Adaptive rate control for video compression and transmission over fading channel in wireless networks

Wei, Jun

2005

Wei, J. (2005). Adaptive rate control for video compression and transmission over fading channel in wireless networks. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/3725>

<https://doi.org/10.32657/10356/3725>

Nanyang Technological University

Downloaded on 26 Apr 2023 22:20:45 SGT

Adaptive Rate Control for Video Compression and Transmission over Fading Channel in Wireless Networks

Wei Jun

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University
in fulfillment of the requirement for the degree of
Doctor of Philosophy

2005

TK
6680.5
W415
2005

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Professor Soong Boon Hee, for accepting me as his student and giving me the opportunity to research on the rate control for video transmission. I am especially thankful to him for his understanding, valuable advice and patient guidance throughout the research. His painstaking effort in correcting my paper and thesis is also very much appreciated. Without his professional and dedicated instructions and patience, this thesis would not have been possible.

I am also grateful to Dr. Li Zhengguo of Institute for InfoComm Research, who shared his knowledge and ideas with me, provided me with many useful references and gave me many valuable advices for my research. I appreciate the many discussions with him, in which he is always full of unique thoughts.

Finally, I would like to express the deepest gratitude to my wife and my parent, for their help and encouragement throughout the course of my research.

Contents

Acknowledgments	i
Table of Contents	ii
Summary	vi
List of Figures	viii
List of Tables	xv
List of Abbreviations	xvii
1 Introduction	1
1.1 Motivation	1
1.2 Objective	5
1.3 Contribution of the thesis	6
1.4 Organization of the thesis	8
2 Rate Control for Video Transmission Over Wireless Network: An Overview	11
2.1 Framework of video communication system	11
2.2 Challenges of wireless video communication system	16
2.2.1 Data transmission rate	16
2.2.2 Channel errors of wireless networks	16

- 2.2.3 Variable payload data rate in wireless network 17
 - 2.2.4 Bursty characteristic of video transmission 17
 - 2.2.5 Sensitivity of video bitstream to bit error rate 19
 - 2.2.6 Challenges of wireless video communication system 20
 - 2.3 Error control for wireless video communication system 21
 - 2.3.1 Unequal error protection 23
 - 2.4 Rate control schemes for video communication system 25
 - 2.4.1 Joint source channel rate control 27
 - 2.4.2 Rate control for the video encoder alone 31
 - 2.5 Tools 38
 - 2.5.1 Measurement of distortion 38
 - 2.5.2 Linear regression method 40
- 3 Adaptive Joint Source-Channel Rate Allocation For Video Transmission 42**
 - 3.1 Introduction 42
 - 3.2 UEP-based video communication system 45
 - 3.2.1 Joint source channel video coding 45
 - 3.2.2 Error control 47
 - 3.3 Problem formulation for rate control in UEP system 50
 - 3.3.1 End-to-end rate distortion model 50
 - 3.3.2 Analysis of end-to-end rate distortion 54
 - 3.4 Adaptive joint source channel rate allocation based on BER control 58
 - 3.4.1 Adaptive rate control for the video encoder alone based on quadratic model 58
 - 3.4.2 Estimation of output bitstream ratio with sliding window 60
 - 3.4.3 Estimation of target symbol error rate 63
 - 3.4.4 Adaptive source channel rate allocation using decoded BER control component 68

3.4.5	Computational complexity	71
3.5	Simulation results	71
3.5.1	Results for constant channel condition	73
3.5.2	Results for variable channel condition	76
3.6	Conclusions	79
4	A New Rate Model Based on Multiple Logarithmic Functions	81
4.1	Introduction	81
4.2	Image distribution analysis after ME/MC	82
4.2.1	Motion estimation and compensation	82
4.2.2	Entropy bits with known distribution	84
4.3	Proposed logarithmic rate model	86
4.3.1	Logarithmic rate model	87
4.3.2	Sliding windows	88
4.3.3	Boundary optimization	89
4.4	Simulation results	92
4.4.1	Results of fixed quantization parameter	92
4.4.2	Results of variable quantization parameter	94
4.5	Conclusions	96
5	Fast Pre-encoding Component and Its Application	97
5.1	Introduction	97
5.2	Conventional rate control structure	98
5.3	Challenge of the new emerging video encoder	100
5.4	Pre-encoding structure for new video encoder	108
5.4.1	Analysis of video encoder	108
5.4.2	New encoding structure based on fast pre-encoding	110
5.4.3	Application of MAD prediction method based on FPE component	115

5.5	Simulation Results	118
5.5.1	Performance of MAD prediction	118
5.5.2	Performance of rate control scheme	119
5.6	Conclusion	123
6	Adaptive Rate Control for H.264 Based on Logarithmic Rate Model	124
6.1	Introduction	124
6.2	Review of standard rate control of H.264 system	126
6.3	Performance improved rate control for H.264	127
6.3.1	Quantization parameter derivation based on logarithmic rate model	127
6.3.2	MAD estimation with fast pre-encoding component	132
6.3.3	Quantizer calculation of starting frame	133
6.3.4	Estimation of number of header bits	136
6.3.5	Adaptive rate control based on optimized components	139
6.4	Simulation results	144
6.4.1	Results for stored video	145
6.4.2	Results for low delay communication	148
6.5	Conclusions	153
7	Conclusions and Recommendations	155
7.1	Conclusions	155
7.2	Recommendations for further research	158
	Author's Publications	160

Summary

The performance of a video communication system is usually constrained by the transmission channel which has limited bandwidth and other kinds of impairments. In practical situation, the video communication system does not optimally adapt to time-varying wireless channels due to an inaccurate rate control of the video encoder and channel encoder, which results in inefficient use of the valuable wireless channel bandwidth. At the same time, the video encoder has limited buffer size to store compressed video bitstream, the video system needs to prevent video encoder buffer from overflow and underflow.

The objective of this thesis is to develop new adaptive rate control schemes for the video encoder and channel encoder so that a better picture quality is achieved as compared to the existing rate control schemes. A novel joint source channel rate control scheme is proposed in this thesis, in which the bit error rate of the video bitstream at the video decoder is used as a control component to derive bit rate for the video encoder and channel encoder. The video bitstreams are protected unequally, so that the most important bitstreams can be delivered to the video decoder correctly. The joint source channel rate control scheme is adaptive to time-varying channel as the bit error rate control component is adaptive to the channel conditions.

In addition to joint source channel rate control scheme, we also want to find novel approaches to exploit inherent characteristics of block-based hybrid video encoder and de-

velop new models to represent relationships between the video encoder parameters and the number of bits produced by the video encoder. A new rate model, which is based on multiple logarithmic functions, is proposed in this thesis. The model's coefficients are updated from previously encoded frames so that the rate model adapts to the input video sequence quickly and effectively. The boundaries of the logarithmic functions are also adapted to characteristics of the encoded video sequence.

Finally, this thesis designs an optimal rate control scheme for small delay H.264 video communication system as H.264 achieves a higher compression rate than previous video coding standards as well as brings challenges to implement rate control algorithms. A pre-encoding procedure is proposed to obtain the knowledge of macroblock. The rate control scheme performs rate allocation at frame level and macroblock, and it is based on the measurement of the encoder buffer level, the channel transmission rate and knowledge of macroblock. The proposed rate control scheme adopts a new technique to compute the quantization parameter of the starting frame of a group of picture. The logarithmic rate model studied previously is used to compute the quantization parameter of the rest frames of the group of picture.

List of Figures

2.1	Basic components of a video communication system, including video capture device, video encoder, channel encoder, modulator, demodulator, channel decoder, video decoder and video display device.	11
2.2	A typical video encoder structure. The encoding process involves motion estimation, motion compensation, discrete cosine transform, quantization, variable length coding and so on.	13
2.3	The number of bits per frame for video test sequence <i>Foreman</i> . The video sequence is encoded by H.264 video encoder at a constant quantization parameter 30.	18
2.4	Subjective picture quality comparison for video test sequence <i>Foreman</i> , 2.4(a): the 59th frame; 2.4(b): the 64th frame. The video sequence is encoded by H.264 video encoder at a constant quantization parameter 30.	19
2.5	subjective quality, PSNR and number of bits comparison for video test sequence <i>Foreman</i> . (a): PSNR=57.86 (dB), number of bits=170264 (bits); (b): PSNR=28.72 (dB), number of bits=6760 (bits)	39
3.1	Wireless video system with feedback. The compressed bitstream is divided into several sub-bitstream, and each sub-bitstream has different priority. .	45

3.2 Data partitioning for MPEG-4 video packet. Each MPEG-4 video packet consists of a re-synchronization marker, which indicates the start of a new video packet. The video packets also includes quantization parameter, motion vectors, and texture (luminance and chrominance) coefficients. 46

3.3 PSNR comparison with RS code $(K_h, 255)$. Video source *Container* is encoded by a MPEG-4 encoder. $u_c = 128kbps$. K_l is fixed to 251. 56

3.4 PSNR comparison with RS code $(K_l, 255)$. Video source *Foreman* is encoded by a MPEG-4 encoder. $u_c = 256kbps$. K_h is fixed to 201. 57

3.5 Comparison results between $\tilde{\varphi}$ and $\varphi = \frac{L_h}{L_h+L_l}$ for *Foreman* QCIF. Fixed quantization parameters 20 is used in MPEG-4 video encoder. 62

3.6 Comparison results between $\tilde{\varphi}$ and $\varphi = \frac{L_h}{L_h+L_l}$ for *Foreman* QCIF. video is encoded with MPEG-4 encoder at a target bit rate $96kbps$, which produces variable quantization parameter for each frame. 63

3.7 PSNR of those decoded video frames versus frame index for various BER in sub-bitstream L . The video source is *News*. The encoded MPEG-4 video bitstream contains random bit errors in sub-bitstream L 64

3.8 Decoded video PSNR versus frame index for various BER in sub-bitstream H . The video source is *News*. The encoded MPEG-4 video bitstream contains random bit errors in sub-bitstream H 65

3.9 Iteratively searching process for P_{t_l} and P_{t_h} based on pilot channel and pilot slice. For each different frame, the pilot slice is protected with a set of different FEC code rates, which is derived from target SERs for sub-bitstream L and H 68

3.10 Simulation configuration to test the performance of adaptive source channel bit allocation algorithm. The priority-based video bitstream is arranged into error control frame for transmission. 72

3.11 PSNR comparison of the proposed ACBA scheme, BM, and rate control scheme based on fixed UEP. The Test sequence is Foreman, which is encoded with a frame rate $f = 30f/s$. $P_s = 10^{-3}$, $u_s = 128kbps$ 74

3.12 PSNR comparison of our proposed ACBA scheme, BM, and rate control scheme based on fixed UEP. Test sequence is *Foreman*, which is encoded with 30f/s and $u_s = 128kbps$ 75

3.13 PSNR comparison of our proposed ACBA scheme, BM, and the rate control scheme based on fixed UEP. Test sequence is Foreman, which is encoded with 30f/s. channel SER varies during transmission and $u_s = 128kbps$. . . 77

3.14 subjective results for frame 70 at QCIF resolution:3.14(a) BM, 3.14(b) our proposed ACBA scheme, and 3.14(c) rate control scheme based on fixed UEP. 78

4.1 Motion estimation and compensation: one MV is estimated for each block in the actual frame to be encoded. The MV points to a reference block of same size in a previously coded frame. The reference block is in a $N \times Z$ pixel search window. 83

4.2 Bit number comparison among original H.263 encoder output, logarithmic rate model, TMN8, quadratic model for continuous frames of 'foreman.qcif'. $QP = 3$ for all MBs in P-frames. 93

4.3 Bit number comparison among original H.263 encoder output, logarithmic rate model, TMN8, quadratic model for continuous frames of 'foreman.qcif'. $QP = 20$ for all MBs in P-frames. 93

4.4 Bit number comparison among original H.263 encoder output, fixed model, our method, TMN8, and quadratic method for continuous frames of 'foreman.qcif'. QP varies from MB to MB. 94

4.5 The boundary values of logarithmic rate model for continuous frames of 'foreman.qcif'. QP varies from MB to MB. The QP_L and QP_H are updated after encoding one frame. 95

5.1 Buffer feedback-based rate control for H.263, MPEG-4 video encoder. After coding each video frame, the buffer level is measured and fed back to the video encoder. 98

5.2 Basic coding structure of H.264. The RDO is used to select the best encoding mode, reference frame and motion vectors. At most 5 frames is kept in the frame stored buffer for ME/MC. 101

5.3 Bits comparison for simple method and complex method. The sequence name *Foreman*. The Sequence is encoded with H.264 video encoder. Frame rate $F_r = 30f/s$. Quantization parameter is 30 for all the frame. Reference number is 5. 103

5.4 PSNR comparison for simple method and complex method. The sequence name *Foreman*. The Sequence is encoded with H.264 video encoder. Frame rate $F_r = 30f/s$. Quantization parameter is 30 for all the frame. Reference number is 5 104

5.5 Encoding time comparison for for simple method and complex method. The sequence name *Foreman*. The Sequence is encoded with H.264 video encoder. Frame rate $F_r = 30f/s$. Quantization parameter is 30 for all the frame. Reference number is 5. 104

5.6 Simplified H.264 video encoder with rate control component. It is obvious that rate control block needs to know *MAD* or *MSE* to perform rate control and compute quantization parameter. On the other hand, the RDO block needs to know quantization parameter. 105

5.7 π_i vs. modes with various values of *QP*. The prediction modes are gathered after encoding 100 frames of each video sequence. The video sequences from top to bottom are: *Foreman*, *Carphone*, *Silent*, *Container*, *Mother and daughter*. The *Foreman*, *Carphone*, and *Silent* are encoded with RDO on. The *Container*, *Mother and daughter* are encoded without RDO. 111

5.8 $\pi_{i,i} = \text{Inter8X8}$ vs. QP for video sequences *Foreman*, *Carphone*, *Silent*, *Container*, *Mother and daughter*. The *Foreman*, *Carphone*, and *Silent* are encoded with RDO on. The *Container*, *Mother and daughter* are encoded without RDO. 112

5.9 $\pi_{i,i} = \text{Inter16X16}$, *Skip* vs. QP for video sequences *Foreman*, *Carphone*, *Silent*, *Container*, *Mother and daughter*. The *Foreman*, *Carphone*, and *Silent* are encoded with RDO on. The *Container*, *Mother and daughter* are encoded without RDO. 113

5.10 Proposed H.264 encoding structure with rate control based on FPE component. The FPE component is responsible for performing RDO using two PMs, i.e., *Inter16X16* and *Skip*, and deriving MAD information and number of overhead bits. 113

5.11 Comparison between the actual MAD values and the predicted MAD values obtained by using the FPE method and CBLP method, *Foreman*, frame index =4, bit rate= 256kbps, RDO is used. 119

5.12 Comparison between the actual MAD values and the predicted MAD values obtained by using the FPE method and CBLP method, *News*, frame index =3, bit rate= 56kbps, RDO is not used. 120

5.13 PSNR comparison between FPER and CBLPR. The video test sequence is *Foreman*, frame rate =30f/s, bit rate= 100kbps, RDO is on. 121

6.1 Verification of function form of the encoding rate (in bits/pixel) as a function of MAD and QP . The results are obtained by encoding 100 frames of the video sequence *Foreman* at 30f/s, using JVT's H.264 encoder JM6.1 [1]. After encoding a MB, we measure MAD of MB, chosen prediction mode and the number of bits required by luminance and chrominance. The curves plot the average rate (in bits/pixel) for each MAD. 129

6.2 QPs before bounding and QPs after bounding vs. MB index for the test sequence *Foreman*. The frame rate is 15f/s and the target bit rate is 96kbps. 130

6.3 The buffer level vs. frame index for test sequence *Silent*. JM6R rate control is employed. The frame rate is 30f/s. The target bit rate is 128kbps. Frame skip is not allowed. The video buffer size B_s is $R/F_r = 128000/30 = 4266$. 134

6.4 The comparison between the estimated number of bits obtained from the proposed model and actual number of bits per frame. The video sequences are encoded with distinct $QP = \{1, 4, 7, 10, 13, 16, 19, 22, 25, 28, 31, 34, 37, 40, 43, 46, 49, 51\}$. The test sequences are *Foreman*, *Silent*, *Container* and *Mother and Daughter*. Their frame rate are 30f/s. 135

6.5 The number of bits for MVs per MB vs. prediction mode. The test sequence *Foreman* is encoded by the H.264 encoder with constant $QPs = 10, 20, 35$. The frame rate is 15f/s. 137

6.6 The number of bits required by MVs per MB vs. MB index. The test sequence *Silent* is encoded by the H.264 encoder with constant $QP = 10$. The frame rate is 30f/s. 138

6.7 PSNR Comparison of JM6R and the proposed LR method. The video sequence is *Foreman*, total bit budget is $R * 100/F_r$, where $R = 128kbps$ and $F_r = 30f/s$ 146

6.8 The number of bits per frame of JM6R and the proposed LR algorithm. The video sequence is *Foreman*, total bit budget is $R*100/F_r$, where $R = 128kbps$ and $F_r = 30f/s$ 147

6.9 Buffer level after encoding each frame of JM6R and the proposed LR algorithm. The video sequence is *Foreman*, total bit budget is $R*100/F_r$, where $R = 128kbps$ and $F_r = 30f/s$ 147

6.10 The number of bits per frame comparison of JM6R and LR algorithm. The video sequence is *Foreman*, and target bit rate $R = 128kbps$, frame rate $F_r = 30f/s$, $M = R/F_r$ 149

6.11 Buffer level comparison of JM6R and and LR algorithm. The buffer level is measured after encoding each frame. The video sequence is *Foreman*, and target bit rate $R = 128kbps$, frame rate $F_r = 30f/s$, $M = R/F_r$ 150

6.12 PSNR comparison of JM6R and and LR algorithm. The video sequence is *Foreman*, and target bit rate $R = 128kbps$, frame rate $F_r = 30f/s$, $M = R/F_r$. 150

6.13 The number of bits per frame comparison of JM6R and the proposed LR algorithm. The video sequence is *Foreman*, bit rate $R = 96kbps$, frame rate $F_r = 10f/s$, $M = R/F_r$ 151

6.14 Buffer level comparison of JM6R and the proposed LR algorithm. The buffer level is measured after encoding each frame. The video sequence is *Foreman*, bit rate $R = 96kbps$, frame rate $F_r = 10f/s$, $M = R/F_r$ 152

6.15 PSNR comparison of JM6R and the proposed LR algorithm. The video sequence is *Foreman*, bit rate $R = 96kbps$, frame rate $F_r = 10f/s$, $M = R/F_r$. 152

List of Tables

2.1	Comparison of MPEG-1, MPEG-2, H.263 and H.264 video encoder. The typical applications, target bit rates and number of reference frames used for motion estimation are compared for the four video encoders.	14
3.1	Comparison of K_l and K_h which are computed by our proposed ACBA scheme, BM, and rate control scheme based on fixed UEP.	75
3.2	Variation of channel SER P_s during transmission period	76
3.3	Comparison of K_l and K_h which are computed by using our proposed ACBA scheme, BM, and the rate control scheme based on fixed UEP.	78
4.1	Comparison of average relative prediction error per frame among TMN8, quadratic model, fixed model, and logarithmic rate model. QP varies from MB to MB.	95
5.1	Summary of Index, PM names and their height and width in terms of pixels. For example, given index 4, the prediction mode of the MB is Inter16X16, therefore the MB is Inter-coding and ME/MC is done in 16×16 search window.	109
5.2	Summary of index, sub-PM names and their height and width in terms of pixels.	109

5.3 Configuration of the H.264 encoder. The number of reference frame for ME/MC is 1, and the search range in the reference frame is 32. The first frame is intra-coding, and the rest frames are inter-coding. 118

5.4 Summary of the test scheme names, video sequence names, frame rate, target bit rate, and RDO feature. 120

5.5 Comparison of the achieved bit rate between the proposed FPER rate control and CBLP-based rate control. A number of video test sequence is used. . . 121

5.6 PSNR comparison between the proposed rate control and CBLPR. 122

6.1 Quantization parameter and quantizer step size in H.264 codec. QP is quantization parameter, and Q is corresponding quantizer step size. Quantization parameter in H.264 ranges from 1 to 51. 131

6.2 The configuration parameters of H.264 video encoder. MV resolution is $1/4$ means that motion vectors may have the non-integral values, which are derived from original pixel values using bilinear interpolation. The entropy coding is Context-Adaptive Arithmetic Coding (CABAC). The number of reference frame for ME/MC is 1, and the search range in the reference frame is 32. The first frame is intra-coding, and the rest frames are inter-coding. 144

6.3 Description of the experimental configuration: Names, video sequence, frame rate F_r and target bit rate R . Each name corresponds to a given video sequence, frame rate and target bit rate. 145

6.4 Comparison of achieved bit rate and average PSNR for JM6R rate control and the proposed LR. 153

List of Abbreviations

JPEG	Joint Photographic Experts Group
UMTS	Universal Mobile Telecommunications System
JVT	Joint Video Team
LAN	Local Area Network
ATM	Asynchronous Transfer Mode
UEP	Unequal Error Protection
MB	MacroBlock
GOP	Group of Pictures
PSNR	Peak Signal Noise Ratio
MAD	Minimum Absolute Difference
QCIF	Quarter Common Intermediate Format
Kbits	kilo bits
Kbps	kilo bits per second
Mbits	mega bits
Mbps	mega bits per second
fps	frame per second
DCT	Discrete Cosine Transform
ME	Motion Estimation
MC	Motion Compensation
AVC	Advanced Video Coding
FEC	Forward Error Correction
WCDMA	Wideband Code Division Multiple Access
VLC	Variable Length Coding
BER	Bit Error Rate
MV	Motion Vector
POCS	Projection Onto Convex Sets

HEC	Header Extension Codes
ARQ	Automatic Repeat reQuest
BCH	Bose Chaudhuri Hochquenghem
EEP	Equal Error Protection
QoS	Quality of Sevice
MDC	Multiple Description Coding
LC	Layer Coding
i.i.d.	independent identically distributed
JSCC	Joint Source Channel Coding
MSE	Mean Square Error
RD	Rate Distortion
VP	Video Packet
RS	Reed-Solomon
SER	Symbol Error Rate
BSC	Binary Symmetrical Channel
CBR	Constant Bit Rate
VBR	Variable Bit Rate
EC	Error Control
EREC	Error-Resilent Entropy Code
BM	Block Matach
TMN	Test Model Near term
VCL	Video Coding Layer
NAL	Network Adaptation Layer
SAD	Sum of Absolute Difference
PM	Prediction Mode
CAVLC	Context-Adaptive Variable Length Coding
CABAC	Context-Adaptive Arithmetic Coding
SSD	Sum of Squared Differences
RDO	Rate Distortion Optimization

Chapter 1

Introduction

1.1 Motivation

Compression of visual information such as image and video can significantly improve the utilization efficiency of a limited network bandwidth and storage space. For example, an uncompressed image of 1024×1024 pixels with 24 bits per pixel requires a 3Mbyte space. But by using the Joint Photographic Experts Group (JPEG) [2] compression technique, the image size can be compressed to as small as 60kbyte. The fast development of video compression technology enables the wide deployment of digital video applications such as internet video streaming, internet videophone and video conference. However, people are dreaming that they can watch video any time any where with any device, and the increasing number of mobile users brings an increasing demand for mobile video applications. With the introduction of new video compression algorithms such as MPEG-4 [3] in which video can be compressed to a bit rate as low as 64kbps or even lower, and evolution of the wireless network which provides a higher data transmission rate, wireless video applications are becoming a reality.

Chip manufactures, network infrastructure operators together with video coding standard

committees are now working side by side to advance wireless video technologies at a rapid pace. With development of system-on-chip technology, multimedia support for mobile phones have been emerging in the last few years and will become pervasive in the next few years. Network infrastructure operators are upgrading their wireless networks to Universal Mobile Telecommunication System (UMTS) [4] standard conformance networks which provide a higher data transmission rate up to 2048kbps and more flexible data services than previous GSM networks. MPEG's video group and ITU-T's Video Coding Experts Group (VCEG) have established a Joint Video Team (JVT) to create a single technical design for a forthcoming ITU-T Recommendation, ITU-T H.264 [5], and for a new part of the MPEG-4 standard, MPEG-4 part 10, which achieves very high compression efficiency and friendly network interface.

Despite the past efforts researchers have made, the data transmission rate of the wireless network is still not high enough as compared with the wireline network whose data rate is up to 1Gbps for optical fibers local area networks (LANs) [6] or even 2.5Gbps for Asynchronous Transfer Mode (ATM) networks [7]. Furthermore, wireless channels are much more hostile than wired channels due to its adverse time-varying features, multipath propagations and interferences from neighbouring cells. Therefore, more powerful error control techniques are required in wireless channels to combat the channel noise at a cost of decreased bandwidth efficiency. The increasing demand of wireless video applications presents a lot of technical challenges which remain to be addressed. These challenges become even more important given the wide deployment of multimedia supported mobile phones that now permeate into people's daily life everywhere and make our dependency on them ever greater.

