5.7 Appendix A: Proof of NP-completeness of the MC-CDMA Admission Control Problem**

While fixed-rate links restrict the access of spreading code to be one per user, multi-code (MC) CDMA allows a user to acquire more than one code. The admission control problem of MC-CDMA thus differs from that of fixed-rate links. For an average arrival rate vector  $\hat{\lambda}$ , find a large integer  $T$  such that  $\hat{\lambda} \cdot T$  is an integer vector.  $\lambda_i \cdot T$  is the number of packets for the  $i$ th flow arriving in  $T$  time slots. Each of the packets consumes a power index  $\gamma_i$  of resources. The admission control problem is to find if there exists an allocation of packets into  $T$  slots such that

the power control feasibility test is not violated at any time. Suppose  $t_{i,j}$  packets of the  $i$ th flow are assigned to the  $j$ th time slot. Admission control verifies:

$$\sum_j t_{i,j} = \lambda_i T \text{ and } \forall j, \sum_i (t_{i,j} \cdot \gamma_i) < 1$$

**Theorem 5.2** The computational complexity of the MC-CDMA admission control problem is NP-complete.

Proof of Theorem 5.2: The problem has the same formulation as the well-known Bin packing problem. See p. 226 of [25].

## 5.8 Appendix B: Proof of the Optimality of Longest Queue First Policy

**Definition:** Consider two discrete-time processes,  $X$  and  $Y$ . Define the process  $X$  to be stochastically smaller than the process  $Y$ , expressed as,

$$X \leq_{st} Y \tag{Eq 5.35}$$

if

$$P(f(X) > z) \leq P(f(Y) > z), \text{ for every } z \in R \tag{Eq 5.36}$$

where  $f: R^{\mathbb{Z}_+} \rightarrow R$  is measurable and  $f(\hat{x}) \leq f(\hat{y})$  for every  $\hat{x}, \hat{y} \in R^{\mathbb{Z}_+}$  and  $x_i \leq y_i$  for  $i \in \mathbb{Z}_+$ .

**Theorem 5.3** Assume packet arrivals in different queues are *i.i.d. Bernoulli* processes. Let  $Q$  be the process of total number of packets in the system when the initial state is  $Q_0$  and some policy  $\pi$  acts on it and  $Q_{LQF}$  the corresponding process when the LQF policy acts on the system. Then,

$$Q_{LQF} \leq_{st} Q \tag{Eq 5.37}$$

**Lemma 5.1** For every policy  $\pi$ , there exists a policy  $\tilde{\pi}$  that acts similarly to LQF at  $t = 0$  and is such that when the system is in state  $\vec{Q}_0$  and policies  $\pi$  and  $\tilde{\pi}$  act on it, the corresponding process of  $Q$  and  $\tilde{Q}$  can be constructed by appropriate coupling of the arrival and service processes so that

$$\tilde{Q}(t) \leq Q(t), \text{ almost surely} \quad (\text{Eq 5.38})$$

Proof of Lemma 5.1:

Let vectors  $\vec{Q}$  and  $\tilde{\vec{Q}}$  be the queue length processes under policies  $\pi$  and  $\tilde{\pi}$ . The queue length dynamics can be described as

$$\vec{Q}(t+1) = \vec{Q}(t) - \vec{D}(t) + \vec{A}(t) \quad (\text{Eq 5.39})$$

where  $\vec{D}(t)$  and  $\vec{A}(t)$  represent the service and arrival processes respectively. Without loss of generality, we assume the service process at time  $t$  only depends on the queue lengths at the beginning of the time slot  $t$ , which is  $\vec{Q}(t)$ . The queue length vector at the beginning of the next time slot is thus equal to that at the current instant after the scheduled transmissions plus the newly arrived packets.

At time  $t = 0$ , policies  $\pi$  and  $\tilde{\pi}$  act differently. We classify queues by the service they receive under both policies to three sets,  $S_1$ ,  $S_2$  and  $S_3$ . For any queue in  $S_1$ , it is served under  $\pi$  but not  $\tilde{\pi}$ . For any queue in  $S_2$ , it is served under  $\tilde{\pi}$  but not  $\pi$ . For queues in  $S_3$ , they are served by both policies.

The LQF policy maximizes the number of concurrent transmissions and always serves the longest queues. Therefore, the size of the set  $S_2$ ,  $|S_2|$ , is larger than or equal to the size of the set  $S_1$ ,  $|S_1|$ . Furthermore,

$$\forall i \in S_1, \forall j \in S_2, Q_i(0) \leq Q_j(0) \quad (\text{Eq 5.40})$$

Since  $|S_1| \leq |S_2|$ , we can construct the fourth set  $S_4$ , which has the size of the set equal to  $|S_2| - |S_1|$ , by randomly selecting  $|S_2| - |S_1|$  queues in  $S_2$ . After that, those queues are removed from  $S_2$  so that  $S_1$  and  $S_2$  would have the same size.