We chose to study one of the wireless video streaming subjects in great detail, which is called adaptive joint source channel rate control. The adaptive joint source channel rate

control in this thesis is divided into two aspects. One is the joint source channel rate allocation, and the other is the adaptive rate control for video encoder alone. This research topic stems from the fact that the real-world video communication system does not optimally adapt to time-varying wireless channels due to an inaccurate rate control of the video encoder and channel encoder, which results in inefficient use of the valuable wireless channel bandwidth. Instead of increasing the compression ratio of the video encoder, or using more powerful error correcting codes, we propose a new method to allocate bit rate between the video encoder and channel encoder. The objective is to, given an overall transmission rate, minimize the overall end-to-end distortion which includes source distortion caused by quantization, and channel distortion caused by channel noise. The adaptive rate control for video encoder alone is also studied as the rate allocation between the video encoder and channel encoder results in a variable bit rate allocated to the video encoder, and an adaptive rate control is required to maintain the performance of the video encoder.

In wireless communication systems with limited bandwidth and poor channel conditions, adaptive joint source channel rate control is an approach to make use of the channel bandwidth efficiently. Compared with fixed rate allocation schemes, the adaptive joint source channel rate control scheme is more appropriate to be used to protect video bitstream from transmission errors as it adds variable amount of redundant bits to information bits for error protection depending on the channel condition, so that the bandwidth can be exploited efficiently. This technique is especially important for video streaming over the next generation wireless network which provides variable data transmission rates and flexible channel coding schemes.

The adaptive rate control for video encoder alone also plays a very important role in video communication systems in that it can not only increase the robustness of the encoded video bitstream but also prevent buffers in the encoder from overflow and underflow. By using

the joint source channel rate allocation, the bit rate allocated to the video encoder depends on the channel condition, error correcting capability of the channel code, and so on. It is therefore important to adopt an adaptive rate control scheme for the video encoder to achieve a given output bit rate while minimizing the source distortion. The encoding parameters such as quantization parameters, encoding frame rate, and so on are determined by the adaptive rate control scheme.

Although rate control for video signals has been studied intensively by many researchers in the past, and a number of joint source channel rate control schemes as well as rate control schemes for video encoder alone have been proposed, no one can safely vouch the end of this active research topic as new findings and results are still being reported frequently in the open literature. Especially after the introduction of new video compression technologies, most of the existing rate control solutions are required to be re-studied carefully and therefore new rate control algorithms and ideas are needed urgently.

To study the performance of the adaptive rate control scheme, different compression schemes have to be considered. The interest in video compression has been shown in international standardization efforts, such as H.263, MPEG-4 and the new emerging H.264. The video compression schemes are normally lossy, which is different from data communication compression schemes that are lossless. In lossy compression, the decoded signal is an approximation to the original signal fed into the encoder. In this thesis, lossy video encoder which employs block-based hybrid video coding approach is considered.

We have thus far explained the main motivations for us to study the adaptive rate control scheme. Here is a list of demands and requirements.

- **Channel constraints** A channel has a limited bandwidth and time-varying features. The situation calls for compressing video signals efficiently and adding redundancy

intelligently.

- **Buffer constraints** When transmitting the compressed video signals through the network, the video bitstreams are firstly put into a buffer whose size is limited. Unlimited buffer size leads to excessive end-to-end delay.
- **Other constraints** It is possible that other constraints operating directly or indirectly on decoded video picture quality, e.g., an inaccurate rate model results in an inefficient use of the channel bandwidth, which degrades the decoded picture quality

1.2 Objective

The main objective of our research is to achieve efficiency improved rate control schemes, as compared with existing methods, by adapting the video encoder and channel encoder to inherent characteristics of video sequences and channel conditions which are derived from long-term statistics or well constructed channel models. Our efforts are focused on the following issues:

- To develop a new adaptive bit allocation scheme between the video encoder and channel encoder to achieve a better performance as compared to the existing rate allocation schemes, or to achieve a comparable performance but with less computational complexity so that the scheme can be used in real time video communication system.
- To find novel approaches to exploit inherent characteristics of block-based hybrid video encoder and develop new models to represent relationships between the video encoder parameters and the number of bits produced by the video encoder by using those parameters.

- To investigate how to implement an adaptive rate control scheme in the new emerging video standard H.264 which achieves a higher compression ratio than previous video coding standards as well as brings new technical challenges. The H.264 standard conformance video encoder is studied because it is a standard which is accepted by both ITU and ISO and achieves the largest compression efficiency among all the existing video standards, and it is believed to dominate the video coder market in the future.

1.3 Contribution of the thesis

The main contributions of the thesis are as follows:

1. A new joint source channel rate control scheme for unequal error protection (UEP)-based video communication system is proposed, in which the bit rate of the video encoder and channel encoder are jointly determined by using the decoded BER as a control component together with quadratic rate model and parameter estimation techniques. The new rate control scheme adapts the video encoder and channel encoder to time-varying channels, so that the performance of the wireless video streaming system can be improved as compared to that of the system which employs the existing rate allocation schemes. The proposed rate control algorithm achieves a similar performance to that obtained by using the other rate allocation method, but with much less computational complexity and streaming delay, so that it can be applied to real-time video streaming.
2. Inspired by the facts that the existing rate models for the video encoder such as the quadratic rate model are not accurate enough, a new rate model is proposed which fits the original rate curve accurately, and it can be used in any block based hybrid video encoder. The new rate model is based on multiple logarithmic functions whose coefficients are updated from previously encoded frames so that the model adapts to

the input video sequence quickly and effectively. The boundaries of the logarithmic functions are also adapted to characteristics of the encoded video sequence. Simulation results show that the proposed model leads to significant accuracy improvement as compared with the existing rate models in terms of the bit number estimation error.

3. A novel encoding structure is proposed to obtain the knowledge of macroblocks (MB) such as MB's complexity, header information and so on. A pre-encoding procedure is proposed to obtain the knowledge of MB and provide it to the video encoder. Furthermore, the pre-encoding structure can be taken as a solution to the the problem of how to perform a rate control effectively for H.264 encoder. In the thesis, we will show that the obtained knowledge of MB by using our proposed pre-encoding procedure is more accurate and useful than other existing methods.
4. Because most of the existing rate control schemes applied to the video encoder cannot be used by H.264 encoder, a hybrid rate control scheme is proposed for the low-delay H.264 video communication system, which takes advantages of the proposed logarithmic rate model and pre-encoding procedure. In the hybrid rate control scheme, a novel method is proposed to calculate the quantizer step size of the starting frame of a group of picture (GOP). It allocates bits intelligently among macroblocks and provides an accurate rate control for the H.264 encoder. The proposed rate control scheme gives rise to a rather constant buffer level and reduces the number of skipped frames, and it leads to significant improvement as compared to the standard rate control scheme of H.264 in terms of peak signal noise ratio (PSNR).

1.4 Organization of the thesis

The thesis is organized as follows:

Chapter 2 The fundamental knowledge of video communication systems is addressed for better understanding of rate control algorithms. The function of rate control is discussed and the existing rate control schemes are classified and reviewed. Finally, some useful tools and mathematical background are introduced.

Chapter 3 Different techniques which perform bit rate allocation between the video encoder and channel encoder are reviewed. The classical rate-distortion framework is then used to formulate an end-to-end distortion for video transmission systems. After identifying independent key parameters of the distortion formula, the decoded BER is proposed to be used as a control component in the rate allocation process. The bit rate allocated to the video encoder and channel encoder are jointly optimized by using the BER control component. The rate allocation scheme between the video encoder and channel encoder is adapted to time-varying channel as the BER control component is adaptive to the channel conditions. Furthermore, each class of the video bitstream is protected appropriately by using an adaptive UEP, so that the channel bandwidth can be used efficiently. The proposed rate allocation scheme is shown to achieve a significant improvement over the schemes with fixed UEP.

Chapter 4 The quadratic rate model is not accurate enough to implement a good rate control. In this chapter, after reviewing several existing rate models and analyzing the image data statistics, a new logarithmic rate model for block-based hybrid video encoder is studied. The proposed model fits the actual rate curve by combining several logarithmic functions, and the boundary of each function is adaptive to the statistics of the input video sequence. The coefficients including boundary values of the new rate model are updated

from previously encoded frames. Experimental results show that the new rate model provides a more accurate bit number estimation than the quadratic model and TMN8 rate model, which are the two most widely used rate models.

Chapter 5 This chapter proposes a new pre-encoding process for the emerging H.264 video encoder. The H.264 encoder is first introduced, and the new technologies employed in H.264 encoder are described. In order to solve the dilemma of how to implement a conventional rate control scheme, e.g., TMN8 rate control, in the new H.264 encoder, the pre-encoding procedure is proposed to get the knowledge of macroblocks, such as complexity, header information and so on before the encoding process. Simulation results show that the new encoding structure leads to better performance as compared with the traditional encoding structure in terms of peak signal to noise ratio.

Chapter 6 This chapter describes a new rate control algorithm for the low-delay H.264 video communication system. The proposed rate control algorithm uses the logarithmic rate model proposed in chapter 4, which results in an accurate rate control for the video encoder, and the issue of how to calculate the quantizer step size by using logarithmic rate model is addressed in detail. The fast pre-encoding structure proposed in chapter 5 is employed to estimate the complexity of MB, which is denoted by minimum absolute difference (MAD). Computation of the number of the overhead bits in a MB, which incorporate specific statistics information of previously encoded frames, is also studied. The problem of how to calculate the quantizer step size of the starting frame of a GOP is also solved in this chapter. These methods mentioned above formulate the new hybrid rate control scheme. Simulation results show that the new scheme is efficient and adaptive to inherent characteristics of input video sequence as well as to different transmission rates, and it leads to significant improvement over the standard rate control scheme when it is applied to the low-delay H.264 system.

Chapter 7 This chapter concludes the thesis by highlighting the main findings and achievements resulted from our research effort made in the area of adaptive rate control for video compression and transmission. Recommendations and further work are also given.

Chapter 2

Rate Control for Video Transmission Over Wireless Network: An Overview

2.1 Framework of video communication system

Figure 2.1 shows a basic structure of a video communication system. The video communi-

Figure 2.1: Basic components of a video communication system, including video capture device, video encoder, channel encoder, modulator, demodulator, channel decoder, video decoder and video display device.

cation system consists of various building blocks, i.e., video capture device, video encoder, channel encoder, modulator, de-modulator, channel decoder, video decoder and video display device. This chapter will preview and introduce the current research activities and problems for video transmission over wireless network.

Video capture device The video capture device is used to generate raw video signals, which consists of consecutive video frames. In this thesis, it is assumed that the raw video signal of the video capture device is Quarter Common Intermediate Format (QCIF). There are a number of standard QCIF video sequences, and they are available in public domain (see [8]). Since each QCIF video frame contains 176×144 pixels, when 8 bits resolution is used to represent a luminance pixel, each frame requires about 202 Kilo bits (Kbits). Channel transmission rate of more than 6 Mega bits (Mbits) per second (Mbps) is required for a 30 frames per second (fps) QCIF video signals. It is obvious that the raw video signals need to be compressed in order to reduce the bandwidth requirement of the transmission channel.

Video encoder The objective of the video encoder is to compress raw video signals by eliminating the temporal and spatial redundancies that exists in the video frames. A typical video encoder is illustrated in Figure 2.2. The video encoder uses block-based coding schemes, where each of the frames are subdivided into smaller units called macroblocks (MBs) that are processed one by one. Motion estimation (ME) and motion compensation (MC) is used to remove the temporal redundancy that exists between successive frames. When the frame rate is sufficiently high, there is a great amount of similarity between neighboring frames. In ME/MC, current MB is compared with the MB of the previously encoded frame to derive a residue MB. The previously encoded frame is thus called reference frame, and the MB of the previously encoded frame is called reference MB. The residue MB contains only the difference between current MB and its reference MB. It is more efficient to code the residue MB rather than the MBs themselves. As there is a great

Figure 2.2: A typical video encoder structure. The encoding process involves motion estimation, motion compensation, discrete cosine transform, quantization, variable length coding and so on.

amount of similarity between MBs in the same frame, Discrete Cosine Transform (DCT) is employed to eliminate the spatial redundancy between MBs of a video image. The quantization block quantizes the DCT coefficients within a MB, and it is responsible for the most compression. Variable length coding (VLC) performs further compression of the quantized coefficients by representing coefficients that occur more often with shorter codewords and less often with longer codewords. The final output of the encoder is called video bitstream. In this thesis, MPEG-4 [3] encoder, and its successor, the new emerging MPEG-4 (part 10) encoder, which is also known as the advanced video coding (AVC) or H.264 [5], are studied because they are designed specifically for use in wireless video communication and can achieve a high compression efficiency.

The video encoders such as MPEG-2 and MPEG-4 support three different frame types. The frame encoded without reference to any past or future frames is called the intra frame (I-frame) and the coding process is intra-frame coding. The frame encoded with reference to any past frames is called the predicated frame (P-frame) and the coding process is inter-frame coding. The frame encoded with reference to both past and future frames is

bidirectional predicated frame (B-frame). In addition to the above three frame types, now the new emerging video encoder H.264 support more frame types (see [5] for more details). As a general rule, a video sequence coded in I-frame achieves low compression. However, it is robust to channel errors and introduces no delay. A sequence coded with I-frame and P-frame achieves moderate compression but with a slight degradation in error tolerance. Incorporation of all three frame types may achieve high compression efficiency, but at the cost of a significantly increased coding delay. Since the delay is not tolerable for wireless video communication system, B-frame is not used in this thesis.

Table 2.1 lists a comparison of several well known video encoders, including their target application, target bit rate range, possible frame types and so on. All the standards follow

Video encoder	MPEG-1	MPEG-2	H.263	MPEG-4	H.264
Typical applications	CD-ROM	TV	PSTN video phone	wide range of applications	wide range of applications
Target bit rate	1.2Mbps	2-10Mbps	a wide range	64Kbps-6Mbps	a wide range
Reference	1	1	1	1	1-5 frames

Table 2.1: Comparison of MPEG-1, MPEG-2, H.263 and H.264 video encoder. The typical applications, target bit rates and number of reference frames used for motion estimation are compared for the four video encoders.

a similar framework in terms of the encoding algorithms. The major differences between these video coding standards lie in the operating bit rates and the application they are target for [9]. MPEG-1, MPEG-2 and MPEG-4 are video coding standard proposed by ISO committee. In comparison, H.263 is a video coding standard proposed by ITU organization. H.264 is recent video encoding standard developed jointly by ISO and ITU in 2002. The H.264 achieves a higher compression efficiency than all the other video coding standards but at a cost of increased complexity.

Channel encoder Channel encoder is an essential component to a video communication system and it is especially important because compressed video signals are very sensitive

and can be distorted by unexpected channel errors. Channel coding is used to detect and correct any transmission errors, by adding redundant bits to the video bitstream.

Modulator Modulator is used to perform modulation which is to frequency translate the channel coded video bitstream onto a carrier wave for transmission. The key concerns of modulation are bandwidth efficiency and implementation complexity. There are a number of modulation techniques available for wireless communication system, e.g., Gaussian Minimum Shift Keying (GMSK) in GSM, Quadrature Phase Shift Keying (QPSK) in WCDMA.

Transmission channel This thesis does not focus on a specific wireless network. Rather, a very general wireless network is considered. Therefore, the transmission channel is modelled as a channel with a limited bandwidth and being error-prone, and these are inherent characteristics of wireless communications. Previous studies, e.g., [10][11][12][13][14], show that a two-state Markov model [15], provide a good approximation in modeling the error process at the packet level in fading channels. In the experiments of this thesis, a two state Markov model is employed to simulate wireless channel burst error pattern.

De-modulator, channel decoder, video decoder, and video display device At the receiver side, the de-modulator is used to retrieve the channel coded video bitstream from received waveform. After that, the channel decoder is employed to decode the channel coded video bitstream in order to obtain pure video bitstream. Finally, the video decoder reconstructs video frames, which are displayed in the video display device.

2.2 Challenges of wireless video communication system

2.2.1 Data transmission rate

It is well known that the widely deployed GSM network, which is the 2nd generation wireless network, typically provides a payload bit rate ranging from 10 to 15kbps. The enhanced GSM network, general packet radio services (GPRS), provides a slightly higher payload transmission rate as compared with GSM network, which ranges from 9 to 21.4kbps (per time slot). The development of 3rd generation wireless network, UMTS, is targeted to provide high speed packet data services in addition to offering high capacity for voice service [9]. The payload data rate of UMTS can reach a maximum of 2Mbps when the user is in indoor environment and nobody competes with him for bandwidth. It is obvious that a user cannot maintain the maximum payload data rate during an entire video communication session. The payload bit rate of UMTS is more likely in a range from 32 to 384kbps, which is meant for use in outdoor-to-indoor and pedestrian environment. In other words, the wireless network puts a strict constraint on the payload data rate. It is important for the video communication system to use the limited payload data rate efficiently and effectively.

2.2.2 Channel errors of wireless networks

Wireless network is known to suffer from high Bit Error Rate (BER). In radio communications, the signals transmitted over radio channel are subjected to both small-scale and large-scale signal variations [16]. The small-scale signal variations are due to multipath signal propagation. The signals in each path may have different time delays and attenuations, which introduces time delay spread and amplitude fluctuations on the received signal. The large-scale signals variations include path attenuations, terrain obstructions, antenna height. Transmission errors of a wireless network range from single bit errors to

burst errors or even a loss of the connection. Therefore, the design of a good wireless video communication system always involves the design of a good error protection scheme that enable the video bitstream to recover from unexpected channel errors.

2.2.3 Variable payload data rate in wireless network

Besides the limited payload data rate, a wireless network also has a variable payload data rate. The reason is that the wireless network provides several channel coding schemes and adopts adaptive bandwidth allocation scheme. In UMTS network, the bandwidth is allocated dynamically to a user depending on the number of users. When the number of users increases, UMTS decrease the bandwidth allocated to each user. Therefore, the payload data transmission rate of UMTS network is variable depending on the number of mobile users. Furthermore, UMTS supports 1/2 and 1/3 rate convolution coding for small number of users and 1/3 rate turbo coding for large number of users. It is well known that different channel coding schemes add different number of redundant bits to the basic payload. The payload data transmission rate is therefore variable even if the same bandwidth is allocated to a mobile user. The video application needs to adapt to variable payload data rate in order to use the wireless network efficiently.

2.2.4 Bursty characteristic of video transmission

As discussed above, the raw 30fps QCIF video sequence without compression requires a transmission data rate of around 6Mbps. Even with state-of-the-art compression, e.g., MPEG-4, television quality video sequence requires a few mega bits per second, while for low resolution, limited motion video sequences, a few tens of kilo bits per second is required [17]. The video transmission puts a lot of demanding in terms of bit rate [9], which offers technical challenges to wireless video communication system.

In addition to the high bit rate characteristics, video applications typically have bursty

characteristics. It is observed that the video encoder described Figure 2.2 produces a variable number of bits per frame that sometimes changes very abruptly. Figure 2.3 shows the number of bits per frame for a video sequence. The video test sequence *Foreman* is encoded by a constant quantization parameter. It is observed that the 64th frame requires

Figure 2.3: The number of bits per frame for video test sequence *Foreman*. The video sequence is encoded by H.264 video encoder at a constant quantization parameter 30.

around 5000 bits, but the 59th frame only needs about 700 bits. The decoded video picture quality are shown in Figure 2.4 which includes the 59th and 64th frames. Different video frames contain distinct texture and motion information, therefore, resulting in different number of bits are generated to represent different video frames. Next, the frame types, i.e., intra-frame and inter-frame, play important roles to the number of bits per frame produced by the video encoder. Generally, inter-frame requires less number of bits than intra-frame because the temporal redundancy is eliminated for Inter-frame. Furthermore, VLC adds variety to the number of bits per frame because different spatial correlations exists for different frames. Finally, the encoding parameters of the video encoder, such as

Figure 2.4: Subjective picture quality comparison for video test sequence *Foreman*, 2.4(a): the 59th frame; 2.4(b): the 64th frame. The video sequence is encoded by H.264 video encoder at a constant quantization parameter 30.

quantizer step size, also affect the number of bits per frame. For example, for a given frame and a frame type, a small quantizer step size generates a larger number of bits than a large quantizer step size. Typically, the reason is that more video information are lost when quantizer step size is large, and therefore it can be encoded with fewer bits. Wang [18] shows that the number of bits normally decreases as the quantization parameter increases, but researchers have found that the average number of bits of a frame does not vary linearly with the quantization parameter of the frame [18][19].

2.2.5 Sensitivity of video bitstream to bit error rate

Without special schemes, compressed video signals, i.e., video bitstreams, are extremely vulnerable against transmission errors. If a single bit error occurs in the transmitted video bitstream, the decoder cannot correctly identify the next unique synchronization codeword in the received video bitstream because of VLC coding, so that the whole data within the MB becomes corrupted and the decoder cannot reconstruct the MB correctly.

Considering compression schemes that rely on inter-frame for high compression efficiency, i.e., the previously encoded and reconstructed video frame is used to predict the next frame, the loss of information in one frame will propagate to the following frames. Therefore, one bit error in the transmitted video bitstream has considerable impact on the quality of the reconstructed video.

2.2.6 Challenges of wireless video communication system

Wireless video communication system offers a number of interesting technical challenges. One of the issues is that video bitstream is sensitive to the high error rate of the wireless network. The well known technique to combat transmission errors is forward error correction (FEC) [20][21][22]. FEC is normally implemented by the channel encoder and decoder. The channel encoder adds redundant bits to video bitstream and the channel decoder uses the redundant bits for error detection and correction. FEC has the effect of increasing transmission overhead and reducing bit rate for the payload data. A good FEC scheme should be adaptive to time-varying channel BER, so that it not only protects the video bitstream well but also uses the available channel bandwidth efficiently. Therefore, the problem of how to allocate the channel bandwidth between the video encoder and channel encoder in order to use the limited channel bandwidth efficiently is an important problem in wireless video communication. This thesis examine the joint source channel rate control method and how it can solve the problem.

Another issue is that bit rate of the video encoder should adapt to the variable payload data rate of the wireless network in order to make full use the capability of the wireless network. However, as discussed in above section, the bit rate of the video encoder is also variable due to inherent characteristics of the video encoder and video frames. The problem can be solved by adopting proper source level rate control, i.e., how to control the video encoder to achieve a specified data rate. In section 2.4, we review the current existing joint

source channel rate control schemes and rate control for video encoder.

2.3 Error control for wireless video communication system

Before reviewing joint source channel rate control, it is important to review different error control schemes for wireless video communication system. The video bitstream is highly sensitive to data packet loss and unexpected channel bit errors, the decoded video quality is bound to suffer dramatically due to high channel bit error rate (BER) (ranging from 10^{-3} to 0.1) as a result from adverse channel conditions. The quality degradation is exacerbated when no error control mechanism is employed to protect encoded video data against the hostility of wireless channels, as the temporal and spatial predications used in the video standards lead to error propagation in both time and space domain. In order to mitigate the degradation of decoded video quality caused by wireless channel errors, error control and concealment schemes must be applied to both the video encoder and decoder. Wang and Zhu [23][24] reviewed the existing error control and concealment techniques in video communication systems.

Zero-redundancy error concealment technique Error concealment schemes are usually employed at the video decoder side to conceal the effects of errors by predicting the lost or corrupted video data from correctly decoded data information. Suh and Ho [25] describe a temporal domain recovery technique by using spatiotemporal redundancies in the received video signal for data recovery. The motion vector (MV) recovery performance is improved with three concealment methods including the modified average algorithm, the extension matching algorithm with an initial MV, and the MV recovery algorithm using optical flow fields. Some other well known error concealment techniques are maximally

smooth recovery [26] and projection onto convex sets (POCS) [27][28].

Error resilience video coding technique Video encoder uses error resilience source coding techniques to enhance the robustness of compressed video bitstream. One of the famous error resilience video coding techniques relies on re-synchronization markers which can help the video decoder to regain synchronization when video bitstream corrupts. Taluri [29] introduces the robust video coding methods standardized in ISO MPEG-4 and they suggested using re-synchronization strategies, data partitioning, reversible VLCs and header extension codes (HEC). Another well known technique is adaptive reference selection, in which the reference frame used for ME/MC can be any of the previously encoded frames. For example, in [30], the video encoder selects frames which are received correctly by the video decoder as references for ME/MC.

Transport level error control The transport level error control is a technique to protect video packets from transmission errors, which operates at transport level. In contrast to the error resilience video coding technique which adds redundant bits within bitstream, the transport level error control technique will not alter the bitstream of the video encoder. Automatic repeat request (ARQ) is one of the widely used error control techniques at the transport level [31]. This technique can efficiently recover packet loss, but the video source must have an ARQ buffer to hold the sent-out packets until the receiver acknowledges receipts of correct frames or packets. If the video packets are corrupted, the video source will retrieve the packets and resend them. FEC is also widely used in communication systems to detect and correct errors at transport level. For example, in H.261 and H.263, a (511, 493) Bose Chaudhuri Hocquenghem (BCH) code is used and it can correct 2 bits of random errors per video packet [32]. Hybrid schemes which combine FEC and ARQ together are also possible and they can be divided into type I and type II ARQ. In type I hybrid ARQ, redundant bits are added to every data packets no matter they are re-

transmitted or not. In contrast, in type II hybrid ARQ, redundant bits are transmitted only when they are needed and the erroneous packet is kept for future use rather than discarded. In [33], an improved hybrid ARQ scheme is proposed. Because certain error correction capability is provided by using FEC protection in each (re)transmitted packet, video packet can be recovered from each transmission or retransmission alone if the errors are within the error correction capability. The ARQ packets can also be combined with previously transmitted packets to formulate a more powerful method to recover the data. Hybrid ARQ have been studied intensively over the past few years, and more research results are presented in [34][35].

2.3.1 Unequal error protection

Masnick and Wolf [36] introduce unequal error protection (UEP) and the UEP is employed in applications where the transmitted data is a coded signal such as speech, audio and image. The basic principle of UEP is that the channel encoder gives unequal protection to the outputs of the source encoder, based on the knowledge of error sensitivity of the source bits, as well as the channel conditions. For example, in video signal transmission, it has been shown that motion vectors (MVs) are more important for image reconstruction than the texture information. Obviously, video bitstream which have a greater impact on the overall picture quality, e.g., MVs, should be protected with a stronger protection. For example, Martini *et al.* [37] evaluates the effect of errors in different bits of a MPEG-4 bitstream and draws the conclusion that UEP is a better method than equal error protection (EEP) to protect video bitstream. Later, he proposes a method in [38] to calculate FEC code rate for UEP, and both UEP-protected bitstream and EEP-protected bitstream have the same number of bits after adding redundancy. More recent work in this area deals with UEP on multicast network [39]. By using different codes for UEP, the loss of multicasting packets on an unreliable multicast network can be recovered. Due to the fact that by using UEP, only one logical multicast channel is required, it is more practical

than other schemes requiring many separate multicast channels. The UEP can not only be used in ME/MC DCT based video communication systems, but has also recently been proposed in wavelet based encoder systems [40][41]. In [40], the data from wavelet encoder are protected by codes with four different levels of error protection capabilities according to their significance. Each level of the UEP code is designed for a specific digital matched filter with pseudo-noise code synchronizer. A physical layer UEP scheme is proposed in [42] and the scheme is not only intended for video service but also other services which require different quality of services (QoS). The QoS attributes of applications such as delay and so on are directly associated with the amount of redundancy that the channel encoder will add.

In some cases, the video bitstream is allocated to different transmission channels, and the main design issues are how to optimally allocate the video bitstream into different channels and how to design error correcting codes with different error correcting capability for different channels. Multiple description coding (MDC) is a main technique used to optimally allocate the bitstreams or represent the input images with multiple bitstreams. Research in this area can be found in [14][43][44]. The problem of how to design the error correcting codes for different sub-channels has been studied in [45][46]. In [45], a video sequence is first divided into two groups of pictures and then protected by multiple sub-bitstream UEP scheme. At the decoder side, the correlation between two groups of pictures are used to recover lost packets. It has been proved that UEP can be combined with sub-channels transmission efficiently. In [47], MDC is used together with sub-channel transmissions to protect motion vectors from transmission errors. These methods can also be regarded as a special kind of UEP because different sub-channels can have different channel coding schemes and are suffered from various error patterns.

Recently, layer coding (LC) has been receiving increasing attention in various video applications. The basic principle of LC is that the first or base layer is coded independently,

and subsequent (enhancement) layers are coded dependently, and each layer of the hierarchy can increase the frequency, spatial, and temporal resolution over that of the previous layer [48]. Research works about LC can also be found in [49][12]. LC can be treated as a special kind of MDC in that LC consists of several sub-bitstreams of an image, in which the enhancement sub-bitstreams depend on the base sub-bitstreams. A comparison between LC and MDC is given in [50]. Similarly, the LC bitstream can also be protected by UEP or transmitted in different sub-channels in order to improve the system efficiency. For example, in [51], Schaar applies more error protection to base layers and less protection to enhancement layers, which results in a significant improvement in packet-loss resilience.

2.4 Rate control schemes for video communication system

Separation Theorem: In conventional communication systems, source coding and channel coding are usually concatenated and the source coder and channel coder are designed separately. This is based on Shannon's famous coding theory [52] which states that the system design can be optimized by separately optimizing the source encoder/decoder pair and channel encoder/decoder pair [53]. Although the separation theorem has been proved theoretically, it has problems when it is applied to a practical scenario. Firstly, the source coding theory assumes that the channel is error free and the channel coding theory assumes that the channel inputs are independent identically distributed (i.i.d.). However, in practical communication systems, the channel errors normally exist and the encoded signals always contain redundancy and therefore the i.i.d. assumption does not hold. Furthermore, the separation theorem assumes that there is no constraints on the complexity and delay of the source and channel coders which is often not true in practical situations.

Joint source channel coding: The basic idea behind almost all joint source channel coding (JSCC) techniques is to exploit the knowledge of the source or/and channel characteristics in the design of the source/channel coder to make it robust to channel errors. Recent research have indicated that JSCC is a necessary means to provide an efficient and robust video transmission in practice [54]. JSCC encompasses a range of schemes which include [55]:

1. Rate allocation (control) between the source and channel encoders;
2. Optimization of the channel encoder according to source characteristics;
3. Design of the channel encoder considering residual source redundancy;
4. Modification of the source encoder and decoder structures according to channel characteristics.

For example, the UEP discussed above is a technique that designs channel codes to capitalize on specific source characteristics. It is apparent that a joint source-channel encoder/decoder is much more difficult to design than separate design of the source encoder/decoder and channel encoder/decoder. A pragmatic approach is to keep the source coder and channel coder separate but optimize their parameters jointly, and the key problem of the optimization is the bit allocation between the source encoder and channel encoder [56]. In this thesis, we call it joint source channel rate control.

In wireless video communications, compressed video bitstream need to be transmitted over wireless channels that have inconsistent and time-varying data rate and variable BER. To make the best use of available wireless network resources at any time, anywhere and provide a maximum level of subjective video quality for users, an adaptive rate control mechanisms must be used in video communication systems to improve the system efficiency as well as the decoded picture quality [57]. Given a time-varying channel bandwidth U_c and

channel BER, the determination of the most appropriate channel coding scheme and the corresponding rate control algorithm must be solved together, in order to achieve an optimal trade-off between the bandwidth efficiency and the performance of the video service. In general, rate control schemes can be classified into two aspects, i.e., rate control between the video encoder and channel encoder [58][59][60] and rate control for the video encoder alone [13][61] [62][63]. Next, we will review the existing joint source channel rate control schemes for video communication systems.

2.4.1 Joint source channel rate control

A variety of techniques have been proposed to enhance the robustness of the encoded video signals, such as error control schemes proposed in [55][64][65] and adaptive intra-frame coding proposed in [11][66]. For error control techniques, even though FEC or ARQ is employed, some redundant information according to a specific error control mechanism are added to the compressed video data to provide error-resilience features to video bitstream. This process will decrease the bandwidth allocated to the video encoder when the total bandwidth is fixed. Even for the video encoder alone, different encoding schemes lead to different compression efficiency and error robustness. For example, intra-frame coding provides robustness to the encoded video bits but results in a higher bit rate than inter-frame coding. Nevertheless, with too many intra-frame coded MBs will significantly decrease the compression efficiency of the video encoder. In designing the video communication systems, a trade-off should be made between throughput and reliability. Reliability can be increased with a reduction in throughput. Conversely, the throughput can be increased with a reduction in reliability.

Error control schemes, such as FEC and ARQ, are often used to correct errors introduced by transmission of the message through error-prone channels. ARQ scheme requires a feedback channel to transmit retransmission request, while FEC has no such requirements.

JSCC with FEC protection have been studied in [64][65], and JSCC with ARQ protection have been discussed in [67][68].

Joint source channel rate control with FEC

When FEC is used as an error control technique, the video communication system must decide on the the optimal FEC code rate. Suppose the input video sequence is X_i , the encoded video bitstream is composed of $b_i^l, l = 1, 2, \dots, L$, each belongs to a priority classed P^l . L is the total number of bitstream components. b_i^l is then coded by a channel encoder with code rate C^l . Given the transmission channel conditions, such as the available bandwidth and BER, the encoder needs to find the optimal bit allocation scheme between the video encoder and channel encoder in order to minimize the overall picture distortion at the receiving end. To be more specific, suppose the overall coding rate is R_{budget} , an optimal bit rate allocation should be made between the video encoder and channel encoder, in order to minimize the total distortion D_{s+c} , which is composed of the source distortion D_s and channel distortion D_c , i.e.,

$$\text{minimize } D_{s+c}, \quad (2.1)$$

$$\text{subject to } R_{s+c} \leq R_{budget} \quad (2.2)$$

In the above equation, R_{s+c} (bps) is the actual overall bit rate after the video encoding and channel encoding. For UEP-based video communication system, R_{s+c} is defined as

$$R_{s+c} = \sum_{l=1}^L R_{s+c,l} \quad (2.3)$$

where L is the maximum number of sub-bitstreams. $R_{s+c,l}$ (bps) is the actual overall bit rate for the l th sub-bitstream b_i^l . For example, when EEP protection is employed to the H.263 bitstream, $L = 1$.

In order to derive an efficient bit allocation scheme between the video encoder and channel encoder as well as achieve good picture quality, one of the key problems is how to obtain the actual rate distortion characteristic of the video communication system. The reason is that once the rate distortion characteristic is obtained, the well known Lagrangian method [65] can be used to solve the above optimization problem. Early works in the estimation of D_{s+c} , e.g., proposed in [69], ignore the error propagation beyond one frame and approximates total block distortion as a simple sum of the quantization distortion of that block and weighted concealment distortion of corresponding blocks in the previous frame. Recently, a number of studies reported in [56] and [64][11][66] focus on robust estimation of D_{s+c} . For example, in [64], an operational universal rate-distortion function is constructed by iterative encoding a frame according to characteristics of selected video coder, channel coder, coding rates, and channel conditions. The performance of this rate distortion function is constrained by its complexity, which grows exponentially with the number of iterative encoding times. Methods for reducing the complexity in obtaining the rate distortion characteristic are normally based on empirical models and theoretical derivations. One of the empirical models is given in [56], in which the distortion is represented by two simple parameters. However, the drawback of this approach is that the necessary model parameters cannot be derived from the commonly used signal statistics, such as the variance, correlation, or power spectral density. Zhang *et al.* [66] derive a recursive algorithm to estimate the overall distortion at pixel-level after analyzing the error concealment method and error propagation between frames. The accuracy of the rate distortion estimation is maintained since the algorithm considers previously decoded frames. However, the computation of the overall distortion could be very complex due to different channel coding schemes, and correlations between the source and channel distortions. He *et al.* [11] thus make an assumption that the source distortion and channel distortion are uncorrelated with each

other, and the overall distortion can be written as [11],

$$D_{s+c} = D_s + D_c \quad (2.4)$$

In this case, the source distortion and channel distortion can be computed separately and then summed together to obtain the overall end-to-end distortion. For example, following [11] within the ρ -domain, the source distortion model D_s can be written as

$$D_s = \sigma^2 e^{-\alpha(1-\rho)} \quad (2.5)$$

where σ^2 is the variance of the video signals [70]. α and ρ are parameters. The channel distortion D_c is a function of the packet loss ratio and mean square error (MSE) between the reconstructed frames n and $n - 1$. Clearly, it is easier to derive the overall distortion by calculating D_s and D_c , separately. Recently, Z.G. Li *et al.* [71] propose a joint source channel coding method which does not rely directly on robust estimation of D_{s+c} . By adapting channel coding rate to time-varying channel conditions, Z.G. Li computes the bit rate for source coding and the FEC rates for channel coding simultaneously.

Joint source channel rate control with ARQ

The joint design of the video encoder and ARQ scheme is introduced in [67] and it is different from the integrated design of the video encoder and FEC, in which a trade-off is made between source coding accuracy and channel error protection capability to minimize the end-to-end distortion. By applying the joint source channel coding with ARQ, lossless-source coding and channel coding are combined to prevent the video decoder from reconstructing garbled data, while maximizing the channel throughput efficiently [68]. The video encoder adds some special markers at the compressed video bitstream and they will be examined by the video decoder for video signal reconstruction. A request for retransmission is generated if a marker does not appear in its proper location.

It has been shown that purely passive error recovery techniques such as ARQ generally provide inadequate reconstructed video quality and therefore some level of FEC may be necessary to provide a reasonable quality of service [64]. In this way, Hybrid ARQ/FEC method is a more attractive method than pure ARQ. Intuitively, a FEC based ARQ scheme is better than conventional ARQ in that during retransmission following an error, there is no need to send the whole packet but only the partial information. For more details, please refers to [72][73].

2.4.2 Rate control for the video encoder alone

In block-transform video coders, there are several different encoding parameters which can be adjusted to control the output bit rate. Any attempt to control the output bit rate of a video encoder involves trade-off between picture quality and compression efficiency. Therefore reducing the output bit rate of the video encoder could be done at the expense of a degradation in picture quality. On the other hand, improving picture quality could be done at a cost of increased output bit rate of the video encoder. In a real-time video communication system, the encoded bits are normally placed into a small buffer, which resides in the video encoder, and a finite number of bits can be sent from the buffer during each frame interval. Rate control for the video encoder alone is therefore responsible for avoiding buffer overflow and underflow under a given output bit rate and buffer size. Examples of work in this area include the model-based rate control [70][74][75] and the operational rate-distortion based rate control [76][10][77]. As indicated, the model-based rate control scheme is based on some empirical rate models, and the output bit rate is a function of encoding parameters of the video encoder. In this way, for a given bit rate allocated to the video encoder, the rate models are used to derive appropriate encoding parameters, such as quantization parameter, the number of skipped frames and so on. On the other hand, the operational rate-distortion based rate control is an optimization

problem to find the best encoding parameters.

Quantizer step size determination

By changing the quantizer step size is the most direct approach for the video encoder to satisfy the bit rate requirement. When the video encoder adopt a larger quantizer step size to encode a frame, the video signal information loss increases, and less bits are required to encode the frame.

Model-based quantizer step size determination In model-based quantizer step size computation, a rate model is firstly built and it is a function of several parameters of the video encoder. For example, the quadratic rate-quantizer model is adopted by the MPEG committee in July 1997 as a part of the video verification model for rate control. It is given by [75]

$$T_{texture} = \frac{x_1 \times MAD}{QP} + \frac{x_2 \times MAD}{QP^2} \quad (2.6)$$

In the above equation, $T_{texture}$ denotes the number of bits for the texture. MAD is the mean absolute difference of the texture, which is an indication of the MB complexity and can be obtained after ME/MC. QP is the quantization parameter for the frame, which has a fixed relationship with quantizer step size. For example, in H.263 the quantizer step size is two times as big as the quantization parameter. x_1 and x_2 are the first- and second order model coefficients. The main procedure of the model-based quantization parameter can be divided into four stages: (i) initialization, (ii) computation of the target bit number for the current frame, (iii) computation of the quantization parameters for the current frame by using a rate model, (iv) updating the rate model parameters and buffer level as well as other parameters of the rate control scheme based on the results obtained from the current frame.

The initialization stage initializes a buffer level of the video encoder. It sets a maximum buffer size, and a frame skip threshold. In the later encoding processing, the video encoder will determine whether it skips the next frame based on frame skip threshold and buffer level.

The computation of the target bit number for the current frame has several different methods. For example, it can be derived directly from the bit rate allocated to the video encoder, i.e.,

$$B_s = R_s / F_r \quad (2.7)$$

where R_s (bps) is the bit rate allocated to the video encoder, F_r (fps) is frame rate, and B_s (bits) is the number of bits allocated to the current frame. However, such method has been replaced by some better target bit number computation algorithms, in which the current buffer level before encoding is taken into account. For example, in [70], the target bit number for the current frame is given by

$$B_s = R_s / F_r - \Delta \quad (2.8)$$

where Δ is defined as

$$\Delta = \begin{cases} \frac{W}{F_r}, & W > Z \times M \\ W - Z \times M, & \text{otherwise} \end{cases} \quad (2.9)$$

in which W (bits) is the current buffer level, Z is the threshold value, and M (bits) is the maximum buffer size. In TMN8 rate control [70], $Z = 0.8$. If current buffer level is greater than $0.8M$, then the encoder knows that too many bits are stored in the video encoder buffer, therefore it uses less bits to encode the next frame. In contrast, if the buffer level

is less than $0.8M$, then the encoder knows that fewer bits are in the video encoder buffer, so it uses more bits to encode the next frame.

The quantization parameter of the current frame can be computed by using known coefficient values. In TMN8 rate control scheme, the quantization parameter is given by [70]

$$Q = \sqrt{\frac{16^2 \times K \times MAD \times S}{B_L \times \alpha}} \quad (2.10)$$

where Q is the quantizer step size. MAD is the mean absolute difference of the texture. B_L (bits) is the number of bits for remaining uncoded MBs. S is the total MAD for remaining uncoded MBs. K and α are model coefficients whose values are known a priori by the video encoder, and shall be updated from previously encoded frames. It is obvious that different rate models lead to different ways to calculate the quantization parameter. For example, in [74], the quantization parameter QP is calculated by

$$QP = \left(\frac{\beta}{B - \alpha_1} \right)^{\frac{1}{\gamma}} \quad (2.11)$$

where B (bits) is the number of bits. α_1 , β and γ are the model coefficients whose values are well known by the video encoder, and shall be updated from previously encoded frames. He et. al [78] proposed a new source model which is given as below

$$QP = \theta \times (1 - \rho) \quad (2.12)$$

where θ is a constant. ρ is the percentage of the zeros among the quantized transform coefficient. ρ monotonically increases with the quantization parameter QP .

It is important to update coefficients of all rate models from previously encoded video

frames, so that an accurate calculation of the quantization parameter for the rest of frames can be achieved. Pan *et al.* [79] update the coefficients of the quadratic rate-quantizer model by using the least-square algorithm. In this way, the quadratic rate model adapts itself to the input video sequence and provides a more accurate calculation of quantization parameters for the rest of the frames. Researchers also propose to improve the rate-quantizer model accuracy by iteratively encoding the same frame and update the model using the data obtained from the iterative process [74] [80].

A lot of work has gone into finding a more accurate rate-quantizer model for the video encoder. In order to improve the accuracy of the rate-quantizer model, one has to accept the complexity increase which is brought by the new models. For example, Lin *et al.* [80] propose a cubic spline approximation model in which the bit numbers and quantization parameters are collected at several sample points and are used to derive a spline function, in which the quantization parameter is an independent variable of the spline function. The cubic spline model achieves good performance for estimating the bit number at a cost of computational complexity because it uses an iterative encoding process.

Rate-distortion based quantizer step size determination The rate distortion problem is formulated as follows [81]: finding an optimal quantizer, or operating point, $x(i)$, for each coding unit i such that

$$\sum_{i=1}^N r_{ix}(i) \leq R_T \quad (2.13)$$

and the total distortion of all coding units is minimized. In the above formulation, N is the total number of coding units. $r_{ix}(i)$ (bits) is the number of bits of the unit i . $d_{ix}(i)$ is the distortion of unit i using quantizer x . R_T (bits) is the total number of bits which is given to the total coding units. The solution to this problem relies on the Lagrangian optimization

technique [82][83]. In Lagrangian optimization approach, the constrained optimization problem as shown in equation (2.13) is equivalent to the unconstrained problem derived by introducing a non-negative Lagrange multiplier λ associated with each rate constraints. The problem can be formulated as

Find the quantizer choice x^ such that* (2.14)

$$x^* = \arg \min \sum_{i=1}^N f(d_{ix(i)}) + \lambda \sum_{i=1}^N r_{ix(i)}$$

where $f(d_{ix(i)})$ is the distortion of the i th coding unit when it is encoded with quantizer x . The solution to the problem is to find out an appropriate value of λ such that no constraint is violated.

One of the key problems of rate-distortion based rate control is how to obtain an accurate and robust rate distortion model, which has been studied intensively. Most of the rate distortion formulations focus on improving estimation accuracy of the rate, and often implicitly assume that the distortion is a linear function of the quantization scale and does not consider error propagation in inter-frame coding. Recently, in [76], the inter frame distortion approximation method has taken the dependencies that arise in the choice of quantizers for the reference frames into account.

Frame rate determination

The frame rate is the number of encoded frames per second, and it is another important encoding parameter which affects the encoded bit rate. Since the frame rate control method targets at reducing the temporal but not the spatial redundancies of the video signals, it is generally used when the quality of individual pictures cannot be compromised [84]. Frame skip has been introduced in some existing rate control schemes [70][75], and it is a very simple way to change the frame rate. In [70], it is proposed that if the current buffer level B_s

is greater than the threshold buffer value B_{th} , the video encoder will skip the next several frames until the buffer level B_s is smaller than B_{th} . At the decoder side, the video decoder will display the recently received image until the next frame comes to the decoder. Frame skip is a very effective method to reduce the output bit rate at the expense of degradation in picture quality.

Reduction of frame rate is often used in transcoder as a way to transmit video signals through different networks with distinct data transmission rates. For example, in a wireless network which normally has limited data transmission rate, the quality degradation due to low bit rate is significant with a frame rate of 25 or 30 frames per second. [85][86][87] present some results on how to sophisticatedly drop the frames.

Other techniques

Although the quantizer step size and the number of skipped frames are the two most widely used encoding parameters for rate control schemes, other encoding parameters can also be used. Since the video encoder has a lot of encoding parameters, modification of any of them will affect the output bit rate of the video encoder. Turaga and Chen [88] apply InterIntra mode decision to the rate control algorithm as inter-frame coding is more efficient when ME/MC is effective and intra-frame coding is more efficient when there is a scene change or when input video sequence consists of fast motion scenes. Another way to modify the output bit rate of the video encoder is to encode only certain number of low frequency coefficients of DCT, which is called pruning technique. Fewer bits are generated per MB at the cost of reduced quality due to removal of high frequency coefficients. Such pruning techniques have been studied in [89][90]. Another simple way to modify the output bit rate of the video encoder is to use different video encoders with various compression efficiency. This method is called spatial scalability [91] and it is widely used in transcoder, in which the compressed bitstream is firstly decoded and then re-encoded by another different en-

coder to produce a different bit rate.

2.5 Tools

2.5.1 Measurement of distortion

For a quantitative analysis of wireless video systems, we need a measurement for the video signal distortion introduced by the video encoder and/or the wireless channel. Although people can perceive the subjective picture quality at the decoder, one can not quantify by how much the picture quality degrades. In practice, the most common distortion measurement relies on MSE which provides a rather consistent measurement result as long as the video signals to be compared are affected by the same type of impairment [92]. For example, the subjective quality produced by a particular video encoder at two different bit rates for the same input video signal can usually be compared by PSNR.

Given a video frame $f(i, j)$ that contains M by N pixels, and a reconstructed frame $F(i, j)$ which is reconstructed by decoding the encoded version of $f(i, j)$. Distortion measurement is computed on luminance signals, and the pixel value $f(i, j)$ ranges from black (0) to white (255). MSE is defined as

$$D_{MSE} = \frac{\sum_{i=1}^M \sum_{j=1}^N [f(i, j) - F(i, j)]^2}{M \times N} \quad (2.15)$$

where D_{MSE} is the MSE of the reconstructed frame relative to the original video frame.

It is noted that MSE is usually converted to PSNR which is then used as a measurement of the distortion of a reconstructed image as compared with the original image. PSNR (dB)

is defined in decible as follows

$$PSNR = 10 \log_{10} \left(\frac{255^2}{D_{MSE}} \right) \quad (2.16)$$

where 255 is the peak-to-peak range of the encoder and decoder video signals. D_{MSE} is the MSE between the original and reconstructed video frames. The evaluation of PSNR is normally based on the luminance component (Y) of the video image. PSNR increases with increasing picture quality. Figure 2.5 shows a subjective comparison of video frame. The

Figure 2.5: subjective quality, PSNR and number of bits comparison for video test sequence *Foreman*. (a): PSNR=57.86 (dB), number of bits=170264 (bits); (b): PSNR=28.72 (dB), number of bits=6760 (bits)

video frame is encoded with H.264 video encoder. In Figure 2.5(a), quantization parameter of the encoder is equal to 2, while in Figure 2.5(b), the quantization parameter of the encoder is 50.

In addition to those objective distortion methods, subjective methods are also very important ways to evaluate video quality. In a common subjective test, the subjects are asked to comment on various aspects of the video image. Does the video resolution of the

image give a good likeness of the subject matter? Is the frame rate of the video display fast enough to give a good indication of the motion of the subject? T. Alpert et. al [93] provides a great details on subjective test methods and results for MPEG-4 video coding.

2.5.2 Linear regression method

Linear regression method or least squares is by far the most widely used modelling method. In this thesis, we will use this method to update coefficients of models used in the thesis. Most analytical methods are characterized by a linear relationship between the independent variable (input) and the response of the system, i.e.,

$$y = m \times x + b \quad (2.17)$$

In the above equation, x is the independent input. y is the response which can be observed. m and b can be determined from a series of experimental results.