We define the mapping  $g$  as a one-to-one function from a queue in  $S_1$ , say  $i$ , to a queue in  $S_2$ ,  $g(i)$ . From (Eq 5.40)(Eq 5.40), we can conclude that the following relation holds

$$\forall i \in S_1, g(i) \in S_2, Q_i(0) \leq Q_{g(i)}(0) \quad (\text{Eq 5.41})$$

After the scheduled transmissions under  $\pi$  and  $\tilde{\pi}$ , and before the arrivals, we define a new symbol,  $Q_i'(t)$  to denote the queue length at this transition period. It is easy to see the following relations.

$$\forall i \in S_1, g(i) \in S_2, Q_i'(0) = Q_i(0) - 1, \tilde{Q}_i'(0) = Q_i(0)$$

$$Q'_{g(i)}(0) = Q_{g(i)}(0) - 1, \tilde{Q}'_{g(i)}(0) = Q_{g(i)}(0) - 1 \quad (\text{Eq 5.42})$$

Next consider the arrivals at the end of time slot 0. We need to distinguish the following cases:

*Case 1:*  $Q_i(0) = Q_{g(i)}(0)$

From (Eq 5.42), we know that  $Q'_{g(i)}(0) = \tilde{Q}_i'(0)$  and  $Q_i'(0) = \tilde{Q}'_{g(i)}(0)$ . The  $i$ th queue under  $\pi$  and the  $g(i)$ th queue under  $\tilde{\pi}$  have the same length. By exchanging the indexes  $i$  and  $g(i)$  under  $\tilde{\pi}$ , we will have two pairs of queues with equal lengths. Now these two queues can be moved to the set  $S_3$  and let them have the same arrivals under  $\pi$  and  $\tilde{\pi}$ .

*Case 2:*  $Q_i(0) < Q_{g(i)}(0)$

Again from (Eq 5.42), we know that

$$Q'_{g(i)}(0) > \tilde{Q}'_{g(i)}(0) \geq \tilde{Q}'_i(0) > Q'_i(0) \quad (\text{Eq 5.43})$$

We need to divide it further into the following two subcases.

$$\text{Case 2a: } Q_i(0) = Q_{g(i)}(0) - 1$$

If both  $i$  and  $g(i)$  have arrivals under  $\pi$ , make the arrivals identical under  $\tilde{\pi}$ . If only  $g(i)$  has arrivals under  $\pi$ , make the arrivals identical under  $\tilde{\pi}$ . If only  $i$  has arrivals under  $\pi$ , make the arrivals identical under  $\tilde{\pi}$ . Observe that after the arrival,

$$Q_{g(i)}(1) = \tilde{Q}_i(1), Q_i(1) = \tilde{Q}_{g(i)}(1) \quad (\text{Eq 5.44})$$

Exchange queue indexes  $i$  and  $g(i)$  under  $\tilde{\pi}$  and move them to  $S_3$ .

$$\text{Case 2b: } Q_i(0) < Q_{g(i)}(0) - 1$$

Let the same arrival variables to apply under  $\pi$  and  $\tilde{\pi}$ .

Lastly, let the arrival and service variables under  $\pi$  and  $\tilde{\pi}$  be the same for all the queues in  $S_3$  and  $S_4$ . This concludes the operation at  $t = 0$ .

At the beginning of time slot  $t = 1$ , we have

$$\forall i \in S_1, g(i) \in S_2, Q_{g(i)}(1) > Q_i(1) + 1 \quad (\text{Eq 5.45})$$

$$\forall i \in S_1, g(i) \in S_2, Q_i(1) = \tilde{Q}_i(1) - 1, Q_{g(i)}(1) = \tilde{Q}_{g(i)}(1) + 1 \quad (\text{Eq 5.46})$$

$$\forall i \in S_3, Q_i(1) = \tilde{Q}_i(1) \quad (\text{Eq 5.47})$$

$$\forall i \in S_4, Q_i(1) > \tilde{Q}_i(1) \quad (\text{Eq 5.48})$$

The sum of the queue lengths under  $\pi$  is greater than or equal to the sum of the queue lengths under  $\tilde{\pi}$ .

Next, mathematical induction is used to show (Eq 5.45)-(Eq 5.48) are true for any  $t$ . First assume these equations are true at time  $t$ . The proof will be focusing

on the scheduling operations of queues in  $S_1$  and  $S_2$  because queues in  $S_3$  and  $S_4$  always have equal or shorter lengths under  $\bar{\pi}$ . One simply assigns the same service and arrival variables of  $S_3$  and  $S_4$  under  $\pi$  to those under  $\bar{\pi}$ .

For a queue  $i$  in  $S_1$  and its corresponding queue  $g(i)$  in  $S_2$ , their queue lengths satisfy the follow relation.

$$Q_{g(i)}(t) > Q_i(t) + 1 \quad (\text{Eq 5.49})$$

We distinguish the following cases:

Case I:  $Q_{g(i)}(t) = Q_i(t) + 2$ . From (Eq 5.45) to (Eq 5.48), we conclude  $\tilde{Q}_{g(i)}(t) = \tilde{Q}_i(t)$ . There are four subcases depending on the service and arrival processes.

Case Ia: If both queues  $i$  and  $g(i)$  are served under  $\pi$  and  $\bar{\pi}$ , and there is an arrival at queue  $i$ . At the beginning of time slot  $t + 1$ , we have

$$\tilde{Q}_i(t+1) = Q_{g(i)}(t+1), \tilde{Q}_{g(i)}(t+1) = Q_i(t+1) \quad (\text{Eq 5.50})$$

Exchange index  $i$  and  $g(i)$  under  $\bar{\pi}$  and move them to  $S_3$ . For other three arrival patterns (arrivals on both, none or  $g(i)$  only), apply the same arrival variables under  $\pi$  and  $\bar{\pi}$ . At the beginning of  $t + 1$ , they still satisfy (Eq 5.45) to (Eq 5.48).

Case Ib: If both queues  $i$  and  $g(i)$  are not served, this subcase can be treated in the same way as *Ia*.

Case Ic: If only  $i$  is served, apply the same arrivals under both policies and the four equations hold.

Case Id: If only  $g(i)$  is served, at the end of the service we have

$$Q'_{g(i)}(t) = \tilde{Q}'_i, \tilde{Q}'_{g(i)}(t) = Q'_i(t) \quad (\text{Eq 5.51})$$

Exchange index  $i$  and  $g(i)$  under  $\tilde{\pi}$  and move them to  $S_3$ . Then apply the same arrivals for both policies.

Case II:  $Q_{g(i)}(t) > Q_i(t) + 2$

Case IIa: If both queues are served, or none is served, or only  $i$  is served under  $\pi$  and  $\tilde{\pi}$ , apply the same arrivals and (Eq 5.45) to (Eq 5.48) hold.

Case IIb: If only  $g(i)$  is served, at the end of the service,  $\tilde{Q}'_i(t) = \tilde{Q}'_{g(i)}(t)$ . Now if there is an arrival at  $i$ , exchange indexes  $i$  and  $g(i)$  under  $\tilde{\pi}$  and move them to  $S_3$ . For the other three arrival patterns, apply the same arrivals under both policies and the four equations hold.

The above cases cover all the possible combinations of services and arrivals. We can conclude that (Eq 5.45) to (Eq 5.48) hold true at  $t + 1$ . By mathematical induction, the lemma holds. ■

Next we use the lemma to prove Theorem 5.3.

Proof of Theorem 5.3:

From Lemma 5.1, we know that for an arbitrary scheduling policy  $\pi$ , one can construct a policy  $\tilde{\pi}$  such that at time  $t = 0$ ,  $\tilde{\pi}$  acts similarly to the LQF policy and in later time slots, the total number of packets under  $\tilde{\pi}$  is always less than or equal to that under  $\pi$ . For the ease of further discussions, we denote the policy  $\tilde{\pi}$  as  $\pi_0$  and use the subscript to tell the ending time when it acts like LQF.

Now consider the scheduling policy  $\pi_1$  which acts similarly to LQF for time  $t \leq 1$ . Since  $\pi_0$  and  $\pi_1$  have the same queue length distribution at the beginning of  $t = 1$ , one can construct  $\pi_1$  to guarantee that in later time slots, the total number of packets under  $\pi_1$  is less than or equal to that under  $\pi_0$ . We can then

repeat this procedure to argue that the total number of packets under  $\pi_T$ , a policy conforms to LQF until time  $T$ , is always less than or equal to the number under  $\pi$ ,  $t = 0, \dots, T - 1$ .

Consider a function  $f$  as in the definition and time slots  $t_1, t_2, \dots, t_n \leq T$ .  $Q_{\pi}$  is the process of total number of packets under  $\pi$ . For all  $z \in R$ , we have

$$P(f(Q_{\pi_T}(t_1), Q_{\pi_T}(t_2), \dots, Q_{\pi_T}(t_n)) > z) \leq P(f(Q_{\pi}(t_1), Q_{\pi}(t_2), \dots, Q_{\pi}(t_n)) > z) \quad (\text{Eq 5.52})$$

Since  $\pi_T$  acts similarly to LQF at time  $t \leq T$ , we can then conclude

$$Q_{LQF}(t_i) = Q_{\pi_T}(t_i), i \in \{1, 2, \dots, n\} \quad (\text{Eq 5.53})$$

which leads to

$$P(f(Q_{LQF}(t_1), Q_{LQF}(t_2), \dots, Q_{LQF}(t_n)) > z) \leq P(f(Q_{\pi}(t_1), Q_{\pi}(t_2), \dots, Q_{\pi}(t_n)) > z) \quad (\text{Eq 5.54})$$

(Eq 5.54) states that the total queue length process of LQF is stochastically smaller than or equal to that of any arbitrary policy.

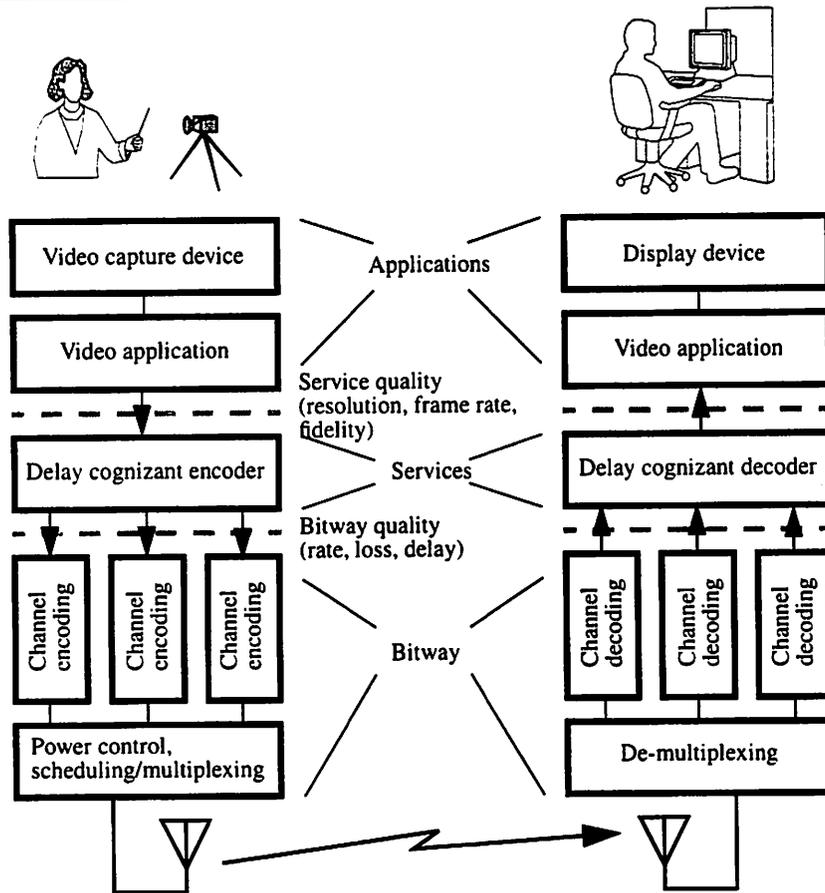
This completes the proof of the theorem. ■