Suppose that we have a collection of input data $X = \{x_1, x_2, x_3, \dots, x_n\}$ and the corresponding response $Y = \{y_1, y_2, y_3, \dots, y_n\}$. m and b can be derived from observed sequence (see Appendix A for the detailed proof).

$$\tilde{m} = \frac{n \sum_{i=1}^n (x_i y_i) - \sum_{i=1}^n x_i \sum_{i=1}^n y_i}{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2} \quad (2.18)$$

$$\tilde{b} = \frac{\sum_{i=1}^n x_i^2 \sum_{i=1}^n y_i - \sum_{i=1}^n (x_i y_i) \sum_{i=1}^n x_i}{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2} \quad (2.19)$$

In this thesis, the size of input data is n and it is assumed that n is fixed. The assumption is based on the facts that not all of previously encoded frames are useful for updating linear model coefficients. There are a lot of similarity between successive frames. However, those similarity do not exist if the the frame is far away from the other one.

Linear regression method relies greatly on the previous encoded frame. The size of the sliding window should be chosen carefully in order to handle both fast motion scenes and slow motion scenes. If the scene of input sequence changes significantly, i.e., the input consists of a high motion scene, a smaller window with more recent data points is used. Otherwise, a larger window with more past data points is used.

Chapter 3

Adaptive Joint Source-Channel Rate Allocation For Video Transmission

3.1 Introduction

The transmission of video signals over networks, especially wireless networks, presents a number of challenging problems that remain to be resolved. One of the most difficult issues stems from the inevitable transmission errors due to various channel impairments such as fading, interference and noise. The raw video signals are compressed by encoding algorithms such as MPEG-4 [3] and H.263 [94] to reduce the number of bits which are required to be transmitted to the decoder. As the video signals that are being transmitted are compressed for bandwidth efficiency, a single bit error may cause severe degradation in video quality.

It is often necessary for a video transmission system to provide some form of error resilience features to protect video bits from channel errors, e.g., layered coding with transport prioritization [23]. FEC is well known, for both its error detection and error correction capability in data communications by adding an amount of controlled redundancy to the original in-

formation. There is no doubt that the channel decoder can correct most of the transmission errors if enough redundancy is added, and that means that channel distortion is reduced at a cost of redundancy. As the transmission rate of the wireless channel is usually limited, when redundant bits are added, the bit rate allocated to the video encoder has to be reduced, which usually leads to an increased quantization distortion to the encoded video signals. In this chapter, we focus on the problem of how to efficiently allocate available channel transmission rate between the video encoder and channel encoder to effectively minimize the end-to-end distortion, thereby achieving the best reconstructed picture quality.

A pragmatic approach for current state of the art joint source channel rate control scheme is to design the video encoder and channel encoder separately, but optimize their parameters jointly [56]. As the encoded video bitstreams are not equally sensitive to channel errors, an effective joint source channel rate control scheme should protect the encoded video bitstreams by different error correcting codes and different FEC code rate, according to the error sensitivity of the bitstream and the channel characteristics [64].

For this chapter, based on the analysis of the behavior of the video encoder, channel encoder and decoder as well as error pattern of the transmission channel, we develop a novel joint source channel rate control scheme for non-layer video communication systems working in error-prone environments. Our objective is to allocate bit rate to the video encoder and channel encoder adaptively, such that for sub-bitstreams with different importance and corresponding error-concealment methods applied to them, the end-to-end distortion is minimized for a given channel condition. This chapter is not to obtain performance bound based on kinds of assumptions and a lot of computational complexity, but a practical solution which can be applied to real time video communication systems. Therefore, the proposed scheme is low in complexity but yet effective.

In order to fulfill the objective, decoded BER is proposed to be used as a control component in our scheme to solve the classical rate allocation problem. The decoded BER is the BER of the video bitstream at the video decoder after channel decoding. The proposed rate allocation scheme avoids searching a large parameter space, which includes parameters of the video encoder and parameters of the channel encoder, and constructing complex rate distortion models, and it is adaptive to time-varying channel conditions such as the channel bandwidth and BER. Since UEP is employed in our scheme to protect each class of video sub-bitstream, and it is also adaptive to time-varying channels, maximum efficiency in using the bandwidth or transmission bit rate of the channel can be achieved. Our extensive experimental results demonstrate that the proposed rate allocation scheme provides very good performance as compared with the existing methods for rate control, while the computational complexity of our method is significantly reduced.

The organization of the rest of this chapter is as follows. An overview of the general wireless video transmission system as well as error control techniques are presented in Section 3.2. The measurement of the overall end-to-end distortion of the video coding and transmission system is introduced, and the equations used in the distortion measurement are simplified in Section 3.3. Section 3.4 gives a detail description of our proposed adaptive joint source channel rate allocation scheme using decoded BER as a control arguments. Section 3.5 evaluates the performance of our proposed scheme. Finally, Section 3.6 concludes the chapter.

3.2 UEP-based video communication system

3.2.1 Joint source channel video coding

Some joint source channel coding approaches assume that the channel condition is unknown. However, if some knowledge of the channel are available, or even if a set of likely channel models is known *a priori*, these can be incorporated into the design of the system to improve the overall performance [55]. Following previous works [56][11], the channel BER P_e is assumed to be known by the encoder. In this chapter, we concentrate on P-frames transmission and assume that I-frame is transmitted to decoder intact as P-frames occupy most of transmission periods in practical situation and they are especially important to the overall video quality.

A general wireless video transmission system with a channel feedback is shown in Figure 3.1. In Figure 3.1, f_n and $\tilde{f}_n$ are the n th frame at the encoder and decoder side, respec-

Figure 3.1: Wireless video system with feedback. The compressed bitstream is divided into several sub-bitstream, and each sub-bitstream has different priority.

tively. f_{nk} is the k th sub-bitstream of frame n after data partitioning. Data partitioning re-arranges the symbols in such a way that all symbols of one data type (e.g., MVs, mac-

roblock headers, texture coefficients) that belong to a single slice are collected into one coded bitstream. Recently, this data partitioning scheme has been implemented in the new standard video coding algorithm, H.264 [5]. Following previous work [95], we give a typical MPEG-4 video packet (VP) structure as following. Data partitioning is enabled in the

Figure 3.2: Data partitioning for MPEG-4 video packet. Each MPEG-4 video packet consists of a re-synchronization marker, which indicates the start of a new video packet. The video packets also includes quantization parameter, motion vectors, and texture (luminance and chrominance) coefficients.

video packets.

In the following analysis, for simplicity, we assume the size of VP is 1000 bits and each VP is divided into two sub-bitstreams with different priorities. The sub-bitstream H includes all headers and motion vectors and it has higher priority. The rest of data in the same VP are in the sub-bitstream L , which has lower priority. VP is taken as a basic processing unit. An MPEG-4 bitstream can be treated as a huge collection of VPs.

At the decoder side, we employ the error concealment scheme as the following [29]: If an error is detected in the motion or header section of an MB, the decoder flags an error in this MB and simply copies the MB at the same location from the previously decoded frame to replace this MB. If an error is detected in the texture coefficients section of an MB, the decoder reconstructs the original texture coefficients from neighboring, or previously

decoded MBs by using interpolation techniques. It is clear that the sub-bitstream H is more important in video picture reconstruction than the sub-bitstream L . UEP is therefore an appropriate method to protect video bitstreams in an error-prone environment [37].

3.2.2 Error control

Reed-Solomon (RS) code is one of the widely used FEC codes [96]. For symbols composed of m bits, a (N, K) RS code with symbol length N and dimension K encodes K information symbols (Km bits) into N symbols (Nm bits). Clearly, the $N - K$ parity symbols are the FEC overhead that consume a portion of the bit rate and lead to a code rate of K/N . For RS codes operating on m -bit symbols, the maximum symbol length is $N_{max} = 2^m - 1$. However, if shortened RS codes are used, the symbol length can be any value smaller than N_{max} , which provides a great flexibility in RS code design. We use the very common choice of 8-bit per symbol, so that $N_{max} = 255$ bytes. The decoded symbol error probability is given by [56]

$$P_{dec} = \sum_{k=\frac{N-K}{2}+1}^N P_D(N, k, P_s) \quad (3.1)$$

where $P_D(N, k, P_s)$ is the *block error density function*. $P_D(N, k, P_s)$ denotes the probability of k symbol errors within a block of N symbols after RS decoding with channel symbol error probability P_s .

A two-state Markov model [15] is employed to simulate wireless channels with channel state transition probability matrix P defined as,

$$P = \begin{bmatrix} P_{GG} & P_{GB} \\ P_{BG} & P_{BB} \end{bmatrix} = \begin{bmatrix} 1 - P_{GB} & P_{GB} \\ P_{BG} & 1 - P_{BG} \end{bmatrix} \quad (3.2)$$

P_{GB} , P_{GG} , P_{BG} , and P_{BB} are the channel state transition probabilities. The BER of the

channel model, P_e , is given by

$$P_e = \frac{P_{GB}}{P_{GB} + P_{BG}} \quad (3.3)$$

The average burst error length L_B is defined as

$$L_B = \frac{1}{P_{BG}} \quad (3.4)$$

Since P_e is assumed to be known by the video encoder, P_{GB} and P_{BG} can be derived by

$$P_{GB} = \frac{P_e}{(1 - P_e)L_B} \quad (3.5)$$

$$P_{BG} = \frac{1}{L_B} \quad (3.6)$$

After obtaining the transition probability P_{GB} , the symbol error rate (SER) is given by [97]

$$\begin{aligned} P_s &= 1 - (1 - P_e)(1 - P_{GB})^{m-1} \\ &= 1 - (1 - P_e)\left(1 - \frac{P_e}{(1 - P_e)L_B}\right)^{m-1} \end{aligned} \quad (3.7)$$

where m is the number of bits a symbol includes. The derivation of $P_D(N, k, P_s)$ for this model can be found in [98], and it is not included in this thesis as BSC channel model is employed in this chapter. It should be noted that the two-state Markov model includes Binary Symmetrical Channel (BSC) model as a special case by setting $L_B = 1/(1 - P_e)$.

In this case, equations 3.5-3.7 become

$$P_{GB} = P_e \quad (3.8)$$

$$P_{BG} = 1 - P_e \quad (3.9)$$

$$P_s = 1 - (1 - P_e)^m \quad (3.10)$$

$$P_D(N, k, P_s) = \binom{N}{k} P_s^k (1 - P_s)^{N-k} \quad (3.11)$$

For a given BER, the BSC channel is actually the worst channel, yielding the highest probability of error in a video slice, as compared with a channel with correlated errors [99]. In this chapter, we consider video transmission over a BSC channel model in that the design of the rate control scheme faces the toughest challenge when the bitstream is transmitted through a channel with worst condition.

In our scheme, UEP is used to provide different levels of error protection to different sub-bitstreams according to their priorities. Suppose that FEC code rate for the sub-bitstream H which includes header, control bits and motion vectors is r_h , and FEC code rate for the sub-bitstream L which includes texture coefficients is r_l . The rates r_h and r_l are calculated by

$$r_h = K_h/N \quad (3.12)$$

$$r_l = K_l/N \quad (3.13)$$

where K_h is the number of information symbols within the coded sub-bitstream H , where as K_l is the number of information symbols within the coded sub-bitstream L . Suppose the symbol error probability of H after channel decoding is P_{sh} , and the symbol error probability of L after channel decoding is P_{sl} . The corresponding packet loss probability after

channel decoding for the two sub-bitstreams are then represented by P_h and P_l respectively. It should be noted that packet loss ratio depends on decoded symbol loss rate [11], i.e.,

$$P_h = 1 - (1 - P_{sh})^l \quad (3.14)$$

$$P_l = 1 - (1 - P_{sl})^l \quad (3.15)$$

where l is the number of symbols for each packet.

3.3 Problem formulation for rate control in UEP system

3.3.1 End-to-end rate distortion model

In this section, we first present a framework of the classical RD model, which is widely used to solve the rate control problems. Inspired by the fact that it is difficult to establish a RD model which represents the source statistics accurately while being robust to channel errors, we simplify the RD model by considering only the most important parameters and relationships among them. The key idea and corresponding equations of our rate allocation scheme are given at the end of this section.

Source rate distortion model

Following [9] (page 349), a typical video frame distortion D_s is roughly given as the following

$$D_s = \frac{K}{B_s} \quad (3.16)$$

where K is a given constant, B_s (bits) is the number of bits that is required to encode the frame, D_s is the source distortion, which can be derived based on original video frame

and the reconstructed video frame. The formula about D_s is given by Eq. (2.5) which was proposed by He et. al [11].

Channel distortion model

It is assumed that a memoryless channel is used to transfer video packet, each VP consists of a MB, and a VP is carried within two transmission data packets, i.e., packet L and packet H . Therefore, the loss rate of high priority data of a MB is the same as the loss rate of packet H , and the loss rate of low priority data of a MB is equal to the loss rate of packet L . Consider a pixel i in an inter-frame n , in the case that it is correctly received, its reconstruction value $\tilde{f}_n^i$ is equal to $\hat{e}_n^i + \tilde{f}_{n-1}^j$ where $\hat{e}_n^i$ is the residue pixel value after motion compensation and $\tilde{f}_{n-1}^j$ is the reconstruction value of the reference pixel. The probability of the event that a pixel is correctly received is $(1 - P_h)(1 - P_l)$. If packet L is lost, we can reconstruct the residual MB from correctly received neighboring residue MBs, represented by $\hat{e}_n^k$. The probability of losing packet L is $P_l(1 - P_h)$. If packet H is lost, we have to copy the MB at the same location in the previously decoded frame to replace the current MB and the reconstruction value of pixel i is $\tilde{f}_{n-1}^i$. From the above description, we can derive the first and second moments of $\tilde{f}_n^i$ as the following [66]

$$\mathbf{E}\{\tilde{f}_n^i\} = (1 - P_h)(1 - P_l)\mathbf{E}(\hat{e}_n^i + \tilde{f}_{n-1}^j) + \quad (3.17)$$

$$P_l(1 - P_h)\mathbf{E}(\hat{e}_n^k + \tilde{f}_{n-1}^j) + P_h\mathbf{E}\{\tilde{f}_{n-1}^i\}$$

$$\mathbf{E}\{(\tilde{f}_n^i)^2\} = (1 - P_h)(1 - P_l)\mathbf{E}\{(\hat{e}_n^i + \tilde{f}_{n-1}^j)^2\} + \quad (3.18)$$

$$P_l(1 - P_h)\mathbf{E}\{(\hat{e}_n^k + \tilde{f}_{n-1}^j)^2\} + P_h\mathbf{E}\{(\tilde{f}_{n-1}^i)^2\}$$

The total channel distortion is [66]

$$D_c = \sum \mathbf{E}\{(f_n^i - \tilde{f}_n^i)^2\} \quad (3.19)$$

$$= \sum (f_n^i)^2 - 2f_n^i\mathbf{E}\{\tilde{f}_n^i\} + \mathbf{E}\{(\tilde{f}_n^i)^2\}$$

where f_n^i is the value of pixel i in frame n at the encoder side. In this way, the total channel distortion can be derived recursively by using equation (3.17).

End-to-end rate distortion model

The end-to-end distortion of a video communication system can be represented as follows

$$D = f(D_s, D_c), \quad (3.20)$$

where

$$\frac{\partial f(D_s, D_c)}{\partial D_s} > 0 \quad (3.21)$$

$$\frac{\partial f(D_s, D_c)}{\partial D_c} > 0 \quad (3.22)$$

that is, D increases whenever D_s or D_c increases. It should be noted that, in most cases, D cannot be obtained by simply adding D_s and D_c together. In some papers, to simplify the formulation of rate distortion model, it is assumed that D_s and D_c are uncorrelated, e.g., in [11], so that D is sum of D_s and D_c . The distortion representation could be very complex, and kinds of assumption is made in order to avoid sophisticated formulation of end-to-end rate distortion model. Furthermore, from previous section, it is known that in order to estimate channel distortion, additional assumptions are needed, which may not be true in practical situation. Therefore, it is hard to obtain a good end-to-end rate distortion model. However, the performance of conventional joint source channel rate allocation scheme depends heavily on D , as most solutions, e.g., Lagrangian multiplier, need the exact value of D to optimize the number of bits allocated to the video encoder and channel encoder.

It is difficult to obtain an accurate and simple end-to-end rate distortion model for general wireless video communication system, so in this chapter, we do not intend to improve the

estimation accuracy of D , which has been studied intensively in [64][11][66], with different kinds of assumptions and complex derivations. Rather, our objective is to analyze D estimation procedure and find the most important parameters of D which can be controlled to achieve an optimal bits allocation between the video encoder and channel encoder. *Since only the most important parameters are considered, the RD formula can be simplified by omitting those complex equations and recursive algorithms. In the following, those complex equations are also simplified to get the key parameters in D estimation as well as establish a simple and clear relationships among those key parameters.* After substituting equations (3.16) and (3.19) into (3.20), we can derive the following equation

$$\begin{aligned} D &= f(D_s, D_c) \\ &= f\left(\frac{K}{B_s}, \sum (f_n^i)^2 - 2f_n^i \mathbf{E}\{\tilde{f}_n^i\} + \mathbf{E}\{(\tilde{f}_n^i)^2\}\right) \end{aligned} \quad (3.23)$$

Examining above equation, it is clear that $\mathbf{E}\{\tilde{f}_n^i\}$ and $\mathbf{E}\{(\tilde{f}_n^i)^2\}$ can be derived recursively from equation (3.17).

$$D = f\left(\frac{K}{B_s}, G(P_h, P_l)\right) \quad (3.24)$$

where G represents the recursive function that is defined by equation (3.17) and can be used to derive channel distortion based on two channel parameters, P_h and P_l . It is obvious that when transmission channel loses more data packets, i.e., whenever P_h increases, or P_l increases or both quantities P_h, P_l increase, the channel distortion increases. Therefore, in equation (3.24), D increases as P_h increases or P_l increases, and D decreases as B_s increases. If P_h and P_l are correlated, the equations (3.17) and (3.18) may have some additional cross correlation terms. The derivation of D by using equation (3.24) is therefore inadequate. However, it is still correct that D depends on the symbol error rates, P_h and P_l .

3.3.2 Analysis of end-to-end rate distortion

Suppose a wireless channel provides a data transmission rate of u_c to video applications, the total number of bits allocated to a frame, B_{s+c} , is given by

$$B_{s+c} = B_s + B_c \quad (3.25)$$

B_s and B_c are the number of bits allocated to the source video and channel encoder, respectively.

Assume that the length of the sub-bitstream H and sub-bitstream L before channel coding are known to the encoder, and denoted by L_h and L_l respectively. After channel coding, the length of sub-bitstream H and sub-bitstream L are L_h/r_h and L_l/r_l , respectively. The total number of bits for a frame B_{s+c} is distributed between the two sub-bitstreams, i.e.,

$$B_{s+c} = \frac{L_h}{r_h} + \frac{L_l}{r_l} \quad (3.26)$$

Let $\varphi = L_h/(L_h + L_l)$, L_h is given by

$$L_h = \frac{\varphi}{1 - \varphi} L_l \quad (3.27)$$

By substituting (3.27) into (3.26), we have

$$L_l = B_{s+c} \frac{(1 - \varphi)r_h r_l}{r_l \varphi + (1 - \varphi)r_h} \quad (3.28)$$

$$L_h = B_{s+c} \frac{\varphi r_h r_l}{r_l \varphi + (1 - \varphi)r_h} \quad (3.29)$$

The number of bits allocated to the video encoder and channel encoder are represented by

$$B_s = L_l + L_h \quad (3.30)$$

$$\begin{aligned} &= B_{s+c} \frac{r_h r_l}{\varphi r_l + (1 - \varphi) r_h} \\ B_c &= B_{s+c} - B_s \\ &= B_{s+c} \left(1 - \frac{r_h r_l}{\varphi r_l + (1 - \varphi) r_h} \right) \end{aligned} \quad (3.31)$$

Given a total number of bits allocated to a frame B_{s+c} , the end-to-end distortion is given by

$$D = f\left(\frac{K}{B_{s+c} \frac{r_h r_l}{\varphi r_l + (1 - \varphi) r_h}}, G(P_h, P_l)\right) \quad (3.32)$$

By substituting equations (3.1), (3.12), and (3.14) into equation (3.32), we derive an expression of overall end-to-end distortion as follows

$$\begin{aligned} D &= f\left(\frac{K}{B_{s+c} \frac{r_h r_l}{\varphi r_l + (1 - \varphi) r_h}}, G(1 - (1 - P_{sh})^l, 1 - (1 - P_{sl})^l)\right) \\ &= f\left(\frac{K}{B_{s+c} \frac{r_h r_l}{\varphi r_l + (1 - \varphi) r_h}}, G\left(1 - \left(1 - \sum_{k=\frac{N-r_l N}{2}+1}^N P_D(N, k, P_s)\right)^l, \right.\right. \\ &\quad \left.\left.1 - \left(1 - \sum_{k=\frac{N-r_l N}{2}+1}^N P_D(N, k, P_s)\right)^l\right)\right) \end{aligned} \quad (3.33)$$

It is easy for us to know that the end-to-end distortion can be approximated by using B_{s+c} , r_h , r_l , N and P_s .

Considering equations (3.30) and (3.33), we know that r_h and r_l are the two independent key parameters which can be used to compute the end-to-end distortion as well as B_s and B_c , given a fixed B_{s+c} , a channel code block length N and channel BER P_e . In the following, an optimal rate allocation is proposed by controlling the value of r_h and r_l .

An experimental environment is set up to test how the FEC code rates of sub-bitstream H and sub-bitstream L , i.e., r_h and r_l , affect the end-to-end distortion D . The MPEG-4 encoder is used to produce MPEG-4 video bitstream. RS code with different FEC code rates are then used to protect video bitstream. The channel coded MPEG-4 bitstream is transferred through a channel which has a fixed SER. Typical results are shown in Figure 3.3 and 3.4.

Figure 3.3: PSNR comparison with RS code ($K_h, 255$). Video source *Container* is encoded by a MPEG-4 encoder. $u_c = 128kbps$. K_l is fixed to 251.

In those experiments, in order to demonstrate how a FEC code rate of sub-bitstream H or sub-bitstream L affects picture quality, we keep one FEC code rate fixed, while changing the other. The PSNR value is used to denote the end-to-end distortion D . In Figure 3.3, for a given channel SER, the average PSNR increases and reaches a peak value as $r_h = K_h/255$ increases to a threshold from very small value, and decreases as r_h

Figure 3.4: PSNR comparison with RS code ($K_l, 255$). Video source *Foreman* is encoded by a MPEG-4 encoder. $u_c = 256kbps$. K_h is fixed to 201.

continues to increase. It can be observed that there exists an optimum value of r_h for a given channel condition such that D can be minimized. For r_l , the simulation results are similar. From the simulation results, we conclude that the end-to-end distortion can be minimized by allocating appropriate FEC code rates to sub-bitstream H and sub-bitstream L respectively. Consequently, our rate allocation algorithm discussed in the next section will be based on the optimum FEC code rate.

3.4 Adaptive joint source channel rate allocation based on BER control

In this section, we present a new rate allocation scheme for UEP-based video communication systems. Our proposed joint source channel rate allocation scheme is described in the following Section 3.4.4 in detail, and in Section 3.4.1, we discuss our rate control scheme for video encoder alone. In Section 3.4.2, we present a heuristic method to estimate φ , which is obtained by dividing the number of bits in sub-bitstream H by the total number of bits in a frame. In Section 3.4.3, we discuss methods used in computation of target SERs.

3.4.1 Adaptive rate control for the video encoder alone based on quadratic model

As the joint source channel rate allocation algorithm allocate bits to the video encoder based on channel conditions, the number of bits allocated to the video encoder, B_s , varies due to the time-varying feature of the channel bandwidth and BER. Rate control scheme for the video encoder alone is therefore needed to achieve the specific B_s by adjusting the encoder parameters. In this chapter, we use the adaptive rate control scheme proposed in [100] to achieve the target number of bits allocated to the video encoder, B_s , while minimizing the source distortion.

In the following, the algorithm proposed in [100] is reviewed briefly and more details can be found in that paper. In the algorithm, firstly, the expected target buffer level $Tar(n+1)$ for frame n is calculated as

$$Tar(n+1) = Tar(n) + \frac{BUF_s - BUF_{c,I}}{N} \times (0.5 + \frac{N-n}{N+1}) \quad (3.34)$$

where N is the number of P-frames, BUF_s (bits) is the total buffer size. $BUF_{c,I}$ (bits) is

the number of bits in the buffer after coding the first I-frame. n is current P-frame number. The initial value of $Tar(n)$ is

$$Tar(0) = BUF_{c,I} \quad (3.35)$$

$BUF_{c,I}$ (bits) is the number of bits in the buffer after coding the first I-frame. The first I-frame is encoded by using the default quantization parameters. Secondly, the actual number of bits for the n th frame, B_s , is derived as

$$B_s = \max\left\{0, \frac{u_c}{F_r} + Tar(n+1) - \gamma \times Tar(n) + (\gamma - 1)BUF_c(n)\right\} \quad (3.36)$$

where F_r (fps) is the frame rate, γ is an adjusting factor, and $BUF_c(n)$ is the number of bits in the buffer now. u_c (bps) bit rate allocated to video application. It has been proved that (3.36) can reduce the difference between the buffer level and target buffer occupancy [100]. The corresponding quantization parameter is then computed through a well known quadratic rate model, which is given as below [100]

$$R = c_2 \frac{\sigma}{Q^2} + c_1 \frac{\sigma}{Q} + H_{hdr} \quad (3.37)$$

where c_1 and c_2 are the first and second-order coefficients, R is the total number of bits used for encoding a frame, and σ mean absolute difference, is computed using motion-compensated residual for the luminance component, H_{hdr} is the amount of bits used for shape information, motion vector and header. Finally, after encoding the frame with computed quantization parameters, the rate model parameters are updated for following pictures. The total number of actual generated bits are added to the current buffer level to determine the number of skipped frames. Li *et al.* [100] have showed that the scheme described above is adaptive to time-varying channel bandwidth while maintaining a rather good picture quality.