# 6

---

## Future Work

---



In this dissertation, we presented a new, delay cognizant perspective for video coding and demonstrated a DCVC design that delivers good subjective quality even with substantial delay offsets for a portion of the total compressed video. We discussed applications of DCVC in a broad range of networking environments including the Internet and wireless. These applications showed that significant quality improvement and network capacity gain are achievable. We conducted sub-

jective quality evaluations on delayed video and surprisingly found delay segmented, low bit rate video could sometimes look subjectively better than video without delay. We identified the cause and constructed artificial stimuli to mimic the natural scenes which caused better subjective quality. These stimuli are simpler to analyze and yet still capture key attributes of this surprising finding. They could lead to the development of new insight into video compression design. We studied CDMA wireless as a paradigmatic transport of DCVC flows, since wireless is likely to be the bottleneck link in the future heterogeneous networking infrastructure. We examined the joint power control and scheduling problem of CDMA wireless in the context of the second and third generation cellular. We compared the throughput performance of wideband CDMA in various operation conditions. Although significant progress in DCVC and its related topics has been made since it was first proposed, there remain a number of open issues that are highlighted in this chapter.

There are two major research challenges ahead: one in video coding and the other in networking. For video coding, the integration of rate scalability, error resilience, and delay cognizance into a single coding algorithm will enable the full QoS abstraction (rate, loss, and delay) of video flows. As prior work, including ours, has focused on one or two aspects, a direct extension to all three may not be straightforward. In Section 6.1, we discuss one possible path that could lead to a fully QoS adaptive coding. For networking, the exploitation of QoS adaptive flows is challenging. A switch node should optimize its decision on when to delay packets, what flows to suspend and which packets to drop. Again, published work has focused on only one or two aspects of the problem. Section 6.2 presents several promising research directions in addressing the issues of QoS adaptive network control.

## **6.1 QoS Adaptive Video Coding**

Due to the heterogeneity of future networking environments and the multiplicity of receiver terminal capabilities, the current methodology of designing

video coding algorithms for specific environments will not provide an effective and efficient solution for all practical uses. Important issues such as end-to-end security, multisource-to-multidestination scalability, and the aforementioned heterogeneity are often compromised, and only later found to be incapable of meeting new demands. There are already several dozen image and video coding standards, public or proprietary, with formats used in different network and terminal settings and we are expecting more to appear. When these formats are applied to environments which they are not designed for, they perform poorly. One partial solution to resolve heterogeneity is transcoding, such as [24][62] and others have proposed. Although transcoding may provide a fix to legacy applications, its adaptability is limited to the formats known to transcoders. The growing list of coding formats is certainly putting an increasing burden. Another weakness is that applying a transcoder adds another vulnerable point in secret communications, because encrypted messages must be decrypted first to perform transcoding.

Although we believe a single, universal video coding standard should not be advocated, an effective and efficient solution to the heterogeneity problem demands new design principles. In Section 1.3 on page 11, we described the principles of loosely coupled joint source/channel coding and a companion flow architecture that potentially offers such a solution. In this architecture, applications including video coding are QoS adaptive, meaning that the flows are characterized by QoS attributes (rate, loss, and delay) and these attributes can be adapted for connection environments and terminal capabilities.

A QoS adaptive video coder achieves rate scalability, error resiliency, and delay cognizance at the same time. Besides delay cognizance described in this dissertation, common approaches to achieve rate scalability include adaptation of spatial resolution (frame size), temporal resolution (frame rate), and compressed quality. Approaches to achieve error resiliency include robust waveform coding, robust entropy coding and multiple description coding. In the case of unicast (one-to-one communications), a QoS negotiation is initiated first to set up the connec-

tion and the associated parameters. The coder then either generates flows or retrieves previously stored flows at the negotiated QoS. A more interesting case is multicast (one-to-many communications). The adaptive coder may choose to send the complete set of flows, from HDTV quality on highly reliable channels to low frame rate, low quality video on best effort channels. It is up to the network to pick and choose a subset of these flows to satisfy end users' demands. Adaptive network control is discussed in the next section.

Figure 6.1 presents a 3D view of the set of generated flows when a two-level decomposition is applied to each dimension. In the figure, the coordinates of each cube represent the QoS triplet (rate, loss, delay) of each flow. What this view does not show is data dependency of the flows. Data dependency is created because of the removal of both visual and statistical redundancy. It is easy to see that video data in a higher bit rate flow may depend on data in a lower bit rate flow. A high error resilient flow (such as those carrying high frequency transform coefficients) depends on a low error resilient flow (such as those carrying DC and low frequency coefficients). And should one choose, higher delay flows depend on lower delay flows. The cube nearest to the origin represents the core video information, the most visually significant data requiring the least bandwidth, least error resilient and lowest delay. The core flow may only carry key frames with the most