The rate control algorithm proposed in [100] is adaptive to time-varying channel bandwidth, however, it suffers from a disadvantage that it does not consider time-varying channel BER. This chapter proposes a joint source channel rate control algorithm, in which time-varying channel bandwidth and BER are taken into account and UEP is employed to protect video bitstream. This chapter can be considered as an extension study based on [100]. The rate control algorithm proposed in this chapter adapts both the video encoder and channel encoder to the time-varying wireless channel. In the following, several new techniques are introduced in detail, which will be used by the proposed rate control algorithm.

3.4.2 Estimation of output bitstream ratio with sliding window

As we do not know the actual number of bits allocated to the video encoder before encoding process, we cannot obtain the actual value of φ of the frame for the encoder. In this chapter, we propose a sliding window method to solve the problem, in which the previously encoded frame is used to predict φ of the current frame. The video sequences normally have large correlations and the change of quantization step size between two successive frames is limited, therefore, the sliding window method provides an acceptable estimation of φ .

We assume that the current frame number is n . The value of φ of the current frame, $\tilde{\varphi}_n$, is estimated as follows

$$\tilde{\varphi}_n = a \times \varphi_{n-1} + b \quad (3.38)$$

where φ_{n-1} is the actual value of φ of the frame $n - 1$, which is already known by the encoder. a is a factor which represents the relationship between the current frame and the referred frame and b is a compensating factor. a and b are updated using linear regression

method with the data within the sliding window.

Only when the frames in the sliding window have small scene change, the linear regression method can achieve an accurate a and b . The size of the sliding window should be chosen carefully in order to handle both fast motion scenes and slow motion scenes. If the scene of input sequence changes significantly, i.e., the input consists of a high motion scene, a smaller window with more recent data points is used. Otherwise, a larger window with more past data points is used. Mathematically, let the ratio $0 < \varsigma(n) < 1$ being expressed as

$$\varsigma(n) = \min \left\{ \frac{\sigma(n-1)}{\sigma(n)}, \frac{\sigma(n)}{\sigma(n-1)} \right\} \quad (3.39)$$

where $\sigma(n)$ and $\sigma(n-1)$ are the roots of the variance of the n th residue frame and the $(n-1)$ th residue frame, respectively, which are known by the video encoder before the encoding process. The window size W_s is given by

$$W_s(n) = \min \{ W_s(n-1) + 1, \varsigma(n) \times W_{max} \} \quad (3.40)$$

where $W_{max} = 20$ is a preset constant [100]. When video scene changes slowly, equation (3.39) is close to 1, so that equation (3.40) increases the size of sliding window and more frames are stored. On the other hand, when video scene change quickly, i.e., equation (3.39) will becomes smaller, which leads less frames are stored into sliding window. Suppose a scene change occurs at time n . It can be known from equations (3.39) and (3.40) that window size should be reduced greatly at time n . After that, since the video changes slowly, $\varsigma(n+2)$ may be close to 1. Then the window size is increase gradually by using equation (3.40).

To test the algorithm our solution, we encode the *Foreman* QCIF test sequence by using

the MPEG-4 encoder [100] with a fixed quantization parameter $QP = 20$. The frame rate of the encoded sequence is $F_r = 30f/s$. In Figure 3.5, we plot the actual values of φ and the predicted values of φ , $\tilde{\varphi}$, for each frame. It shows that the predicted values are able to track

Figure 3.5: Comparison results between $\tilde{\varphi}$ and $\varphi = \frac{L_h}{L_h + L_l}$ for *Foreman* QCIF. Fixed quantization parameters 20 is used in MPEG-4 video encoder.

and is approximately close to the actual values of φ . Figure 3.6 shows the predicted values of φ when the test sequence is encoded by a MPEG-4 encoder with variable quantization parameter encoding, which is the case in most practical situations. In Figure 3.6, the target bit rate is 96kbps and the quantization parameter is derived using the method described in section 3.4.1. It can be observed that the proposed sliding window method gives a good approximation of φ no matter whether constant quantization parameter or variable quantization parameter is used in the video encoder, and therefore, it can be used in our adaptive joint source channel bit allocation scheme.

Figure 3.6: Comparison results between $\tilde{\varphi}$ and $\varphi = \frac{L_h}{L_h + L_l}$ for *Foreman* QCIF. video is encoded with MPEG-4 encoder at a target bit rate 96kbps , which produces variable quantization parameter for each frame.

3.4.3 Estimation of target symbol error rate

It is important for the video encoder to obtain appropriate target SERs, i.e., P_{t_h} and P_{t_l} , since r_h and r_l are decided by target SERs. In general, a reduction in the value of target SERs decreases the number of bits allocated to the video encoder, which indirectly gives rise to an increase in the source distortion. On the other hand, increasing the value of target SERs increases the channel distortion.

Experiments have been performed to test the end-to-end distortion of video signals when random errors occur in sub-bitstreams H and L and these results are shown in Figures 3.7 and 3.8. In Figure 3.7, a plot of PSNR versus frame index with random bit errors in sub-bitstream L is shown. Examining Figure 3.7, it can be observed that the end-to-end distortion of the video frame is small when BER is equal to or less than 10^{-4} . If BER

Figure 3.7: PSNR of those decoded video frames versus frame index for various BER in sub-bitstream L . The video source is *News*. The encoded MPEG-4 video bitstream contains random bit errors in sub-bitstream L .

in sub-bitstream L is greater than 10^{-4} , the reconstructed video quality degrades greatly. Similarly, a plot of PSNR versus frame index with random bit errors in sub-bitstream H is shown in Figure 3.8. From Figure 3.8, it can be observed that when BER of sub-bitstream H is equal to or less than 10^{-5} , the video quality is roughly equal to that in error-free case. An interesting feature in Figure 3.8 is that the picture quality has a large deviation when BER in sub-bitstream H equals to 10^{-4} . Some of the frames are damaged severely. The reason is that the MBs copied from previously decoded pictures may have large differences to the original MBs. The situation should be avoided because the errors in a decoded frame will propagate to following frames.

From the above experimental results, it is observed that a good picture quality is provided when BER in sub-bitstream L and BER in sub-bitstream H are less than 10^{-4} and 10^{-5} , respectively. We define the threshold of target SER of sub-bitstream L as P_{th} , and the

Figure 3.8: Decoded video PSNR versus frame index for various BER in sub-bitstream H . The video source is *News*. The encoded MPEG-4 video bitstream contains random bit errors in sub-bitstream H .

threshold of the target SER of sub-bitstream H as $P_{thh} = 8 \times 10^{-5}$. They are given by equation (3.10),

$$P_{th} = 1 - (1 - 10^{-4})^8 \approx 8 \times 10^{-4} \quad (3.41)$$

$$P_{thh} = 1 - (1 - 10^{-5})^8 \approx 8 \times 10^{-5} \quad (3.42)$$

In the above equation, $m = 8$ as 8 bit are grouped into a symbol. The acceptable SER in a frame is also discussed by Cote *et al.* [99], who conclude that a slice error rate of above 20% would result in video reproduction quality below 20 dB in PSNR that is typically not usable. In this chapter, it is assumed that each VP contains 1000 bits and each symbol is composed of 8 bits, therefore, each VP contains $l = 2^8 = 256$ symbols. According to (3.14), the target SERs of both sub-bitstreams H and L should be less than 8.7127×10^{-4} in order to provide an acceptable decoded picture quality. It can be observed that our proposed

SER thresholds for sub-bitstreams H and L are within the acceptable target SERs range prescribed in (3.14).

Based on the above two thresholds, we propose a heuristic target SERs search method to obtain P_{t_h} and P_{t_l} within Ω_{p_h} and Ω_{p_l} using pilot channel and pilot slice.

$$P_{t_h} \in \Omega_{p_h} \quad (3.43)$$

$$P_{t_l} \in \Omega_{p_l} \quad (3.44)$$

Ω_{p_h} is a collection of possible target SER values for sub-bitstream H , Ω_{p_l} is a collection of possible target SER values for sub-bitstream L . All the SER values in Ω_{p_l} are less than $P_{t_{lh}} = 8 \times 10^{-4}$, which has been determined from the above experimental results. All the SER values in Ω_{p_h} are less than $P_{t_{hh}} = 8 \times 10^{-5}$.

A pilot slice consists of several VPs coming from sub-bitstreams H and L , and there is a dedicated pilot channel for the pilot slice. The pilot channel suffers from the same error pattern as that in channel for actual video sub-bitstreams. The pilot channel keeps on transmitting pilot VPs during the video transmission session. The detailed P_{t_h} and P_{t_l} search algorithms is given as follow:

Step A. For the frame n , video encoder sets $P_{t_h} = 8 \times 10^{-5}$ and $P_{t_l} = 8 \times 10^{-4}$. Given P_{t_h} and P_{t_l} and channel bandwidth, the available channel transmission rate is allocated between the video encoder and channel encoder. Encode the frame and protect the video sub-bitstreams with appropriate FEC code rates.

Step B. Set P_{t_l} to be one SER value of Ω_{p_l} , set P_{t_h} to be one SER value of Ω_{p_h} , where P_{t_l} and P_{t_h} are temporal SER variables for sub-bitstream L and H , respectively. Given

P_{l_h} and P_{l_l} and channel bandwidth, the available channel transmission rate is re-allocated between the video encoder and channel encoder. Re-encode the frame n and protect the video sub-bitstreams with newly obtained FEC code rates. The first 10 MBs are grouped into pilot slice.

Step C. At the video decoder side, the PSNR of the first 10 MBs, which is produced by the two steps, is compared. The target SERs for the next frame, $n + 1$, is given as

$$P_{t_h} = \begin{cases} P_{l_h}, & \text{if } P_a < P_b \\ P_{t_h}, & \text{otherwise} \end{cases} \quad P_{t_l} = \begin{cases} P_{l_l}, & \text{if } P_a < P_b \\ P_{t_l}, & \text{otherwise} \end{cases} \quad (3.45)$$

where P_a is PSNR of the first 10 MBs generated by **Step A.**, and P_b is PSNR of the first 10 MBs generated by **Step B.**

The above steps are performed iteratively at the encoder side and decoder side to search the best target SERs for sub-bitstreams H and L under a specific channel situation.

For a given channel condition, a typical example of how to determine P_{t_h} and P_{t_l} is shown in Figure 3.9. At first, the video encoder sets $P_{t_h} = P_{l_h} = P_{t_{hh}}$ and $P_{t_l} = P_{l_l} = P_{t_{lh}}$. After encoding process, the first P-frame is protected by FEC code rates obtained in Step 2. The first pilot slice is protected by FEC code rates obtained in Step 3. For the second frame, P_{t_h} and P_{t_l} do not change because PSNR of 10 MBs in the first P-frame is less than that of the first pilot slice, so that there is no feedback from the decoder. On the other hand, the encoder selects new $P_{l_h}(1)$ and $P_{l_l}(1)$ as target SERs for the second pilot slice. For the 3th frame, since $P_{l_h}(1)$ and $P_{l_l}(1)$ lead to a smaller distortion than $P_{t_{hh}}$ and $P_{t_{lh}}$, the video encoder sets $P_{t_h} = P_{l_h}(1)$ and $P_{t_l} = P_{l_l}(1)$, and uses those new SERs to perform rate allocation and calculate FEC code rates. Meanwhile, the target SERs of pilot slice change into $P_{l_h}(2)$ and $P_{l_l}(2)$. These steps are performed iteratively until all target SER

Figure 3.9: Iteratively searching process for P_{th} and P_{tl} based on pilot channel and pilot slice. For each different frame, the pilot slice is protected with a set of different FEC code rates, which is derived from target SERs for sub-bitstream L and H .

combinations are examined.

By using the above method, we can find an optimum set of P_{th} and P_{tl} for a given channel environment. However, there are a few disadvantages of this method: 1) One frame has to be encoded twice, which increases computational complexity of the video encoder. 2) An additional forward channel is required for pilot slice transmission. However, the bandwidth of the additional channel is very small because few bits is required to encode the first 10 MBs of the re-encoded frame. 3) A new search procedure has to be performed if the channel condition changes.

3.4.4 Adaptive source channel rate allocation using decoded BER control component

To accomplish adaptive joint source channel rate control, we first define the following two target SERs, one is P_{th} , which is the target SER of sub-bitstream H after channel decoding; the other is P_{tl} , which is the target SER of sub-bitstream L after channel decoding. The computation of P_{th} and P_{tl} can be found in Section 3.4.3 and they are used to compute

FEC code rates r_h and r_l , respectively.

Our *rate allocation scheme* is performed by several steps and they are summarized as follows:

Step A. Initialization. Construct two look-up tables, e.g., Γ_h that contains possible values of P_s , P_{t_h} and K_h , and Γ_l that contains possible values of P_s , P_{t_l} and K_l .

Step B. Set target SERs for sub-bitstreams H and L to be P_{t_h} and P_{t_l} , respectively.

Step C. Compute r_h given a set of parameters P_s , P_{t_h} , Γ_h , and N .

sub-step 1. Search the look-up table Γ_h with given values of P_s and P_{t_h} to find $\bar{k}_L$ and $\bar{k}_H$ which correspond to P_h and P_{h+1} , respectively. P_h and P_{h+1} are the SERs stored in Γ_h which satisfy two requirements: (i) $P_h \leq P_s \leq P_{h+1}$. (ii) P_{t_h} is the target SER after channel decoding. Let K_h equal to $\bar{k}_L$.

sub-step 2. Calculate SER after channel decoding, P_{dec} , by using K_h , P_s and N .

sub-step 3. If $P_{dec} > P_{t_h}$, use the sub-sampling technique to calculate K_h as $K_h = \max(K_h - 2, \bar{k}_H)$ and then go to sub-step 2 }, else, go to the sub-step 4.

sub-step 4. If $K_h = \bar{k}_H$, go to the sub-step 5, else, let $K_h = K_h + 1$ and then calculate P_{dec} . The final K_h is given by

$$K_h = \begin{cases} K_h; & \text{If } P_{dec} \leq P_{t_h} \\ K_h - 1 & \text{otherwise} \end{cases} \quad (3.46)$$

sub-step 5. Derive the FEC code rate of sub-bitstream H as $r_h = K_h/N$. Update the look-up table Γ_h with the obtained values of K_h , P_{t_h} and P_s .

Step D. Use look-up table Γ_l and other corresponding parameters, P_s , P_{t_l} , and N to compute r_l . The computation of r_l is similar to that of r_h described in Step C.

Step E. Estimate φ of the current frame. Given the total number of frame bits B_{s+c} , r_h , r_l and φ , compute the number of bits allocated to the video encoder and channel encoder, B_s and B_c , by using (3.30).

Step F. Use the video encoder to encode the current frame. The FEC redundant bits are added by the channel encoder. Transmit the coded bitstream through the channel.

Step G. Go to Step B. Repeat until all the video sequences are transmitted.

By using the above steps, the number of bits allocated to the video encoder and channel encoder are adaptive to the time varying channel bandwidth u_c and SER P_s . We will show that the output bitstream is well protected by our proposed scheme and the reconstructed video quality is optimized for a decoder with given error-concealment capability. The adaptive joint source channel rate allocation scheme proposed can be used together with any error control codes [96]. It can be extended to any other error control codes by re-calculating the block error density and error probability after channel decoding.

The proposed joint source channel rate allocation algorithm does not make any assumption on the video encoder and video sequences. It can be applied to general video communication system, in which any video encoder or video sequence can be used. To implement the proposed algorithm does not alter the structure of existing video communication system.

The estimation of output bitstream ratio and adaptive source level rate control based on quadratic model require less computations as compared with estimation of target SER. In order to estimate target SER, the video encoder needs to encode a frame twice. One encoding result is transmitted to the video decoder, and the other one is used to create pilot slice. As discussed in the chapter 2, the encoding process is complex due that the encoder eliminates temporal redundancy and spatial redundancy. However, the computational complexity of the proposed joint source channel rate allocation scheme is still low as compared with some other existing rate allocation schemes for UEP-based video system such as the one proposed in [64]. In [64], in order to obtain rate distortion characteristics of a frame, the frame is encoded iteratively to get the video bitstream with different bit

rate, i.e., the encoding process has to be performed several times. Later, a different channel coding schemes are examined, which results in additional computational complexity. The proposed rate allocation scheme reduces the number of encoding process significantly.

3.4.5 Computational complexity

In this section, we briefly describe the complexity of the proposed algorithm. In equation (3.46), we need to perform an iterative calculation of decoded SER which is based on equation (3.1). For this case, it is difficult to provide an analytical expression for the computational complexity because, in equation (3.1), it depends on how K is chosen. Roughly speaking, the computation complexity of the proposed algorithm increases in N^2 multiplications as K decreases. However, the sub-sample technique can make the final K converge quickly. In addition, the target SERs and corresponding K can be computed and stored in the look-up table. Therefore, the FEC code rates can be computed in advance and the look-up table will be employed to find the appropriate FEC code rates for a given channel condition. The computation that needs to be repeated for each frame is just one look-up table search. In short, the proposed approach may have applications in real-time streaming or video-on-demand systems.

3.5 Simulation results

To assess the performance of our Adaptive Joint Source Channel Bit Allocation scheme (ACBA), extensive simulations have been performed, using various video test sequences such as *News*, *Foreman*, *Container*, *Mother-daughter* etc. These test sequences are provided in QCIF format, and their lengths are of 300 frames. We assume that the first frame is intra-coded, and the rest of the frames are inter-coded. The adaptive video encoder described in Section 3.4.1 is used for source encoding. The two sub-bitstreams generated

by the MPEG-4 encoder, i.e., sub-bitstream H and sub-bitstream L , are protected by FEC codes with different code rates. Particularly, we apply $(K_h, 255)$ RS code to sub-bitstream H and $(K_l, 255)$ to sub-bitstream L , in which K_h and K_l change from VP to VP according to channel conditions. The reason of using $N = 255$ is that the RS code efficiency increases with N , therefore, if applicable, N should be chosen as large as possible [16]. It is assumed that the channel SER of a BSC channel is $\{10^{-3}, \text{ or } 10^{-2}\}$, and the transmission bit rate is 128kbps . In our simulations, we set the collection of possible target SER values for sub-bitstream H and L to be $\Omega_{ph} = \{8 \times 10^{-5}, 10^{-5}, 10^{-6}\}$ and $\Omega_{pl} = \{8 \times 10^{-4}, 10^{-4}, 10^{-5}, 10^{-6}\}$.

The simulation configuration is shown in Figure 3.10. As the channel decoder needs to

Figure 3.10: Simulation configuration to test the performance of adaptive source channel bit allocation algorithm. The priority-based video bitstream is arranged into error control frame for transmission.

know boundaries between sub-bitstreams H and L , which are encoded with different FEC code rates r_h and r_l , we add an error control (EC) layer in our scheme, which is proposed by [101]. Each EC frame consists of an EC header and N_s slots. Each slot contains class 1 sub-slot and class 2 sub-slot. Class 1 sub-slot consists of sub-bitstream H , and class 2 sub-slot consists of sub-bitstream L . The FEC code rate and length of each class is indicated

in EC header. The error-resilient entropy code (EREC) [102] approach is applied to assign different classes of VPs into the N_s slots of an EC frame. The length of EC header is fixed for all the EC frames. Generally, it includes tens of bits for an EC frame that includes many video packets [101]. The length of each class 1 sub-slot and class 2 sub-slot varies from frame to frame.

We apply the universal rate-distortion characteristics based rate allocation scheme [64] to MPEG-4 video transmission system to generate anchors, which are used for comparison purposes (in the following, it is called BM). By using BM, the end-to-end distortion of a particular frame component is firstly evaluated iteratively as a function of the target source bit rate and BER. Then, the BER of a specified channel code is computed as a function of channel BER and FEC code rate. Next, the universal rate-distortion characteristic is constructed for each frame component. After obtaining the universal rate-distortion characteristic for each frame component, a near optimal rate allocation can be achieved [64]. Although BM is too complex to be applicable, it can be used to generate anchors for performance comparison purpose. In this chapter, when BM is used, three source coding rate, which are $0.5u_c$, $0.7u_c$, and $0.9u_c$, where u_c is the data transmission rate, and four target SERs, which are 10^{-6} , 10^{-5} , 10^{-4} and 10^{-3} , are considered.

3.5.1 Results for constant channel condition

We first test the performance of the proposed joint source channel rate control scheme when the channel condition is constant, i.e., both P_s and u_s are constant. The channel SER and bandwidth are assumed to be $P_s = 10^{-3}$ and $u_s = 128k\text{bps}$, respectively. Simulation results are shown in Figure 3.11, including the performances of our proposed scheme based on adaptive UEP, BM, and rate control scheme based on fixed UEP. In the rate control scheme based on fixed UEP, r_l and r_h are set to be r_{l0} and r_{h0} , respectively, for all the video frames, where r_{h0} is the FEC code rate of sub-bitstream H and r_{l0} is FEC code rate

Figure 3.11: PSNR comparison of the proposed ACBA scheme, BM, and rate control scheme based on fixed UEP. The Test sequence is Foreman, which is encoded with a frame rate $f = 30f/s$. $P_s = 10^{-3}$, $u_s = 128kbps$.

of sub-bitstream L . r_{h0} and r_{l0} are obtained from the first P-frame using the BM method.

It can be observed in Figure 3.11 that the performance of our proposed joint source channel rate control scheme based on adaptive UEP is better than that of the rate control scheme based fixed UEP, and approximates the performance of BM. Our proposed rate control scheme leads to 1dB improvement in terms of PSNR as compared with rate control scheme based on fixed UEP. It can also be observed that the proposed scheme outperforms BM for some specific frames. The reason is that when BM is used, the rate allocation is not optimal if the estimated overall distortion is different from the actual distortion.

The corresponding K_l and K_h of the three methods mentioned above are shown in Table 3.1. The results show that the proposed ACBA rate control scheme is effective, leading K_l and K_h to be adapted to channel condition. It can be observed that K_l and K_h which

P-Frame	K_l/K_h ACBA	K_l/K_h BM	K_l/K_h Fixed UEP
1	249/247	247/239	247/239
2	249/247	247/237	247/239
3	247/245	249/241	247/239
4	249/245	249/241	247/239
5	249/245	247/235	247/239

Table 3.1: Comparison of K_l and K_h which are computed by our proposed ACBA scheme, BM, and rate control scheme based on fixed UEP.

are obtained from the rate control scheme based on fixed UEP are constant, and the same FEC code rate is applied to all the video frames.

The average PSNR of 10s video sequence of *Foreman* with frame rate 30f/s, encoded by using MPEG-4 encoder and the three methods mentioned above, transmitted through channels with different channel SERs P_s , is shown in Figure 3.12. It can be observed that, by

Figure 3.12: PSNR comparison of our proposed ACBA scheme, BM, and rate control scheme based on fixed UEP. Test sequence is *Foreman*, which is encoded with 30f/s and $u_s = 128kbps$.

using our ACBA method, the average PSNR is slightly lower than that obtained by using BM. Despite performance variations of some specific frames, the average PSNR obtained by using BM is about $0.5dB$ higher than that obtained by using our method. However, our method is much simpler than BM and capable of providing a good video quality without requiring iterative encoding and a wideband feedback channel. As shown in Figure 3.12, our method leads to about $0.6-1dB$ improvement in terms of PSNR as compared with the rate control scheme based on fixed UEP. This is because the rate control scheme based on fixed UEP is not optimal for some specific frames and leads to a performance degradation.

The computational complexity of BM is much greater than the proposed ACBA. ACBA only needs to calculate decoded SERs based on given channel conditions and FEC code rates. However, BM needs not only evaluate all possible FEC code rates for given channel condition, but also re-encodes and decodes a frame to obtain actual end-to-end distortion and total bit rate. The computational complexity of BM increases exponentially if a new FEC code rate is adopted. In contrast, the computational complexity of ACBA increases linearly as ACBA does not need to re-encode the frame.

3.5.2 Results for variable channel condition

We also test the performance of our proposed scheme when channel SER P_s varies during transmission periods. Suppose P_s changes at frame 51, as shown in Table 3.5.2 The se-

P-Frames	P_s
1-50	10^{-3}
51-300	10^{-2}

Table 3.2: Variation of channel SER P_s during transmission period

quence *Foreman* is encoded by MPEG-4 encoder. It lasts for about 10s and the frame rate is 30f/s. The channel bandwidth u_s is $128kbps$.

Figure 3.13 gives experimental results in PSNR obtained by using our proposed scheme, BM, and the rate control scheme based on fixed UEP. From Figure 3.13, it can be observed

Figure 3.13: PSNR comparison of our proposed ACBA scheme, BM, and the rate control scheme based on fixed UEP. Test sequence is Foreman, which is encoded with 30f/s. channel SER varies during transmission and $u_s = 128kbps$

that the performance of our proposed ACBA scheme approximates that of BM. The excellent performance of our scheme is due to the fact that the FEC code rates obtained in our scheme changes adaptively with the channel condition. As shown in Figure 3.13, the performance of the rate control scheme based on fixed UEP degrades quickly when channel SER changes. In contrast, our method provides a good protection to the transmitted video bitstream even when channel SER changes during the transmission. The average PSNR difference between our proposed scheme and the rate control scheme based on fixed UEP is about 3dB, which is large enough to be perceived in decoded pictures.

K_l and K_h of some P-frames are shown in Table 3.3. It can be observed that K_h and K_l computed by using our proposed scheme change adaptively with channel conditions,

P-Frame	K_l/K_h ACBA	K_l/K_h BM	K_l/K_h Fixed UEP
50	249/247	251/243	247/239
51	249/229	235/217	247/239
52	235/229	243/239	247/239
53	235/229	233/227	247/239
54	235/231	239/225	247/239

Table 3.3: Comparison of K_l and K_h which are computed by using our proposed ACBA scheme, BM, and the rate control scheme based on fixed UEP.

therefore, the rate allocation is also adaptive to time-varying channel.

Figure 3.14 shows the subjective picture quality of *Foreman* obtained by using our proposed ACBA scheme, BM, and the rate control scheme based on fixed UEP. It is evident

Figure 3.14: subjective results for frame 70 at QCIF resolution: 3.14(a) BM, 3.14(b) our proposed ACBA scheme, and 3.14(c) rate control scheme based on fixed UEP.

that a much more stable and acceptable image quality can be obtained by using our proposed scheme as compared with the rate control scheme based on fixed UEP. For example, for frame 70 and the channel SER $P_s = 10^{-2}$, when rate control scheme based on fixed UEP is used, channel impairments cause artifacts as the FEC codes cannot provide enough protection to the transmitted video bitstreams. In contrast, no artifact appears for our proposed scheme.

3.6 Conclusions

In this chapter, we have studied joint source channel rate control schemes for video signal transmission. A new rate allocation scheme which exploits adaptive UEP based on the encoded video sub-bitstream importance and decoded BER in the decoded bitstream is proposed. The adaptive UEP enables the video communication system to protect each sub-bitstream of the encoded video frame according to its priority, so that the most important sub-bitstream has the highest probability to be correctly decoded. FEC is used in our scheme for UEP and the error correction capability is determined by the FEC code rate. Increasing the FEC code rate reduces the error correction capability of the FEC code and leads to an increase of the BER in the decoded bitstream. Therefore, FEC code with high code rate is used to protect sub-bitstreams with low priority. In contrast, FEC code with low code rate is used to protect sub-bitstreams with high priority and leads to a low BER in the decoded bitstream.

This chapter provides a theoretical analysis of the end-to-end distortion of the UEP-based video transmission system, revealing the inherent relationships among key parameters of the video encoder and channel encoder. The BER in the decoded data packets is the key parameters to achieve a performance improvement rate control scheme. In order to achieve the target BER, an appropriate FEC is required for each bitstream with a specific priority. An adaptive joint source channel rate allocation scheme using the BER in the decoded bitstream data as a control component is proposed. In the proposed scheme, an optimal target BER of the decoded bitstream is obtained for a given channel condition by using a pilot channel. Then, by using an error correction code, e.g., RS code, the target BER is employed to compute the FEC code rate for each video sub-bitstream. Finally, the number of bits allocated to the video encoder is computed based on the total available bandwidth and the number of bits allocated to the channel encoder.

The proposed scheme significantly outperforms the rate control scheme based on fixed UEP, especially when the channel SER varies during transmission periods. The PSNR improvements can reach as high as 3dB which can be easily perceived at the decoded picture. The results show that the proposed rate control scheme can change FEC code rate quickly and effectively according to different channel conditions, so that the transmitted video bitstream is well protected and the bandwidth is exploited more efficiently. Experimental results demonstrate that the proposed rate allocation scheme can achieve a comparable performance to that of BM scheme no matter for constant channel condition or variable channel condition. However, our proposed scheme has a much lower computational complexity as compared with BM proposed in [64], as the proposed scheme avoids iterative encoding process.

Chapter 4

A New Rate Model Based on Multiple Logarithmic Functions

4.1 Introduction

In the previous chapter, the joint source channel rate control algorithm allocates a number of bits to the video encoder based on the channel condition. As channel condition changes quickly, the number of bits allocated to the video encoder is time-varying, which poses serious constraint on the acceptable bit rate. It is important for the video encoder to adopt an effective rate control algorithm, so that the video encoder produces the expected number of bits. Not surprisingly, the quality of the encoded video depends heavily on the efficiency of the rate control technique [70]. A quadratic rate control algorithm, which determines encoding parameters according to a quadratic rate model, is employed in the previous chapter to achieve the expected bit rate.

Since an effective rate model is critical to the performance of a rate control algorithm, the modelling issue is studied in detail in this chapters. A new rate model is proposed, which consists of several logarithmic functions that adjust their boundaries and coefficients based

on the statistics of the video sequence during the encoding process. The computational complexity of the proposed model remains low because the video encoder codes each frame only once. The accuracy of the new rate model is greatly improved, compared with some of the current existing rate models. With the proposed rate distortion model, the fluctuations of bit rate estimation can be significantly reduced while maintaining the same picture quality.

The organization of this chapter is as follows. In Section 4.2, the characteristics of inter-frame coding is analyzed. In Section 4.3, a new rate model based on multiple logarithmic function is proposed. The update of the parameters of the new rate model is also addressed in great details. The complexity of the rate model is analyzed. In Section 4.4, the simulation results are presented and conclusions are provided in Section 4.5.

4.2 Image distribution analysis after ME/MC

4.2.1 Motion estimation and compensation

Inter-frame coding uses the block match (BM) motion estimation technique to eliminate temporal redundancy exist in successive frames, where each block in the currently encoded frame is compared with the reference blocks that are in the previous reconstructed frame. A search window is defined by the user to restrict the search area so that all reference blocks are within the window area. A best match block is the one which has a minimum difference to the encoding block in term of SAD or MSE. The principle of block matching is depicted in Figure 4.1.

Generally, only luminance value of the pixel is used in ME/MC process. When the best match block is found, it is then offset by the displacement between its position and the

Figure 4.1: Motion estimation and compensation: one MV is estimated for each block in the actual frame to be encoded. The MV points to a reference block of same size in a previously coded frame. The reference block is in a $N \times Z$ pixel search window.

position of the current encoding block. The displacement vector between the best match block and current encoding block is MV, which is to be transmitted to the video decoder. Suppose, for frame n , the luminance value of the pixel i, j in the current block is denoted by $P_{i,j}(n)$, and the luminance value of the best match block pixel in the previous frame is $\tilde{P}_{u,v}(n-1)$, then the encoder only needs to encode the residue luminance value of the pixel, which is given as follows

$$e_{i,j}(n) = P_{i,j}(n) - \tilde{P}_{u,v}(n-1) \quad (4.1)$$

where $e_{i,j}(n)$ is the residue luminance value of the pixel after ME. The advantage of using residue pixel, rather than original pixel, is that the residue pixel is expected to have a much smaller energy than the original pixel, which requires a fewer number of bits to encode. At the decoder, a reversible procedure is performed to reconstruct the original pixel n correctly by using residue pixel, reference pixel and MV. Similarly, predictions for all the blocks in the current frame are obtained and the prediction frame is constructed.

4.2.2 Entropy bits with known distribution

The residue frame consists of residue blocks produced by motion estimation, and it is transformed by the 2D-DCT transform coding procedure [103] to remove spatial redundancy. The 2D-DCT is performed based on MB. The reason is, in practical, it is impossible to use infinite length transforms to decompose a signal sequence, and a typical approach is to partition a signal sequence into non-overlapped blocks and perform block transformation on each data block, separately [104]. The block of DCT coefficients are assembled to form a zigzag scanned vector $X = [X_0 \dots X_{L-1}]$, which is quantized to compress residue pixels further. The behaviors of a quantizer can be analyzed by the following equations for inputs with known probability distribution [104].

$$H_Q(\epsilon) = \sum_{k=-N}^N P_X[k] \times \log P_X[k] \quad (4.2)$$

where

$$P_X[k] = \int_{(k-1/2)\epsilon}^{(k+1/2)\epsilon} p_X(x) dx \quad (4.3)$$

The $p_X(x)$ is the probability density function of a zero mean independent identically distributed source X . $H_Q(\epsilon)$ is entropy bits produced by this quantizer ϵ . It is well known that the distribution of the residue frame after ME/MC can be modelled by a Gaussian distribution or Laplacian distribution [104]. The probability density function of a zero-mean Gaussian source with variance σ is

$$p_G(x) = \frac{1}{\sqrt{2\pi}} \exp\left\{-\frac{x^2}{2\sigma^2}\right\} \quad (4.4)$$

On the other hand, the probability density function of a zero-mean Laplacian source with variance σ is

$$p_L(x) = \frac{1}{\sqrt{2\sigma}} \exp\left\{-\sqrt{2}\frac{|x|}{\sigma}\right\} \quad (4.5)$$

Thus, we can derive that [104]

$$\begin{aligned} P_L[k] &= \int_{(k-1/2)\epsilon}^{(k+1/2)\epsilon} \frac{1}{\sqrt{2\sigma}} \exp\left\{-\sqrt{2}\frac{|x|}{\sigma}\right\} dx \\ &= \frac{1}{2} e^{-|(k-1/2)\epsilon/\sqrt{2\sigma}|} (1 - e^{-\sqrt{2}\epsilon/\sigma}) \end{aligned} \quad (4.6)$$

Besides above traditional entropy analysis, the rate-distortion theory is another major technique to explore the relationship between entropy bits and a given QP . From rate-distortion optimization (RDO) discussed in [81][105][106] we know that as QP increases, the rate decreases and the distortion increases. Let the Lagrange parameters λ_{mode} , λ_{motion} and the quantizer value QP be given. It was shown via experimental results that the following relationship is efficient for H.263/MPEG-4 [105][106]:

$$\lambda_{mode} = 0.85 \times QP^2 \quad (4.7)$$

The experiment that lead to the relationship has also been conducted for H.264/AVC providing the following equation [106]:

$$\lambda_{mode} = 0.85 \times 2^{(QP-12)/3} \quad (4.8)$$

It is therefore a possibility to derive an optimal QP by using RDO. However, this method suffers from the disadvantages of heavily computational complexity, as RDO is an operational control.

4.3 Proposed logarithmic rate model

Following previous work [70][104], it is assumed that the DCT coefficients of the inter-coding frame are approximately uncorrelated and Laplacian distributed with variance σ^2 . σ^2 is the variance of the residue frame pixels. The empirical entropy of the quantized DCT coefficients is given by [70]

$$H(Q) = \begin{cases} \frac{1}{2} \log_2(2e^2 \frac{\sigma^2}{Q^2}), & \frac{\sigma^2}{Q^2} > \frac{1}{2e} \\ \frac{e}{\ln 2} \frac{\sigma^2}{Q^2}, & \frac{\sigma^2}{Q^2} \leq \frac{1}{2e} \end{cases} \quad (4.9)$$

where $H(Q)$ is the empirical entropy, Q is the quantizer step size. However, we notice that the coefficients of equation (4.9) are constant, which is inappropriate due that video scene changes frame to frame. It is our objective to develop a new rate model, which is in a unified expression and can adapt model coefficients to various frame characteristics. Note that when $\frac{\sigma^2}{Q^2} > \frac{1}{2e}$, $H(Q)$ is also given

$$\begin{aligned} H(Q) &= \frac{1}{2} \log_2(2e^2 \frac{\sigma^2}{Q^2}) \\ &= -\frac{1}{2 \ln 2} \ln(\frac{Q^2}{2e^2 \sigma^2}), \quad \text{where } 0 < \frac{Q^2}{2e^2 \sigma^2} < \frac{1}{e} \end{aligned} \quad (4.10)$$

The rate function (4.10) is further expanded into a Taylor series

$$\begin{aligned} H(Q) &= \frac{-1}{2 \ln 2} \{ (\frac{Q^2}{2e^2 \sigma^2} - 1) - \frac{1}{2} (\frac{Q^2}{2e^2 \sigma^2} - 1)^2 + \bar{R}_3(\frac{Q^2}{2e^2 \sigma^2}) \} \\ &= \frac{-1}{2 \ln 2} \{ -\frac{3}{2} + (\frac{e\sigma}{Q})^{-2} - \frac{1}{8} (\frac{e\sigma}{Q})^{-4} + \bar{R}_3(\frac{Q^2}{2e^2 \sigma^2}) \} \end{aligned} \quad (4.11)$$

where $\bar{R}_3$ represents the remaining quantities after 3 terms. This remainder is small because $(\frac{Q^2}{2e^2 \sigma^2})^3 < \frac{1}{e^3}$. By analyzing equations (4.9) and (4.11), we find that the number of bits can be represented by a similar function no matter whether quantization parameter is large or not. By using a similar function to represent the entropy, the rate model is implemented

and updated more easily than equation (4.9) in that the model parameters could be updated using the same method.

4.3.1 Logarithmic rate model

Based on equations (4.9)-(4.11), we propose a new rate model to evaluate the number of bits using several simple parameters. The formulation is shown as follows

$$H(Q) = \begin{cases} e^{\alpha(1)}(\frac{\sigma}{Q})^{\beta(1)}, & 1 \leq QP < QP_L \\ e^{\alpha(2)}(\frac{\sigma}{Q})^{\beta(2)}, & QP_L \leq QP < QP_H \\ e^{\alpha(3)}(\frac{\sigma}{Q})^{\beta(3)}, & QP_H \leq QP \leq 31 \end{cases} \quad (4.12)$$

where QP is quantization parameter and $Q = 2 \times QP$, QP_L, QP_H are quantization parameter boundaries of local exponential fitting function and $1 < QP_L < QP_H < 31$. $\alpha(n), \beta(n), n = 1, 2, 3$ are coefficients. In this equation, the number of bits measurement is represented as a function of quantization step size and variance of a macroblock (MB). The variance of MB is a used to represent the complexity of the MB. It is general that more bits are required for those complex MBs. The boundaries enable each logarithmic function to fit only a part of actual rate curve.

The $n = 1, 2, 3$ is made empirically, i.e. a actual rate curve is divided into three separate parts and each part corresponds to an independent logarithmic function. We find that three exponential functions are sufficient to accurately fit original rate quantization curves of different video sequence and the computational complexity remains low. The parameters of each logarithmic function are updated independently, although three functions share the same update procedure. There are three independent sliding windows, S_1, S_2, S_3 , which store kinds of useful information used to update parameters.

If QP lies in the n th quantization parameter range, we can rewrite (4.12) into logarithmic expressions

$$\ln H(Q) = \alpha(n) + \beta(n) \ln\left(\frac{\sigma}{Q}\right), \quad n = 1, 2, 3 \quad (4.13)$$

where n corresponds to a quantization parameter range, e.g. $n = 1$ refers to $1 \leq QP < QP_L$ range. After actual encoding process, the encoder has determine the number of bits and quantization step size of MBs in the current frame to update corresponding sliding window. The model parameters are then updated through linear regression analysis [107]

$$\beta(n) = \frac{N(n) \sum_{i=1}^{N(n)} \ln R_i \ln \frac{\sigma_i}{Q_i} - \sum_{i=1}^{N(n)} \ln R_i \sum_{i=1}^{N(n)} \ln \frac{\sigma_i}{Q_i}}{N(n) \sum_{i=1}^{N(n)} (\ln \frac{\sigma_i}{Q_i})^2 - (\sum_{i=1}^{N(n)} \ln \frac{\sigma_i}{Q_i})^2} \quad (4.14)$$

$$\alpha(n) = \frac{\sum_{i=1}^{N(n)} \ln R_i - \beta(n) \sum_{i=1}^{N(n)} \ln \frac{\sigma_i}{Q_i}}{N(n)} \quad (4.15)$$

where $N(n)$ is the total number of MBs observed in the n th quantization parameter range (sliding window), R_i is the actual number of bits of MB i , Q_i is the actual encoding quantization step size, and σ_i^2 is the variance of the i th MB.

4.3.2 Sliding windows

As there are three separate sliding windows, S_1, S_2, S_3 , it is important to insert encoding results of each MB, i.e., Q_i, σ_i, R_i , into appropriate windows in order to update the model coefficients α and β . The algorithm is shown as follows:

If $1 \leq Q_i < Q_L$

Insert R_i, Q_i and σ_i into sliding window S_1

Increase $N(1)$ by 1;

Else If $Q_L \leq Q_i < Q_H$

Insert R_i, Q_i and σ_i into sliding window S_2

Increase $N(2)$ by 1;

Else $Q_L \leq Q_i \leq 31$

Insert R_i , Q_i and σ_i into sliding window S_3

Increase $N(3)$ by 1;

EndIf

$N(n)$, $n = 1, 2, 3$ is the number of MBs available in the i th sliding window.

4.3.3 Boundary optimization

The average prediction error of the n th quantization parameter range can be computed in terms of MAD or MSE.

- In terms of minimum absolute difference (MAD)

$$err_{mad}(n) = \frac{1}{N(n)} \sum_{i=1}^{N(n)} |(e^{\alpha(n)} (\frac{\sigma_i}{Q_i})^{\beta(n)} - R_i)| \quad (4.16)$$

where $err_{mad}(n)$ is the average prediction error of n th quantization range, $N(n)$ is the total number of samples in the n th quantization range, $\alpha(n)$ and $\beta(n)$ are the model coefficients obtained from (4.14), σ_i is the root of the variance of the i th MB sample in the sliding window which is encoded with quantization step size Q_i , and R_i is the number of bits.

- In terms of mean square error (MSE)

$$err_{mse}(n) = \frac{1}{N(n)} \sum_{i=1}^{N(n)} (e^{\alpha(n)} (\frac{\sigma_i}{Q_i})^{\beta(n)} - R_i)^2 \quad (4.17)$$

Generally, the average prediction error is represented in term of MAD as it does not need to calculate the square and it is much easier to be computed as compared with MSE.

The average prediction error of the proposed rate model is given

$$E = \frac{1}{3} \sum_{n=1}^3 err(n) \quad (4.18)$$

$$= \begin{cases} \frac{1}{3} \sum_{n=1}^3 \left\{ \frac{1}{N(n)} \sum_{i=1}^{N(n)} (|e^{\alpha(n)} (\frac{\sigma_i}{Q_i})^{\beta(n)} - R_i|) \right\}, & \text{If MAD is used} \\ \frac{1}{3} \sum_{n=1}^3 \left\{ \frac{1}{N(n)} \sum_{i=1}^{N(n)} (e^{\alpha(n)} (\frac{\sigma_i}{Q_i})^{\beta(n)} - R_i)^2 \right\}, & \text{If MSE is used} \end{cases}$$

The optimal QP_L and QP_H is derived by minimizing E subject to the observed R_i and Q_i in the recently encoded frames. In order to avoid rapid boundary change, we further set a constraint, i.e., $-1 \leq QP_L(n) - QP_L(n-1) \leq 1$ and $-1 \leq QP_H(n) - QP_H(n-1) \leq 1$, where $QP_L(n)$ and $QP_H(n)$ are the optimal QP_L and QP_H for the n th frame, respectively, and $QP_L(n-1)$ and $QP_H(n-1)$ are the optimal QP_L and QP_H for the $(n-1)$ th frame, respectively.

The exhaustive search method is used to find optimal QP_L and QP_H because of its simplicity. Supposed that before encoding current frame the optimal QP_L and QP_H are Q_{preL} and Q_{preH} , respectively. After encoding current frame, an iterative search is done to find the best QP_L and QP_H from a combination of Q_{preL} , Q_{preH} , $Q_{preL}-1$, $Q_{preL}+1$, $Q_{preH}-1$, and $Q_{preH}+1$. We summarize the search procedure as follows:

Step A: Set $QP_L = Q_{preL} - 1, QP_H = Q_{preH} - 1$. The distortion D_q is set to be 10^{10} . Set $QP_{oL} = Q_{preL}$, and $QP_{oH} = Q_{preH}$. QP_{oL} and QP_{oH} is used to stored temporal optimal QP_L and QP_H .

Step B: Update the sample data in three separate sliding windows $S1, S2, S3$ based on new boundary values QP_L and QP_H . The boundary values changes, the number of sample data in each sliding window also changes.

Step C: $\alpha(n), \beta(n), n = 1, 2, 3$ are computed using the sample data in $S1, S2, S3$, respectively. Finally, E is derived from equation (4.18). The optimal boundary values are

given as follows:

$$QP_{oL} = \begin{cases} QP_{oL}, & \text{If } E \geq D_q \\ QP_L, & \text{Otherwise} \end{cases} \quad (4.19)$$

$$QP_{oH} = \begin{cases} QP_{oH}, & \text{If } E \geq D_q \\ QP_H, & \text{Otherwise} \end{cases} \quad (4.20)$$

At the same time, if $E < D_q$, D_q is set to E .

Step D: Change the boundary values, i.e., QP_L or QP_H , go to Step 2.

The complexity of the boundary search depends on the search range of boundary and the number of multiple logarithmic functions that the proposed rate model uses. It is obvious that there are $N_{log} - 1$ boundaries need to be optimized in case that the proposed rate model has N_{log} piecewise functions. Supposed that the search range of boundary is N and there are N_{log} logarithmic functions, the possible combination of search space is $N^{N_{log}-1}$. For each set of selected combination of boundary values, we need compute the parameters of each logarithmic functions and their prediction errors $err_{mad}(n)$ or $err_{mse}(n)$, which has been studied in detail in previous section.

In this thesis, $N_{log} = 3$. As the search range of QP_L is $QP_{preL} \pm 1$, and QP_H is $QP_{preQ} \pm 1$, where QP_{preL} and QP_{preQ} are the most recent optimal QP_L and QP_H , respectively, so each optimal boundary is selected from 3 possible values, and the search size is $3^{N_{log}-1} = 9$. The computational complexity of exhaustive search remains low as compared with ME/MC in the encoding process, as for ME/MC each MB needs compare with neighboring ± 32 MBs to find the best matching MB.

4.4 Simulation results

To assess the performance of the new rate distortion model, numerous simulations have been performed using the standard picture test sequences in QCIF format: “foreman”, “silent”, and “news” etc. We implemented logarithmic rate model based on H.263 [108], by adding the logarithmic functions and parameters update procedure.

The proposed model is compared with TMN8 model [70] and quadratic model [109]. The TMN8 model is given by

$$H(Q) = AK \frac{\sigma^2}{Q^2} \quad (4.21)$$

where $A = 16^2$, K is an adaptive factor, which is updated using a recursive method [70]. The quadratic rate model is proposed by Lee et al. [62],

$$H(Q) = a \frac{\sigma}{Q} + b \frac{\sigma}{Q^2} \quad (4.22)$$

where a and b are adaptive parameters, which are updated using linear method [109]. For fair comparison, the same Q and σ are used for all models. The first I-frame is encoded using default $QP = 20$ and the rest frames are Inter-coded. For logarithmic rate model, we initially set $QP_L = 5$, $QP_H = 15$.

4.4.1 Results of fixed quantization parameter

First, we tested estimation accuracy of the rate distortion models using constant QP . Figure 4.2-4.3 presents the comparison results together with the actual rate curve. It is evident that the proposed method produces smaller estimation errors than the other two models. The rate curve obtained by using our method approaches the original rate curve. The derivation of TMN8 model is based on the assumption $\frac{\sigma^2}{Q^2} \leq \frac{1}{2e}$, i.e. large Q . On

Figure 4.2: Bit number comparison among original H.263 encoder output, logarithmic rate model, TMN8, quadratic model for continuous frames of 'foreman.qcif'. $QP = 3$ for all MBs in P-frames.

Figure 4.3: Bit number comparison among original H.263 encoder output, logarithmic rate model, TMN8, quadratic model for continuous frames of 'foreman.qcif'. $QP = 20$ for all MBs in P-frames.

the other hand, quadratic model is suitable for small Q . Our model considers both cases, and the boundary of each logarithmic function can adapt to the statistics of the encoded frames, therefore it outperforms the other two models.

4.4.2 Results of variable quantization parameter

It is interesting to show the estimation accuracy of all models when QP is varying from MB to MB, which is often the case in a practical system. In the following simulations, QP varies over all 31 values, starting from 1, increasing 1 MB by MB until 31, then decreasing 1 MB by MB until 1. The loop continues until all MBs are encoded. 100 frames are encoded for each test sequence. For fixed boundary logarithmic rate model (in the following, it is called fixed model), we kept $QP_L = 5, QP_H = 15$ regardless of the image statistics. Figure 4.4 shows the bit number estimation comparison when QP varies. It is

Figure 4.4: Bit number comparison among original H.263 encoder output, fixed model, our method, TMN8, and quadratic method for continuous frames of 'foreman.qcif'. QP varies from MB to MB.

clear that the accuracy of the proposed rate distortion model is greatly improved. The

optimal QP_H and QP_L are shown in Figure 4.5. The optimal QP_L and QP_H are found by

Figure 4.5: The boundary values of logarithmic rate model for continuous frames of 'foreman.qcif'. QP varies from MB to MB. The QP_L and QP_H are updated after encoding one frame.

exhaustive search method minimizing E . The relative prediction error of a frame is defined as $\sum_{i=1}^{99} (|O_i - R_i|) / \sum_{i=1}^{99} R_i$, where O_i is the number of bits predicted by each model.