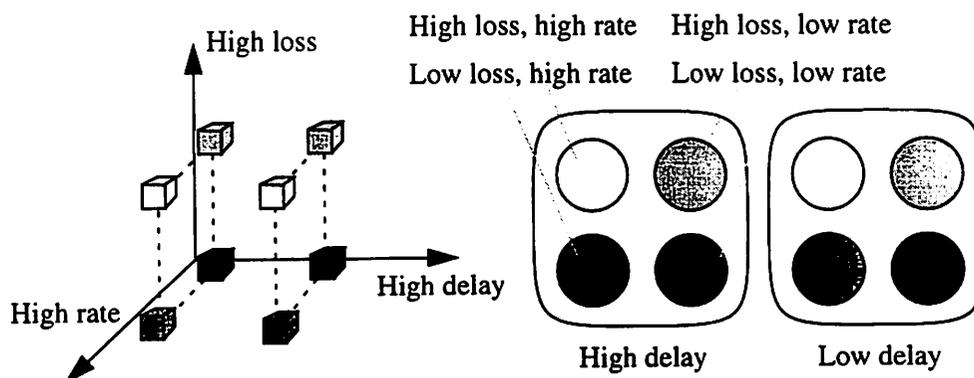


Figure 6.1 A 3D view of the set of generated flows when a two-level decomposition is applied to each dimension. The coordinates of each cube represent the QoS triplet (rate, loss, delay) of each flow.

aggressive compression. All other flows add adaptability in one or more dimensions to improve quality.

As prior work, including ours, has focused on one or two dimensions in this figure, developing a QoS adaptive coder is an open issue. One possibility is to extend the current design of DCVC, as illustrated on the right of Figure 6.1. The two-flow DCVC is extended into eight flows, with two levels of delay as illustrated by the two outer squares. At each delay level, video information is further decomposed into combinations of bit rate and loss. Since DCVC delay flows are processed independently, we envision the possibility of employing techniques like temporal and spatial resolution for rate scalability, and techniques like robust waveform coding for error resiliency. While this divide-and-conquer strategy could achieve our goal, a joint optimization of QoS parameters may lead to a better solution and requires further investigation.

## 6.2 QoS Adaptive Network Control

The development of QoS adaptive coders has a profound influence on networking operations such as admission control, scheduling, routing, and resource management. It provides flexibility and yet increases the complexity of traffic management. For example, at the link layer, as the set of QoS flows arrive at a switch node, the link scheduler must decide when to delay packets, what flows to suspend and which packets to drop. At the network layer, the routing algorithm

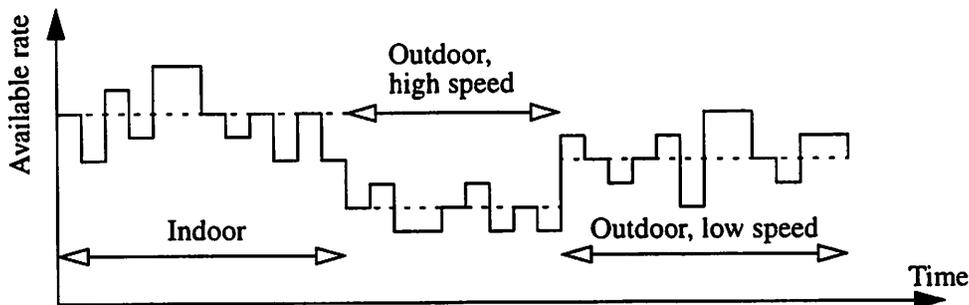


Figure 6.2 The available channel rate varies in heterogeneous wireless environments; slow time scale changes are associated with the connection environment and the speed of the mobile; fast time scale changes are associated with fadings.

must estimate the resource usage based on the flow characteristics to decide if the set of flows should travel through the same path. While the complexity of network management increases, potential gains of exploiting these QoS adaptive flows can be fairly significant, as seen in Chapter 3.

Among the aforementioned issues, one fundamental problem is link layer scheduling. Take wireless networking as an example. The available channel rate to a mobile user varies depending on locations and mobile speed. As shown in Figure 6.2, at a given bit error rate and the same channel bandwidth, the available link rate is fastest in an indoor environment and it is slowest when the mobile is moving high speed outdoors. Changes in connection environments typically reflect slow time scale variations of link rates, in the order of seconds or longer. Accompanied with slow time scale variations are fast time scale variations, which are resulted from radio propagation effects like path loss and fadings. Fast time scale variations are in the order of ten's of milliseconds. Both time scale variations may contribute to significant changes in bit rates, which are adapted by link scheduler.

One possible adaptation scheme is to map the three dimensions in Figure 6.1 to different time scales. One proposal is to map the rate dimension to the slow time scale, the delay dimension to the fast time scale, and the loss (resiliency) dimension to the fastest time scale (symbol-to-symbol variations). Figure 6.3 illustrates such mappings by showing the flows carried in each connection environment. Shaded circles are used to represent flows, following the legend in Figure 6.1. In the indoor environment (the first segment), all eight flows are carried to provide the best possible quality. In the outdoor, high speed environment (the second segment), only low rate flows are carried. These four low rate flows are combinations of two levels of delay and two levels of resiliency. Differential delay flows are used to adapt to fast time scale changes and differential loss flows adapt to channel bit errors. In the outdoor, low speed environment (the third segment), the link carries six flows, including four low rate flows and two low loss, high rate flows. In the example illustrated, the mappings of flows to time scales need not be

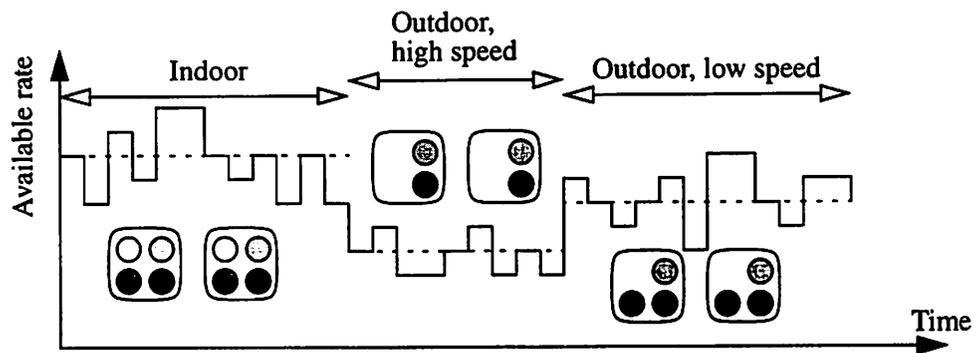


Figure 6.3 Different connection environments carry different sets of flows. Shaded circles are used to represent flows, following the legend in Figure 6.1.

unique. One may choose to transmit low delay flows only. Impacts on application quality by different mappings need to be evaluated. These issues will serve as good research topics well in coming years.

## Bibliography

- [1] "One-way Transmission Time," *ITU Recommendation G.114*, International Telecommunication Union, Feb. 1996.
- [2] "Video coding for low-bit rate communications: draft recommendation ITU-T H.263," International Telecommunications Union - Telecommunication Standardization Sector, May 1996.
- [3] J. B. Andersen, T. S. Rappaport, and S. Yoshida, "Propagation measurements and models for wireless communications channels," *IEEE Communications Magazine*, vol. 33, no. 1, pp. 42-9, Jan. 1995.
- [4] T. W. Anderson and J. D. Finn, *The new statistical analysis of data*, published by Springer-Verlag, 1996.
- [5] M. A. Arad and A. Leon-Garcia, "Scheduled CDMA: a hybrid multiple access for wireless ATM networks," *IEEE PIMRC*, Taipei, Taiwan, 1996.
- [6] J. Beck, B. Hope, and A. Rosenfeld, *Human and machine vision*, published by Academic Press, 1983.
- [7] T. Berger, *Rate distortion theory*, published by Prentice Hall, 1971.
- [8] S. Bradner and A. Mankin, "The recommendation for IP next generation protocol," *IETF RFC 1752*, 1995.
- [9] R. Braden et al., "Resource Reservation Protocol (RSVP) - version 1, functional specification," *IETF RFC 2205*, 1997.
- [10] T. Carney, "PC-MatVis", Neurometrics Institute, Berkeley, CA, see also [www.neurometrics.com](http://www.neurometrics.com).

- [11] T. Carney, Y. C. Chang, S. A. Klein, and D. G. Messerschmitt, "Effects of dynamic quantization noise on video quality," *Proceedings of the SPIE - Human Vision and Electronic Imaging*, San Jose, CA, 1999.
- [12] Y. C. Chang and D. G. Messerschmitt, "Delay cognizant video coding," *Proceedings of International Conference on Networking and Multimedia*, Kaohsiung, Taiwan, pp. 110-117, 1996.
- [13] Y. C. Chang and D. G. Messerschmitt, "Segmentation and compression of video for delay-flow multimedia networks," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, Seattle, WA, 1998.
- [14] Y. C. Chang, T. Carney, S. A. Klein, D. G. Messerschmitt, and A. Zakhor, "Effects of temporal jitter on video quality: assessment using psychophysical methods," *Proceedings of the SPIE - Human Vision and Electronic Imaging*, San Jose, CA, 1998.
- [15] Y. C. Chang and D. G. Messerschmitt, "Improving network video quality with delay cognizant video coding," *Proceedings of IEEE International Conference On Image Processing*, Chicago, IL, 1998.
- [16] Y. C. Chang, D. Tse and D. G. Messerschmitt, "Multimedia CDMA wireless network design: the link layer perspective," submitted to *IEEE International Conference on Communications*, 1999.
- [17] Y. C. Chang, T. Carney, S. A. Klein, and D. G. Messerschmitt, "Delay cognizant video coding: architecture, applications and quality evaluation," submitted to *IEEE Transactions on Image Processing*.
- [18] R. J. Clarke, *Digital compression of still images and video*, published by Academic Press, 1995.

- [19] T. M. Cover and J. A. Thomas, *Elements of information theory*, published by John Wiley & Sons, 1991.
- [20] S. Daly, "The visible differences predictor: an algorithm for the assessment of image fidelity," *Digital Images and Human Vision*, A. B. Watson, ed. MIT Press, 1993.
- [21] A. I. Elwalid and D. Mitra, "Effective bandwidth of general Markovian traffic sources and admission control in high-speed networks," *IEEE/ACM Trans. Networking*, vol. 1, no. 3, pp. 329-343, 1993.
- [22] K. Feher, *Wireless digital communications*, published by Prentice Hall, 1995.
- [23] J. Feng, K. T. Lo, and H. Mehrpour, "Error concealment for MPEG video transmissions," *IEEE Transactions on Consumer Electronics*, vol. 43, no. 2, pp. 183-7, May 1997.
- [24] A. Fox and E. A. Brewer, "Reducing WWW latency and bandwidth requirements by real-time distillation," *Computer Networks and ISDN Systems*, vol. 28, no. 7-11, pp. 1445-5, May 1996.
- [25] M. R. Garey, D. S. Johnson, *Computers and intractability: a guide to the theory of NP-completeness*, published by Bell Telephone Laboratories, 1979.
- [26] B. Girod, "What's wrong with mean-squared error?" *Digital Images and Human Vision*, A. B. Watson, ed. MIT Press, 1993.
- [27] S. Glisic and B. Vucetic, *Spread spectrum CDMA systems for wireless communications*, published by Artech House, 1997.