The average relative prediction error per frame is sum of the relative prediction error of all frames divided by the number of encoded frames. Table 4.1 list a comparison of average relative prediction error per frame using different test sequences. Result shows that by

video sequence	tmn8	quadratic	fixed	logarithmic
foreman	0.749	0.394	0.405	0.204
silent	1.185	1.412	2.05	0.454
news	2.589	2.472	3.884	0.836
mother and daughter	0.747	1.143	2.687	0.352

Table 4.1: Comparison of average relative prediction error per frame among TMN8, quadratic model, fixed model, and logarithmic rate model. QP varies from MB to MB.

using the proposed rate distortion model, the average relative prediction error per frame is greatly reduced. At the same time, we notice that the performance of the fixed model varies greatly depending on statistics of test sequence because the boundaries are not optimal.

4.5 Conclusions

A good rate model can improve the performance of a rate control algorithm as the rate control algorithm derives the encoding parameters based on the rate model. In this chapter, an improved rate model, which achieves a more accurate estimation to the number of bits as compared with some existing rate models, is proposed. By using the proposed rate model, the rate control algorithm has a higher probability to control encoding parameters better to achieve the target number of bits as well as minimize source distortion caused by quantization.

The proposed rate model is based on multiple logarithmic functions, and it exploits a local curve fitting technique. In order to match the actual rate distortion data accurately, three separate logarithmic functions are employed in the new model, which consists of all quantization ranges. Furthermore, the boundaries of logarithmic functions are adaptive to statistics of previously encoded frames, and the coefficients of the logarithmic functions are derived from previously encoded frames. Experimental results show that the proposed model provides a more accurate estimation to the number of bits than existing rate models.

Chapter 5

Fast Pre-encoding Component and Its Application

5.1 Introduction

In Chapter 4, the performance of a novel rate model, which exploits useful information of previously encoded frames, has been examined. In this chapter, a novel encoding structure which is based on a fast pre-encoding component is proposed for the new emerging video encoder, i.e., H.264 video encoder, to implement rate control algorithm. The general rate control structure for most previous video standards such as H.263 and MPEG-4 is described by Wang in [18], and it is further studied in [70]. The rate control structure is simple yet effective and it is widely deployed by those encoders to achieve a target bit rate. However, with the new compression and optimization techniques are introduced to the new emerging video encoder to improve compression efficiency, it is not so easy to implement an rate control algorithm in conventional rate control structure. The new challenge, which is discussed in the next section, must be addressed, so that most of existing rate control schemes could be used directly by the new video encoder. The fast pre-encoding component is therefore proposed to solve the problem, in which, prior to actual encoding, a pre-encoding process

is firstly performed by the encoder to retrieve some useful information of the current frame that can be used by the encoder for actual encoding process.

In Section 5.2, the conventional encoding structure with rate control schemes is reviewed and related work on successful rate control algorithms is described. In Section 5.3, we discuss rate control problems for the new emerging video encoder. A resolution to the problem based on co-relationship between the successive frames is also reviewed. In Section 5.4, our proposed rate control structure is described in detail. The performance of the new rate control structure is compared with the existing linear based solution in Section 5.5. 5.6 concludes the chapter.

5.2 Conventional rate control structure

Consider the video communication system with limited buffer size and delay, a buffer regulation scheme is usually needed. Several parameters of the encoder may be used to control the bit rate generated by the video encoder. The conventional rate control structure is the backward buffer control structure illustrated in Figure 5.1. In this structure, a

Figure 5.1: Buffer feedback-based rate control for H.263, MPEG-4 video encoder. After coding each video frame, the buffer level is measured and feedback to the video encoder.

rate control algorithm is based on the feedback from the encoder buffer, and controls the encoding parameters such as quantization parameter of the encoder. A simple example of the rate control scheme is that the quantization parameter QP is derived from buffer level without considering characteristics of the encoded frame. Kim [110] suggests a non-linear relationship between the buffer occupancy and QP , and it is given by

$$QP(n) = \begin{cases} \alpha \left(\frac{b(n-1)}{\alpha} \right)^k, & 0 < b(n-1) < \alpha \\ 1 - (1 - \alpha) \left(\frac{1-b(n-1)}{1-\alpha} \right)^k, & \text{otherwise} \end{cases} \quad (5.1)$$

where $b(n-1)$ is the normalized buffer occupancy for the $(n-1)$ th MB, $QP(n)$ is the normalized quantization parameter for the n th MB. A linear relationship between the buffer occupancy and QP is also suggested and is given by

$$QP(n) = b(n-1) \quad (5.2)$$

The performance of the rate control scheme could be improved by incorporating the characteristics of the encoded frame into QP computation. The example is given by Oh *et al.* [111], in which an adaptive QP-rate table which adapts to the characteristics of encoding sequence is built and is used in the rate control scheme. The rate control achieves a better performance than those rate control schemes without considering the characteristics of the encoding sequence.

It is well known that the motion compensated frame follows Gaussian, or Laplacian distribution with signal variance σ^2 , and the rate model is represented by [104]

$$B(Q) = \frac{1}{\alpha} \log_e \left(\epsilon^2 \times \beta \times \frac{\sigma^2}{Q^2} \right) \quad (5.3)$$

where Q is quantizer step size, α , β and ϵ are adjusting parameters. The variance of motion

compensated frame, i.e., MSE or MAD, is widely used by different rate control schemes [70][75] to compute quantizer step size. The larger the MSE or MAD is, the more complex the MB is, and more bits should be allocated to it. The reason is that the MB with large signal variance normally contains more motion and scene changes, which need more bits to encode in order to prevent picture quality from degrading.

5.3 Challenge of the new emerging video encoder

Overview of emerging video encoder

Since 1997, the ITU-T's Video Coding Experts Group (VCEG) has been working on a new video coding standard with the internal denomination H.26L. In late 2001, MPEG's video group and VCEG decided to work together as a JVT and to create a single technical design for a forthcoming ITU-T Recommendation, ITU-T H.264, and for a new part of the MPEG-4 standard, MPEG-4 part 10, based on the current working draft of JVT/H.26L coding. The official title of the new standard [5] is AVC. However, it is widely known by its old working title, H.26L, and by its ITU document number, H.264. The primary goals of the H.264 project are improved coding efficiency, improved network adaptation and simple syntax specification [112].

The H.264 design covers a Video Coding Layer (VCL), which provides a core high-compression representation of the video picture content, and a Network Adaptation Layer (NAL), which packages that representation for delivery over a particular type of network. The rate control algorithm works in VCL layer and controls the parameters of coding options such as QP and skipped frames etc. The H.264 VCL is basically similar to the compression process of prior standards, e.g., H.263, in that it is a block-based motion-compensated hybrid transform video coder. Figure 5.2 shows a block diagram of basic VCL coding structure of

H.264.

Figure 5.2: Basic coding structure of H.264. The RDO is used to select the best encoding mode, reference frame and motion vectors. At most 5 frames is kept in the frame stored buffer for ME/MC.

It is obvious that the basic coding structure of H.264 is similar to most of previous encoders such as H.263 and MPEG-4, which is illustrated by Figure 2.2 in chapter 2. H.264 consists of ME and MC block, transform block, quantization block and entropy encode block. The ME/MC is used to eliminate temporal redundancy between successive frames. The transform block removes spatial redundancy between different blocks in a frame. Video pictures are divided into MBs of 16×16 luminance samples each, with two associated 8×8 chrominance samples [113]. Each MB is encoded in *intra* or *inter* mode. *inter* coding employs prediction from other previously decoded pictures. The encoding process for *inter* prediction consists of choosing motion vectors, reference pictures, and a spatial displacement that is applied to all samples of block. The motion vectors which are transmitted as side information are used by the encoder and decoder to simultaneously provide the *inter*

prediction signal.

A number of new features are introduced to H.264 in order to achieve the primary goals of the standard, such as *intra* prediction, long-term ME/MC and so on. In *intra* prediction, each sample of a block in a *intra* MB is predicted using spatially neighboring samples of previously coded blocks. The encoder chooses which and how neighboring samples are used for intra prediction, which is simultaneously conducted at the encoder and decoder using the transmitted intra prediction side information [114]. In long-term ME/MC case, then the video encoder can use any previously encoded frame which are between $n - 1$ to $n - 5$ as the reference frame for the n th frame, where n is the frame number. Long-term ME/MC can let the H.264 encoder select the best reference frame, so that a good residue frame is produced.

Furthermore, H.264 supports motion compensation block size ranging from 16×16 to 4×4 luminance samples for a MB with many options between the two, called *prediction mode of a MB* (PM). The PM decision is performed together with ME/MC, reference selection by choosing the right mode, reference and motion vectors which results in minimum sum of absolute difference (SAD) [113].

Last but not the least, the transform used in H.264 is applied to 4×4 blocks, and instead of a 4×4 discrete cosine transform (DCT), a separable integer transform is used. For the quantization of transform coefficients, H.264 uses scalar quantization. One of 52 quantization parameters is selected for each MB, which is denoted by QP . In H.264, two methods of entropy coding are supported, namely context-adaptive variable length coding (CAVLC) and context-adaptive arithmetic coding (CABAC).

In H.264 video encoder, one of the five previously decoded frame is selected and used as

a reference to find an optimal MB PM among 7 possible MB PMs. This results in 35 combinations of block sizes and reference frames. There are two non-normative methods, i.e., simple method and complex method [115], to obtain PM and corresponding motion vectors. The complex method is more attractive than the simple method, since it requires much less bits than the simple method to encode a picture but provides similar picture quality as that provided by the simple method, and it needs only a slightly longer time to encode the same sequence. Figure 5.3 shows the number of bits per frame encoded by using simple method and complex method, respectively. Figure 5.4 shows PSNR per frame

Figure 5.3: Bits comparison for simple method and complex method. The sequence name *Foreman*. The Sequence is encoded with H.264 video encoder. Frame rate $F_r = 30f/s$. Quantization parameter is 30 for all the frame. Reference number is 5.

encoded by using simple method and complex method, respectively. Figure 5.5 shows encoding time per frame of simple method and complex method, respectively.

The core technique of complex method is rate-distortion optimization (RDO). The rate-distortion problem is solved by using Lagrangian multiplier. However, the application of

Figure 5.4: PSNR comparison for simple method and complex method. The sequence name *Foreman*. The Sequence is encoded with H.264 video encoder. Frame rate $F_r = 30f/s$. Quantization parameter is 30 for all the frame. Reference number is 5

Figure 5.5: Encoding time comparison for for simple method and complex method. The sequence name *Foreman*. The Sequence is encoded with H.264 video encoder. Frame rate $F_r = 30f/s$. Quantization parameter is 30 for all the frame. Reference number is 5.

Lagrangian technique to control a hybrid video encoder is not straightforward because of temporal and spatial dependencies of the rate distortion costs. Given the Lagrangian multiplier λ_{mode} and the quantizer value Q , the Lagrangian mode decision for a MB S_k is to minimize the cost associated with PM I_k and Lagrangian multiplier λ_{mode} , which can be expressed as

$$J_{mode}(S_k, I_k|Q, \lambda_{mode}) = D(S_k, I_k|Q) + \lambda_{mode}R(S_k, I_k|Q) \quad (5.4)$$

where I_k is the prediction mode which varies over a sets of possible PMs, $D(S_k, I_k|Q)$ is the distortion which is measured as the sum of squared differences (SSD) between reconstructed and original MB pixels, $R(S_k, I_k|Q)$ is the actual rate after encoding the MB using selected I_k . More details can be found in [105][106].

After taking rate control scheme into account, the video encoder becomes more complex. A simplified H.264 encoder with rate control component is shown in Figure 5.6. In order

Figure 5.6: Simplified H.264 video encoder with rate control component. It is obvious that rate control block needs to know MAD or MSE to perform rate control and compute quantization parameter. On the other hand, the RDO block needs to know quantization parameter.

to derive the best PM for a MB, a Lagrangian multiplier should be given firstly. However,

the lagrangian multiplier is computed through quantization parameter QP . For example, in [106], Wiegand *et al.* computes λ_{mode} for H.264 encoder as follows

$$\lambda_{mode} = 0.85 \times 2^{(QP-12)/3} \quad (5.5)$$

where λ_{mode} is the Lagrangian multiplier for PM decision. Several other optimal Lagrangian multiplier for different video standards are given in [105][106]. The derivation of optimal Lagrangian multipliers for different video standards is also studied in [116]. As we have discussed above, the rate control performance is improved significantly by taking characteristics of the encoding frame into account. A typical rate model is given as follows,

$$R = G(\sigma, \alpha, \beta, QP) \quad (5.6)$$

where R is the number of bits, α and β are adjusting factors, σ is MAD or MSE value and $G(\cdot)$ has been studied in the previous chapter. It is obvious from the above equation that for given α , β , σ and QP , the number of bits of an MB can be approximated by using a rate model $G(\cdot)$, e.g., the multiple logarithm rate model derived in the previous chapter.

Linear MAD prediction model

From Figure 5.6, it can be observed that the MAD or MSE of a MB is derived after performing RDO because MAD or MSE are computed based on the motion-compensated frame. The legitimate question might be: *What value of MAD should be used to derive QP before RDO is done?*

In order to solve the above problem, the closed loop shown in Figure 5.6 need to be broken. One possible solution is to use QP of previously encoded MB to do RDO [117]. The other solution is to use a predicted MAD or MSE to perform a rate control to derive QP for RDO and later quantization process [118][119]. Using QP of previously encoded MB

usually leads to iterative encoding process to search optimal QP for RDO and quantization. The reason is that the complexity and characteristics of each MB vary greatly from neighboring MBs. In contrast, predicting MAD or MSE of the current MB from previously encoded MBs avoids iterative searching process and provides a good estimation of QP. One of typical methods to predict MAD or MSE of the MB is linear MAD prediction model, which has been discussed in the standard rate control of H.264 encoder [118].

To predict MAD of the current MB, a linear model is introduced by Li [118][119], in which MAD of the current MB is predicted by using MAD of the co-located MB in the previously encoded frame, and the model is given by

$$MAD_c = a_1 \times MAD_p + a_2 \quad (5.7)$$

where MAD_c is the predicted MAD of the current MB, MAD_p is the actual MAD of the co-located MB in the previously encoded frame, a_1 and a_2 are two coefficients of prediction model, which are derived from previously encoded frames by using linear regression method. The co-located MB is defined as follows: suppose current MB is at the i th column and j th row in the encoding frame, then the MB at the i th column and j th row in the previously encoded frame is considered as co-located MB of current MB.

Although linear MAD prediction model represent a major step to obtain a good rate control scheme for the new emerging H.264 encoder, it suffers from the drawback that it requires the current MB and co-located MB in the previously encoded frame have a strong co-relationship between each other, which may not be true in practical situation. This prediction of MAD becomes less accurate when the motion in a frame becomes fast and complex or when there is a scene change. Recently, M. Jiang *et al.* [120] proposed an

improved MAD estimation based on frame complexity. The MAD prediction is given as

$$MAD_p = \alpha \times MAD_l + (1 - \alpha) \times MAD_{psnr} \quad (5.8)$$

where MAD_p is the final predicted MAD value, MAD_l is the MAD value predicted using linear method discussed above, MAD_{psnr} is the estimated PSNR drop ratio of current frame. $\alpha = 0.7$ is a constant. This method considers frame complexity by measuring the PSNR drop ratio. It is easy to understand that the linear MAD prediction value affect the final MAD prediction value significantly. Therefore, the improved MAD estimation suffers from the same disadvantages as the linear MAD prediction. In the following, a new MAD prediction method is proposed, which does not need such requirement, and it gives a more accurate MAD prediction.

5.4 Pre-encoding structure for new video encoder

5.4.1 Analysis of video encoder

The dilemma problem of rate control for H.264 is solved by the linear MAD prediction model described in the previous section. However, the scene content in a video sequence varies quickly, and MBs in the residual frame change drastically depending on their spatial complexity and temporal difference. The instant change in scene content could lead to very severe consequences on MAD estimation. Furthermore, PM of the current MB may be different from that of the co-located MB in the previous frame and the variation of PMs further makes MAD prediction unstable.

Considering the tree structured motion compensation for a MB of the P-frame. Each MB (16×16 pixels) may be predicted by using 1 of 7 PMs [113] listed below

Index	PM name	height x width (pixels)
1	Intra4X4	4 x 4 pixels
2	Intra16	16 x 16 pixels
3	Skip	
4	Inter16X16	16 x 16 pixels
5	Inter16X8	16 x 8 pixels
6	Inter8X16	8 x 16 pixels
7	Inter8X8	8 x 8 pixels

Table 5.1: Summary of Index, PM names and their height and width in terms of pixels. For example, given index 4, the prediction mode of the MB is Inter16X16, therefore the MB is Inter-coding and ME/MC is done in 16×16 search window.

In case that the Inter8X8 PM is used, each block (8×8 samples) can be predicted by using 1 of 5 sub-modes listed below [113] These partitions and sub-partitions results in a large

Index	PM name	height x width (pixels)
s1	Intra4X4	4 x 4 pixels
s2	Inter8X8	8 x 8 pixels
s3	Inter8X4	8 x 4 pixels
s4	Inter4X8	4 x 8 pixels
s5	Inter4X4	4 x 4 pixels

Table 5.2: Summary of index, sub-PM names and their height and width in terms of pixels.

number of possible combinations in prediction mode. The actual PM of a MB is computed by using RDO which is explained in previous section 5.3.

Define π_i as follows

$$\pi_i = \frac{N_{MB,i}}{N_{MB}}, i = 1, 2, 3, 4, 5, 6, 7 \quad (5.9)$$

where i is the PM index obtained by using RDO, $N_{MB,i}$ is the number of MBs that are predicted by using PM with index i , N_{MB} is the number of total encoded MBs. π_i represents the percent of MBs that are predicted by using PM with index i among all encoded MBs.

Figure 5.7, which includes five sub-figures, shows the values of π_i for different video test sequences *Foreman*, *Carphone*, *Silent*, *Container*, *Mother and daughter*. In Figure 5.7, the JM6.1 encoder [1] is used as the H.264 video encoder to encode video test sequences. The quantization parameter QP varies over six values: 10, 20, 28, 30, 35, and 45, producing six normalized plots for each specified video test sequence. Each video test sequence includes 100 frames. As observed in all the five sub-figures, the peaks of π_i occur when i is equal to 3, 4 and 7, which correspond to PM Inter16X16, Skip and Inter8X8. It can also be observed from the five sub-figures that π_7 grows monotonously as QP decreases and π_3 and π_4 grows monotonously as QP increases. Those characteristics are confirmed by results shown in Figure 5.8 where relationship between π_7 and QP is shown, and in Figure 5.9 where relationship between π_3 , π_4 and QP is shown. It can be observed from these figures that more than 85% of MBs are predicted by these three PM modes, i.e., Inter16X16, Skip and Inter8X8, no matter what QP is used.

5.4.2 New encoding structure based on fast pre-encoding

A MAD prediction method based on Fast-Pre-Encoding (FPE) component is proposed in this section to replace linear MAD prediction method, and the new method eliminates mismatch of the predicted MAD value produced by the linear MAD prediction model. The encoding system based on FPE is shown in Figure 5.10. The new encoding structure consists of two additional components, i.e., a fast pre-encoding component and a MAD calculation component. In the fast pre-encoding component, only 2 PMs among the 7 PMs mentioned in the previous section, i.e., Inter16X16 and Skip, are used to perform RDO. The key function of FPE component is to derive rough MAD value and other useful information about the characteristics of the current MB by using only two PMs. Since the MAD value and the useful information are derived from the current MB rather than predicted from previously encoded MB, the information reflect actual characteristics of the current

Figure 5.7: π_i vs. modes with various values of QP . The prediction modes are gathered after encoding 100 frames of each video sequence. The video sequences from top to bottom are: *Foreman*, *Carphone*, *Silent*, *Container*, *Mother and daughter*. The *Foreman*, *Carphone*, and *Silent* are encoded with RDO on. The *Container*, *Mother and daughter* are encoded without RDO.

Figure 5.8: $\pi_i, i = \text{Inter8X8}$ vs. QP for video sequences *Foreman*, *Carphone*, *Silent*, *Container*, *Mother and daughter*. The *Foreman*, *Carphone*, and *Silent* are encoded with RDO on. The *Container*, *Mother and daughter* are encoded without RDO.

MB. The MAD prediction method based on FPE component is better than linear MAD prediction model in that it is not based on the assumption that the previously encoded MB tightly relates with the current MB, which may not be true in practical situations.

One of the other core component in the proposed encoding structure is the MAD calculation block, which is behind the FPE component. In MAD calculation, a mathematical model is built in order to predict an accurate MAD of the current MB based on the rough MAD value produced by using mode Inter16X16 or Skip. The predicted MAD of the i th MB is given by the following equation

$$\bar{m}_i = \begin{cases} \mu \times M_i + \nu, & \text{if } (M_i \geq (\gamma_n \times M_{ave})) \\ M_i, & \text{otherwise} \end{cases} \quad (5.10)$$

Figure 5.9: $\pi_i, i = \text{Inter16X16, Skip}$ vs. QP for video sequences *Foreman*, *Carphone*, *Silent*, *Container*, *Mother and daughter*. The *Foreman*, *Carphone*, and *Silent* are encoded with RDO on. The *Container*, *Mother and daughter* are encoded without RDO.

Figure 5.10: Proposed H.264 encoding structure with rate control based on FPE component. The FPE component is responsible for performing RDO using two PMs, i.e., Inter16X16 and Skip, and deriving MAD information and number of overhead bits.

where M_i is the rough MAD value of the i th MB generated from FPE component. μ and ν are adjusting factors and they are updated by using linear regression method after encoding

a frame. M_{ave} is given by

$$M_{ave} = (\sum_{i=1}^N M_i) / N \quad (5.11)$$

where N is total MBs of a frame. γ_n is calculated recursively by

$$\gamma_n = (1 - \alpha)\gamma_{n-1} + \alpha\zeta \quad (5.12)$$

where $\gamma_0 = 1$, α is moving average parameter (0.75), and ζ is defined as follows

$$\zeta = \sum_{M_i > m_i} M_i / s / M_{ave} \quad (5.13)$$

where m_i is the actual MAD of the i th MB, s is the number of MBs which meet the constraint $M_i > m_i$.

The detailed encoding process based on FPE component is summarized as follows:

Firstly, the encoder performs ME/MC and RDO to choose a PM between two PMs (Inter16X16 and Skip). The chosen PM leads to a minimum source distortion D_{e1} with the corresponding rough MAD value M_i . The average value of QPs of the previously encoded frame are used as QP for the current MB. After that, the encoder stores the chosen PM p_i , the minimum distortion D_{e1} , and the corresponding MAD value M_i .

Next, the MAD calculation component is employed to predict the MAD value of the current MB from M_i . The predicted MAD value is then used by a rate control algorithm to compute QP for the current MB.

Finally, RDO is performed again by using the QP obtained in the second step to choose a

PM among the rest 5 PMs which are not considered in the first step. The obtained source distortion D_{e2} is compared with D_{e1} obtained in the first step to decide the optimal p_i which leads to minimum source distortion.

The RDO is performed by two sub-steps. One is performed in the FPE component and the other is after rate control. In the first sub-step, RDO is performed to choose an optimal PM between two PMs and meanwhile produce some useful information of the current MB, such as rough MAD value, to the following steps. In the second sub-step, RDO is performed by using the optimized QP obtained in the rate control step and an optimal PM is chosen among the rest 5 PMs. As in the first sub-step, only 2 possible PMs are considered, RDO can be performed very fast.

5.4.3 Application of MAD prediction method based on FPE component

The linear MAD prediction mentioned above is used by the standard rate control scheme for the H.264 video encoder [118][119]. In this section, the standard rate control scheme is improved based on new proposed encoding structure. The rate allocation algorithm employed in the standard H.264 rate control scheme [118][119] at the frame layer is not changed.

However, at MB layer, the rate control scheme employed in the standard H.264 rate control scheme is changed based on the new proposed encoding structure. In the new proposed MB layer rate control scheme, the MAD prediction method based on FPE component is employed to predict MAD of MB and the rate allocation is performed according to the predicted MAD of each MB. Since complexity of each MB is determined by MAD, the video encoder knows complexity of each MB before the rate control at MB layer is performed. Therefore, it can use such information in order to achieve a more accurate rate control

performance.