- [28] R. Guerin, H. Almadi, and M. Naghshineh, "Equivalent bandwidth and its application to bandwidth allocation in high-speed networks," *IEEE Journal on Selected Areas in Communication*, vol. 9, no. 7, pp. 968-981, 1991.
- [29] R. Han, "Progressively reliable packet delivery for interactive wireless multimedia," Ph.D. dissertation, University of California at Berkeley, 1997.
- [30] S. Hara and R. Prasad, "DS-CDMA, MC-CDMA, and MT-CDMA for mobile multimedia communications," *Proceedings of IEEE Vehicular Technology Conference*, Atlanta, GA, 1996.
- [31] H. Hashemi, "The indoor radio propagation channel," *Proceedings of the IEEE*, vol. 81, no. 7, pp. 943-68, July 1993.
- [32] P. Haskell, D. G. Messerschmitt and L. C. Yun, "Architecture principles for multimedia networks," *Wireless Communications: Signal Processing Perspectives*, H. V. Poor and G. W. Wornell, Ed., Prentice Hall, 1998.
- [33] P. Haskell and D. G. Messerschmitt, "Resynchronization of motion compensated video affected by ATM cell loss," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, San Francisco, CA, vol. 3, pp. 45-8, May 1992.
- [34] C. L. I, C. A. Webb, H. C. Huang, S. Brink, S. Nanda, and R. Gitlin, "IS-95 enhancements for multimedia services," *Bell Labs Technical Journal*, pp.60-87, Autumn 1996.
- [35] A. K. Jain, *Fundamentals of digital image processing*, published by Prentice Hall, 1989.
- [36] G. Kesidis, J. Walrand, and C. S. Chang, "Effective bandwidth for multi-class Markov fluids and other ATM sources," *IEEE/ACM Trans. Networking*, vol. 1, no. 4, pp. 424-28, Aug. 1993.

- [37] G. C. Kessler, *ISDN: concepts, facilities and services*, published by McGraw-Hill, 1993.
- [38] J. B. Kim and M. L. Honig, "Outage probability of multi-code DS-CDMA with linear interference suppression," to appear in *IEEE MILCOM*.
- [39] S. A. Klein, "Image quality and image compression: a psychophysicist's viewpoint," *Digital Images and Human Vision*, A. B. Watson, ed. MIT Press, 1993.
- [40] L. Kleinrock, *Queueing systems*, published by John Wiley & Sons, 1975.
- [41] T. J. Kostas, et al., "Real-time voice over packet-switched networks," *IEEE Network Magazine*, pp. 18-27, Jan. 1998.
- [42] C. Lambrecht, "A working spatio-temporal model of the human visual system for image restoration and quality assessment applications," *Proceedings of IEEE Int. Conf. On Acoustics, Speech, and Signal Processing*, Atlanta, GA, pp. 2291-4, 1996.
- [43] C. Lambrecht and O. Verscheure, "Perceptual quality measure using a spatio-temporal model of the human visual system," *Proceedings of the SPIE* vol. 2668, San Jose, CA, pp.450-61, 1996.
- [44] W. S. Lee, M. R. Pickering, M. R. Frater, and J. F. Arnold, "Error resilience in video and multiplexing layers for very low bit-rate video coding systems," *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 9, pp.1764-74, Dec. 1997.
- [45] Wei-Yi Li, "Agent-augmented network signaling for call setup," Ph.D. dissertation, University of California at Berkeley, 1998.

- [46] H. Liu, M. El Zarki, "Adaptive source rate control for real-time wireless video transmission," *Mobile Networks and Applications*, vol. 3, no. 1, pp. 49-60, 1998.
- [47] Z. Liu, M. J. Karol, M. E. Zarki, and K. Y. Eng, "A demand-assignment access control for multi-code DS-CDMA wireless packet networks," *IEEE INFOCOM: The Conference on Computer Communications*, San Francisco, CA, 1996.
- [48] J. Lubin, "The use of psychophysical data and models in the analysis of display system performance," *Digital Images and Human Vision*, A. B. Watson, ed. MIT Press, 1993.
- [49] T. H. Meng, "Low-power wireless video systems," *IEEE Communications Magazine*, vol. 36, no. 6, pp. 130-6, June 1998.
- [50] J. L. Mitchell, W. B. Pennebaker, C. E. Fogg, and D. J. LeGall, *MPEG video compression standard*, Chapman & Hall, 1997.
- [51] S. Moshavi, "Multi-user detection for DS-CDMA communications," *IEEE Communications Magazine*, pp. 124-36, Oct. 1996.
- [52] J. M. Moura, R. S. Jasinschi, H. Shiojiri, and J. Lin, "Retrieving quality video across heterogeneous networks: video over wireless," *IEEE Personal Communications*, vol. 3, no. 1, pp. 44-54, Feb. 1996.
- [53] S. Oh and K. M. Wasserman, "Adaptive resource management for DS-CDMA networks subject to energy constraints," *IEEE INFOCOM: The Conference on Computer Communications*, San Francisco, CA, 1998.
- [54] J. Ohm, "Advanced packet-video coding based on layered VQ and SBC techniques," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 3, no. 3, pp. 208-21, June 1993.

- [55] T. Ojanpera and R. Prasad, "An overview of air interface multiple access for IMT-2000/UMTS," *IEEE Communications Magazine*, vol. 36, no. 9, pp. 82-6, 91-5, Sept. 1998.
- [56] V. K. Paulrajan and J. A. Roberts, "Capacity of a CDMA cellular system with variable user data rates," *IEEE GLOBECOM*, London, United Kingdom, 1996.
- [57] R. Prasad, *CDMA for wireless personal communications*, published by Artech House, 1996.
- [58] K. Ramchandran, A. Ortega, K. Uz and M. Vetterli, "Multiresolution broadcast for digital HDTV using joint source/channel coding," *IEEE Journal on Selected Areas in Communications*, vol. 11, no. 1, pp. 6-22, Jan. 1993.
- [59] J. Reason, A. Lao, D. G. Messerschmitt, "Asynchronous video coding for wireless transport," *IEEE Workshop on Mobile Computing and Applications*, Dec. 1994.
- [60] J. Reason, L. C. Yun, A. Lao, D. G. Messerschmitt, "Asynchronous video: coordinated video coding and transport for heterogeneous networks with wireless access," *Mobile Computing*, H. F. Korth and T. Imielinski, editors, Kluwer Academic Press, 1995.
- [61] M. Schwartz, *Telecommunication networks: protocols, modeling and analysis*, published by Addison-Wesley, 1987.
- [62] J. R. Smith, R. Mohan, and C. S. Li, "Transcoding Internet content for heterogeneous client devices," *Proceedings of the 1998 IEEE International Symposium on Circuits and Systems*, vol. 3, p. 599-602, Monterey, CA, 1998.

- [63] W. Stalling, "IPv6: the new Internet Protocol," *IEEE Communications Magazine*, vol. 34, no. 7, pp. 96-108, July 1996.
- [64] E. Steinbach, N. Farber, and B. Girod, "Standard compatible extension of H. 263 for robust video transmission in mobile environments," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, no. 6, pp. 872-81, Dec. 1997.
- [65] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop ratio networks," *IEEE Transactions on Automatic Control*, vol. 37, no. 12, pp. 1936-48, Dec. 1992.
- [66] D. Taubman and A. Zakhor, "Multirate 3D subband coding of video," *IEEE Transactions on Image Processing*, vol. 3, no. 5, pp. 572-88, Sep. 1994.
- [67] J. Y. Tham, S. Ranganath, and A. A. Kassim, "Highly scalable wavelet-based video codec for very low bit-rate environment," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 1, pp.12-20, Jan. 1998.
- [68] D. N. C. Tse, R. G. Gallager, and J. N. Tsitsiklis, "Statistical multiplexing of multiple time-scale Markov streams," *IEEE Journal on Selected Areas in Communication*, vol. 13, no. 6, pp. 1028-1038, 1995.
- [69] D. Tse and S. Hanly, "Effective bandwidths in wireless networks with multiuser receivers," *IEEE INFOCOM: The Conference on Computer Communications*, San Francisco, CA, 1998.
- [70] D. Tse and S. Hanly, "Linear multiuser receivers: effective interference, effective bandwidth, and capacity," *to appear IEEE Trans. on Information Theory*.

- [71] J. Uddenfeldt, "Digital cellular-its roots and its future," *Proceedings of the IEEE*, vol. 86, no. 7, pp. 1319-24, July 1998.
- [72] P. Viswanath, V. Anantharam and D. Tse, "Optimal sequences, power control and capacity of spread-spectrum systems with multiuser linear receivers", submitted to *IEEE Transactions on Information Theory*, Jan. 1998. (Revised July, 1998.)
- [73] A. J. Viterbi, *CDMA - principles of spread spectrum communication*, published by Addison-Wesley, 1995.
- [74] S. C. Yang, *CDMA RF system engineering*, published by Artech House, 1998.
- [75] G. S. Yu, M. Liu, and M. W. Marcellin, "POCS-based error concealment for packet video using multiframe overlap information," *IEEE Transactions on Circuits and Systems for Video Technology*, vol.8, no.4, pp. 422-34, Aug. 1998.
- [76] L. C. Yun, "Transport for multimedia in wireless networks," Ph.D dissertation, University of California at Berkeley, 1995.
- [77] L. C. Yun and D. G. Messerschmitt, "Digital video in a fading interference wireless environment," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, pp.1069-1072, Atlanta, GA, 1996.
- [78] L. C. Yun and D. G. Messerschmitt, "Variable quality of service in CDMA systems by statistical power control," *IEEE International Conference on Communications*, Seattle, WA, 1995.
- [79] B. E. Wampold and C. J. Drew, *Theory and application of statistics*, published by McGraw-Hill, 1990.

- [80] B. Wandell, *Foundations of vision*, published by Sinauer Associates, 1995.
- [81] Y. Wang and Q. Zhu, "Error control and concealment for video communication: a review," *Proceedings of the IEEE*, vol. 86, no. 5, pp. 974-97, May 1998.
- [82] A. B. Watson, "Image data compression having minimum perceptual error," US Patent No. 5,629,780.
- [83] M. H. Willebeek-LeMair, Z. Y. Shae, and Y. C. Chang, "Robust H. 263 video coding for transmission over the Internet," *IEEE INFOCOM: The Conference on Computer Communications*, vol. 1, pp.225-32, San Francisco, CA, 1998.
- [84] G. Wu, A. Jalali, and P. Mermelstein, "On channel model parameters for microcellular CDMA systems," *IEEE Transactions on Vehicular Technology*, vol. 44, no. 3, pp. 706-11, Aug. 1995.
- [85] J. Wu and R. Kohno, "Performance evaluation of wireless multimedia CDMA networks using adaptive transmission control," *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 9, pp. 1688-97, Dec. 1996.
- [86] J. Zhang, M. R. Frater, J. F. Arnold, and T. M. Percival, "MPEG 2 video services for wireless ATM networks," *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 1, pp. 19-28, Jan. 1997.