Supposed $B_{i,j}$ is the number of bits for the current MB i in the j th frame. $B_{i,j}$ can be derived from the predicted MAD as

$$B_{i,j} = \frac{(B_j - h_a(N - i)) \times \bar{m}_{i,j}^2}{\sum_{n=i}^N \bar{m}_{n,j}^2} \quad (5.14)$$

where B_j is the number of remaining bits for the non-coded MBs in frame j , N is the total number of MBs in the current frame, $\bar{m}_{i,j}$ is the predicted MAD of the MB i in the frame j derived from (5.10), h_a is the number of bits used for header, syntax, motion vectors and other administrating markers. After $B_{i,j}$ is obtained, a quadratic rate model is employed to derive the quantization step size, Q_i , of the current MB. The relationship between $B_{i,j}$ and Q_i , according to the quadratic rate model, is shown below

$$B_{i,j} = a_1 \frac{\bar{m}_{i,j}}{Q_i} + a_2 \frac{\bar{m}_{i,j}^2}{Q_i^2} \quad (5.15)$$

where a_1 and a_2 are the model parameters that are updated from previously encoded parameters by using the linear regression method [118].

Since the rate allocation algorithm of our proposed rate control scheme is the same as that employed in the standard H.264 rate control scheme at frame layer and GOP layer, the target number of bits for the current frame can be derived by using the same method as that used in the standard H.264. When the target number of bits, B_j , for the current frame j is decided, the following steps are performed to encode the current frame:

Step A: Set $i = 1$ and define N as the total number of MBs in a frame, e.g., for a QCIF video sequence, $N = 99$. Define $h_{a,1}$ as a temporal variable to store the number of header bits. If the current frame is the first P-frame, $h_a = h_{a,1} = 10$, otherwise, set $h_{a,1}$ equal to

the value of h_a .

Step B: Perform RDO and ME/MC for the current MB by using two PMs, i.e., Inter16X16 or Skip, which is the function of FPE component. Predict the MAD of each MB in the MAD calculation component. The detailed procedure has been addressed in the above section. After that, the encoder stores the chosen PM p_i , the minimum distortion D_{e1} , and the corresponding MAD value M_i .

Step C: Compute the target number of bits for the current MB, $R_{i,j}$, by using (5.14). Note that h_a will be updated MB by MB.

Step D: Compute the quantization step size of the current MB, Q_i , by using the quadratic rate model (5.15). Refine the quantization parameter so as to reduce the blocking artifacts by $QP = \max\{QP' + 2, \min\{QP' - 2, QP\}\}$, where QP' is the quantization parameter of the previously encoded MB.

Step E: Perform RDO using the above QP value and the rest 5 PMs, which results in a new source distortion D_{e2} . The obtained source distortion D_{e2} is compared with D_{e1} obtained in the first step to decide the optimal p_i which leads to minimum source distortion. The actual encoding process is then performed based on optimal p_i .

Step F: Update the quadratic model parameters a_1 and a_2 using the linear regression method [119]. Compute the average number of overhead bits for syntax and motion vectors by using a recursive form:

$$\tilde{h}_a = \tilde{h}_a(i-1)/i + \hat{h}_a/i$$

$$h_a = \tilde{h}_a(i/N) + h_{a,1}(N-i)/N$$

where $\hat{h}_a$ is the actual number of bits used for syntax and motion vectors.

Step G: If $i < N$, let $i = i + 1$, and go to step 2. Otherwise, stop as all MBs have been encoded.

5.5 Simulation Results

The proposed rate control algorithm based on FPE component is applied to the JM6 [1] video encoder, which is a standard H.264 encoder. The simulation configuration for the encoder is given in table 5.3.

MV resolution	1/4
Hadamard	on
RD optimization	on
Search Range	32
Restrict Search Range	2
Reference Frames	1
Symbol Mode	CABAC
GOP structure	I P P Intra period= 0
Frames to be encoded	300

Table 5.3: Configuration of the H.264 encoder. The number of reference frame for ME/MC is 1, and the search range in the reference frame is 32. The first frame is intra-coding, and the rest frames are inter-coding.

In our simulation, color QCIF sequences are employed to test the performance of the proposed rate control scheme, as well as, the accuracy of the predicted MAD. The linear MAD prediction method and rate control scheme proposed in [119] are used as a comparison anchor. In the following, we call our MAD prediction method based on FPE component FPE method and the method proposed in [119] CBLP method. The rate control scheme based on FPE component is called FPER, and the rate control scheme based on CBLP method is named CBLPR.

5.5.1 Performance of MAD prediction

In Figure 5.11, the predicted MAD values obtained by using FPE method and CBLP method with RDO are compared with the actual MAD values. The test sequence is *Foreman*, which is encoded at a bit rate of 256 kbps and frame rate of 30 fps. It is obvious that the FPE method provides a better prediction accuracy for MAD as compared with

Figure 5.11: Comparison between the actual MAD values and the predicted MAD values obtained by using the FPE method and CBLP method, *Foreman*, frame index =4, bit rate= 256kbps, RDO is used.

the CBLP method. The reason is that the CBLP method greatly relies on the similarity of MBs between two successive frames, and mismatch occurs when the co-located MB in the previous frame is different from the MB in the current frame.

Similar experiments without RDO are also conducted, and results are shown in Figure 5.12. It can be observed that the FPE method substantially outperforms the CBLP method.

5.5.2 Performance of rate control scheme

In order to test the performance of the proposed rate control scheme, a lot of experiments have been performed. It is assumed that the first frame is I-frame with $QP = 38$, and the rest frames are P-frames. For fair comparison, the same bits allocation scheme is used in both FPER and CBLPR at GOP layer and frame layer. However, at MB layer, FPER allocates the number of bits to each MB according to the predicted MAD obtained by our FPE method, and CBLPR uniformly allocates the number of bits to each MB [119]. In

Figure 5.12: Comparison between the actual MAD values and the predicted MAD values obtained by using the FPE method and CBLP method, *News*, frame index =3, bit rate=56kbps, RDO is not used.

table 5.4, the name of the test video sequence, frame rates and target bit rates for different test schemes are described. At least 100 successive frames are encoded for each test sequence.

Test Name	Video sequence name	Frame rate F fps	Target bit rate (kbps)	RDO
fmn100	foreman	30	100	on
fmn80	foreman	20	80	off
sil48	silent	10	48	on
news64	news	10	64	off
md100	mother daughter	30	100	off

Table 5.4: Summary of the test scheme names, video sequence names, frame rate, target bit rate, and RDO feature.

Figure 5.13 shows the PSNR results obtained by using FPER and CBLPR, respectively, for the test scheme fmn100. The comparison result shows that FPER outperforms CBLPR slightly (0.2dB) in term of PSNR, and leads to a better picture quality.

Figure 5.13: PSNR comparison between FPER and CBLPR. The video test sequence is *Foreman*, frame rate = 30f/s, bit rate = 100kbps, RDO is on.

In table 5.5, the bit rate achieved by using FPER and CBLPR [118] are compared with the target bit rate for different test schemes. It can be observed in table 5.5 that in general, the

Test scheme name	CBLPR Achieved rate(kbps)	FPER Achieved rate(kbps)	Target bit rate(kbps)
fm100	100.1	100.01	100
fmn80	80.08	80	80
sil48	48.06	48.02	48
news64	64.1	64.02	64
md100	100.05	100.01	100

Table 5.5: Comparison of the achieved bit rate between the proposed FPER rate control and CBLP-based rate control. A number of video test sequence is used.

FPER achieves a bit rate closer to the target bit rate, while the CBLP based rate control leads to a slightly higher bit rate. The reason is that the FPE method provides a more accurate MAD of each MB, which represents the complexity of the MB, and therefore the FPER based on the FPE method achieves a better bit rate estimation.

The performance comparison between the FPER and CBLPR in terms of PSNR for different test schemes are listed in table 5.6. The results show that our rate control scheme

Test scheme name	CBLPR Achieved PSNR(dB)	FPER Achieved PSNR(dB)
fmn100	38.37	38.49
fmn80	38.75	38.85
sil48	38.69	38.91
news64	41.24	41.99
md100	41.64	41.85

Table 5.6: PSNR comparison between the proposed rate control and CBLPR.

leads to a 0.2-0.5dB improvement in term of average PSNR as compared with CBLPR. The better performance of FPER is because it allocates the number of bits to each MB according to the complexity of the MB, which leads to a slightly improvement in PSNR.

The marginal improvement shown in Fig 5.13 and Table 5.6 is because the rate control algorithm depends on not only MAD but also a lot of additional coefficients such as bits allocation scheme. In the rate control scheme of this Chapter, MAD has less impact on improving PSNR. However, accurate MAD is special important for those new emerging rate control techniques take into account frame complexity during bits allocation. The more accurate MAD estimation means more accurate estimation of frame complexity, so that the bits allocation would be more fair. The complex frames desire more bits in order to maintain high picture quality. From Fig. 5.11 and 5.12, we can tell that FPE has a better performance than CBLP. The proposed scheme would be important to those emerging rate control techniques.

5.6 Conclusion

Most of the current existing rate control schemes cannot be applied directly to the new emerging H.264 video encoder because of the technical challenge imposed by the encoder. In this chapter, a new encoding structure, which introduces a FPE component into MAD prediction method and rate control scheme for H.264, is proposed. The FPE component performs RDO and ME/MC by using two PMs in order to obtain a rough MAD value for the current MB, which is further used by the MAD calculation component to predict the actual MAD value of the current MB. Experimental results show that the MAD prediction method based on FPE component can achieve a more accurate MAD value than the existing linear MAD prediction method, which is the component of the standard H.264 rate control scheme. Furthermore, Based on FPE component, the H.264 standard rate control scheme is improved at MB layer. The new proposed rate control scheme at MB layer allocates the number of bits to each MB according to the MAD of the MB, so that it achieves a better control on the bit rate than the standard H.264 rate control. Experimental results show that the new rate control scheme achieves an average bits saving of 0.1kbps as compared with the standard H.264 rate control scheme. However, the average PSNR obtained by using the proposed rate control scheme is about 0.5dB higher than that obtained by using the standard H.264 rate control scheme.

The marginal improvement of PSNR is because the rate control algorithm depends on not only MAD but also a lot of additional coefficients such as bits allocation scheme. In the rate control scheme of this Chapter, MAD has less impact on improving PSNR. However, accurate MAD is special important for those new emerging rate control techniques take into account frame complexity during bits allocation. The more accurate MAD estimation means more accurate estimation of frame complexity, so that the bits allocation would be more fair. The complex frames desire more bits in order to maintain high picture quality.

Chapter 6

Adaptive Rate Control for H.264 Based on Logarithmic Rate Model

6.1 Introduction

In previous chapter, a new MAD prediction model for H.264 video encoder is studied, which leads to an improved rate control algorithm. In this chapter, we propose a new rate control scheme for H.264 video encoder. The new scheme takes into account not only the characteristics of H.264 video encoder but also the new logarithmic rate model, which has been studied in chapter 4. A number of new techniques have been adopted in the new video encoder, therefore most of existing rate control algorithms cannot be applied. The reason has been addressed in the chapter 4. Several techniques have been proposed to achieve a rate control algorithm for H.264 video encoder [117][118]. Ma *et al.* [117] propose a rate control algorithm based on TM5 [121], in which the previously encoded MB's quantization parameter is used to perform RDO for the current MB. However, RDO often has to be done two times or even more as the quantization parameter used for RDO is not optimal for the current MB. The iterative encoding process introduces a lot of additional calculations and computational complexity to the system. An adaptive rate control algorithm using

a fluid traffic model was proposed by Li *et al.* in [118]. The current MB's quantization parameter is derived from the linear MAD prediction model together with a quadratic rate model. Lately, based on [117] and [118], Li and Ma work together to create a rate control algorithm [119], which has been accepted by JVT as the standard rate control for H.264 video encoder. [119] is the same as [118] in many ways because they use the same fluid traffic model, the linear MAD prediction model and quadratic rate model. Recently, based on [119], M. Jiang *et al.* [122] propose an improved frame level rate allocation scheme, in which the number of bits is allocated to all non-coded frames unequally according to frame complexities. The basic idea behind [122] is to save bits from those frames with relatively less complexity and allocate more bits to frames with higher complexity due to high motion or scene changes.

In this chapter, we study a performance improved rate control algorithm for H.264 video encoder. The new rate control algorithm is based on the new proposed logarithmic rate model and novel MAD estimation technique. The other contribution of this chapter is that a new method is proposed to obtain the quantization parameter of the I-frame. The quantization parameter of the I-frame is now computed incorporating measurement of the buffer size, thus the rate control algorithm prevents buffer from underflow and overflow from the start.

The organization of this chapter is as follows. The next section reviews the standard rate control algorithm for H.264 video encoder. In Section 6.3, the new rate control algorithm is studied in great detail. Section 6.4 provides computer simulation results, and Section 6.5 concludes this chapter.

6.2 Review of standard rate control of H.264 system

The H.264 video encoder has been discussed in the Section 5.3. Some important techniques are also discussed in the previous Chapter, e.g., RDO is discussed in the Section 5.3 and 5.4. This section briefly reviews the adaptive rate control which has just been accepted by Joint Video Team (JVT) as an informative part of H.264 standard (Thereafter, we refer to it as JM6R since the rate control is implemented within JM6.1 [1], which is a H.264 video encoder).

The rate control is performed at three layers: GOP layer, frame layer and unit layer [119]. JM6R algorithm divides a frame into a group of contiguous MBs, which is called basic unit, to implement the rate control [119]. The basic unit consists of one or more MBs, and the choice of the number of MBs depends on the user.

1. GOP layer;
 - a. computation of total number of bits for current GOP.
 - b. computation of starting quantization parameter of current GOP.
2. frame layer;
 - a. pre-encoding stage: computation of number of bits for each frame.
 - b. post-encoding stage: update quadratic rate model parameters, and decide the number of frames that needs to be skipped.
3. unit layer;
 - a. computation of the number of bits for current encoding unit.
 - b. prediction of MAD of current unit.
 - c. computation of quantization parameters of current unit.
 - d. encoding of current unit.

- e. update the number of remaining bits and the number of non-coded units for current frame.

To predict the MAD of current MB, a linear model is introduced by using MAD in the co-located position of the previously encoded frame. The linear prediction model is given by

$$MAD_c = a_1 \times MAD_p + a_2 \quad (6.1)$$

where MAD_c is predicted MAD of current MB and MAD_p is the actual MAD of co-located position in previous frame, a_1 and a_2 are compensating factors, their initial values are 1 and 0, respectively, and they are updated after encoding each frame. The rate control algorithm employs a quadratic rate model to compute the quantization parameter of each unit [100][123].

6.3 Performance improved rate control for H.264

In this section, the improved performance of the rate control algorithm for H.264 video encoder is discussed. Firstly, this section presents quantization parameter derivation based on logarithmic rate model, which is proposed in the chapter 4. A technique to compute the quantization parameter of starting frame is given. The rate control algorithm is presented in the end of this section.

6.3.1 Quantization parameter derivation based on logarithmic rate model

In order to verify the relationship between the number of bits and quantization parameter, QP , a number of experiments have been performed and the results are described in Figure

6.1. The test sequence *Foreman* is encoded at a frame rate $30f/s$ by the H.264 video encoder. Several constant QPs ($QPs = 1, 12, 30, 36, 40, 46$) are used. The RDO is used when $QPs = 1, 12, 36, 46$. These experiments show that at a low bit rate the curve of the number of bits follows a convex-cup form. However, as QP decreases (bit rate increases), the curve of the number of bits shows a different form from convex-cup form. Therefore, it is better to use different curve functions to represent the number of bits for different QP range.

This is the reason that JM6R places a lot of restrictions on the QP which is derived directly from quadratic rate model, i.e., bounding QPs . In this way, even the QP derived from quadratic rate model is not accurate, it is bounded to a more appropriate QP . For example, we encode a test sequence *Foreman* at a target bit rate 96kbps and the frame rate is $15f/s$. The JM6R rate control is employed by the video encoder to compute QPs for MBs. The QPs before bounding and the QPs after bounding are compared to show how important those restrictions in JM6R are. Figure 6.2 gives our experimental results which including QP values before and after bounding. It is observed that the value of around 40% QPs are not really obtained by using the quadratic rate model. Instead, the actual values of QPs are obtained by different bounding. It is better to avoid such a situation, in which a lot of QP values are derived from kinds of bounding rather than by using actual rate model.

Now the new logarithmic rate model is used for quantization parameter derivation. Because the number of luminance and chrominance bits has a large deviation for different QP ranges, the logarithmic rate model can fit into rate curve better, and each logarithmic function represents a segment of actual rate curve. The new rate model is rewritten as

Figure 6.1: Verification of function form of the encoding rate (in bits/pixel) as a function of MAD and QP . The results are obtained by encoding 100 frames of the video sequence *Foreman* at $30f/s$, using JVT's H.264 encoder JM6.1 [1]. After encoding a MB, we measure MAD of MB, chosen prediction mode and the number of bits required by luminance and chrominance. The curves plot the average rate (in bits/pixel) for each MAD.

Figure 6.2: QPs before bounding and QPs after bounding vs. MB index for the test sequence *Foreman*. The frame rate is $15f/s$ and the target bit rate is 96kbps.

follows

$$B_{l,c} = \begin{cases} e^{\alpha(1)(\frac{\bar{\sigma}}{Q})^{\beta(1)}}, & 1 \leq QP < QP_L \\ e^{\alpha(2)(\frac{\bar{\sigma}}{Q})^{\beta(2)}}, & QP_L \leq QP < QP_H \\ e^{\alpha(3)(\frac{\bar{\sigma}}{Q})^{\beta(3)}}, & QP_H \leq QP \leq 51 \end{cases} \quad (6.2)$$

where $B_{l,c}$ is the number of bits of luminance and chrominance, $\alpha(1)$, $\alpha(2)$, $\alpha(3)$, $\beta(1)$, $\beta(2)$, and $\beta(3)$ are coefficients, which are updated after encoding a frame, $\bar{\sigma}$ is MAD of the MB. QP_L and QP_H denotes the boundaries of logarithmic functions.

For practical situation, the number of bits allocated to a MB is well known by the video

encoder. It is therefore to compute QP from (6.2) by the following

$$Q = \begin{cases} \frac{\sigma}{(\frac{B_{l,c}}{e^{\alpha(1)}})^{\beta(1)}}, & \text{if } B_{l,c} > B_L \\ \frac{\sigma}{(\frac{B_{l,c}}{e^{\alpha(2)}})^{\beta(2)}}, & \text{else if } B_{l,c} > B_H \\ \frac{\sigma}{(\frac{B_{l,c}}{e^{\alpha(3)}})^{\beta(3)}}, & \text{Otherwise} \end{cases} \quad (6.3)$$

where Q is the quantizer step size, $B_{l,c}$ is the number of bits allocated to the luminance and chrominance, which is well known by the video encoder, σ is the predicated MAD value of the encoding MB and the estimation process has been discussed in detail in chapter 5, B_L and B_H are boundary. In this chapter, we set $B_L = 150, B_H = 60$. The values of B_L and B_H are made empirically and users can select B_L and B_H by themselves.

In contrast to $Q = 2 \times QP$ within H.263, the relationship between Q and QP for H.264 is quite complex. The details are given in Table 6.1. From Table 6.1, it is therefore easy to

QP	0	1	2	3	4	5	6	7	8
Q	0.625	0.6875	0.8125	0.875	1	1.125	1.25	1.375	1.625
QP	9	10	11	12	...	18	...	24	...
Q	1.75	2	2.25	2.5		5		10	
QP	30	...	36	...	42	...	48	...	51
Q	20		40		80		160		224

Table 6.1: Quantization parameter and quantizer step size in H.264 codec. QP is quantization parameter, and Q is corresponding quantizer step size. Quantization parameter in H.264 ranges from 1 to 51.

derive the quantizer step size of MB after knowing QP , or vice versa.

After encoding, the H.264 video encoder collects the actual number of bits for chrominance and luminance, the actual MAD of the MB and quantizer step size to update corresponding

coefficients of the model.

$$\beta(n) = \frac{N(n) \sum_{i=1}^{N(n)} \ln B_i \ln \frac{\bar{\sigma}_i}{Q_i} - \sum_{i=1}^{N(n)} \ln B_i \sum_{i=1}^{N(n)} \ln \frac{\bar{\sigma}_i}{Q_i}}{N(n) \sum_{i=1}^{N(n)} (\ln \frac{\bar{\sigma}_i}{Q_i})^2 - (\sum_{i=1}^{N(n)} \ln \frac{\bar{\sigma}_i}{Q_i})^2} \quad (6.4)$$

$$\alpha(n) = \frac{\sum_{i=1}^{N(n)} \ln B_i - \beta(n) \sum_{i=1}^{N(n)} \ln \frac{\bar{\sigma}_i}{Q_i}}{N(n)} \quad (6.5)$$

where $n = 1, 2, 3$, $N(n)$ is the total number of MBs observed in the n th range, B_i is the actual luminance and chrominance bits of the i th MB, Q_i is quantizer step size of the MB, $\bar{\sigma}_i$ is the actual MAD value of the MB.

6.3.2 MAD estimation with fast pre-encoding component

The fast pre-encoding component has been studied in detail in Chapter 5. The extensive experimental results show that the new MAD prediction method based on fast pre-encoding component achieves more accurate MAD estimation than the linear MAD prediction model employed in JM6R. The reason is that the scene content in a video sequence varies quickly, and MBs in the residual frame change drastically depending on their spatial complexity and temporal difference. The instant change in scene content could lead to very severe consequences on MAD estimation. Furthermore, PM of the current MB may be different from that of the co-located MB in the previous frame and the variation of PMs further makes MAD prediction unstable.

In the proposed MAD calculation, a mathematical model is built in order to predict an accurate MAD of the current MB based on the rough MAD value produced by using mode Inter16X16 or Skip. The predicted MAD of the i th MB is re-written as the following

$$\bar{m}_i = \begin{cases} \mu \times M_i + \nu, & \text{if } (M_i \geq (\gamma_n \times M_{ave})) \\ M_i, & \text{otherwise} \end{cases} \quad (6.6)$$

where M_i is the rough MAD value of the i th MB generated from FPE component. μ and ν are adjusting factors and they are updated by using linear regression method after encoding a frame. M_{ave} is given by

$$M_{ave} = (\sum_{i=1}^N M_i)/N \quad (6.7)$$

where N is total MBs of a frame. γ_n is calculated recursively by

$$\gamma_n = (1 - \alpha)\gamma_{n-1} + \alpha\zeta \quad (6.8)$$

where $\gamma_0 = 1$, $\alpha = 0.75$ is moving average parameter, and ζ is defined as follows

$$\zeta = \sum_{M_i > m_i} M_i/s/M_{ave} \quad (6.9)$$

where m_i is the actual MAD of the i th MB, s is the number of MBs which satisfy the constraint $M_i > m_i$.

6.3.3 Quantizer calculation of starting frame

The QP of starting frame in each GOP is predefined based on the available channel bandwidth and the GOP length in JM6R [119]. Nevertheless, the generated number of bits of the first I-frame may cause buffer overflow or underflow because JM6R does not take buffer level into account. If a small buffer size is available to the video encoder, the buffer may overflow. Otherwise, the buffer may underflow if a fast channel is provided. Figure 6.3 shows a typical result of JM6R rate control. The video sequence *Silent* is encoded at a target bit rate $R = 128\text{kps}$, a frame rate $F_r = 30\text{f/s}$. The buffer size B_s is given as R/F_r [70]. From experiment result, it is observed that the QP of the first frame is 30, which generates 20592 bits. The number of bits for the first frame is about five times bigger than the given buffer size, which causes buffer overflow.

Figure 6.3: The buffer level vs. frame index for test sequence *Silent*. JM6R rate control is employed. The frame rate is $30f/s$. The target bit rate is 128kbps. Frame skip is not allowed. The video buffer size B_s is $R/F_r = 128000/30 = 4266$.

A new method is therefore proposed to calculate starting quantization parameters of each GOP. It considers buffer size. Since the video encoder buffer size is already known by the encoder before actual encoding, the number of bits allocated to the first frame of a GOP is given as follows

$$f(n_{G,1}) = R/F_r + (\gamma \times B_s - B_c(G-1, N_{GOP})) \quad (6.10)$$

where $f(n_{G,1})$ is the number of bits allocated to the first frame in the i th GOP, R is channel transmission bit rate, F_r is target frame rate, B_s is the video encoder buffer size, $B_c(G-1, N_{GOP})$ is the actual buffer level after finishing $(G-1)$ GOP, the length of $(G-1)$ GOP is N_{GOP} , γ is a adjusting coefficient, which is set to 0.5. We further present a simple rate model for the first frame of the GOP since the first frame of the GOP is normally

Intra-encoded.

$$\log B_{total} = \begin{cases} \mu_1 \times QP + \mu_0, & \text{if target bits} > 10000 \\ \nu_1 \times QP + \nu_0, & \text{otherwise} \end{cases} \quad (6.11)$$

where μ_1 , μ_0 , ν_1 and ν_0 are adjusting factors, B_{total} is the number of total bits allocated to the first frame of the GOP, QP is quantization parameter of the frame. In this chapter, for simplicity, μ_1 and μ_0 are set to constant -0.07 and 12.2 , respectively. ν_1 and ν_0 are set to constant -0.128 and 13.8 , respectively.

Figure 6.4 shows the estimation results of the proposed rate model. In our experiments, the

Figure 6.4: The comparison between the estimated number of bits obtained from the proposed model and actual number of bits per frame. The video sequences are encoded with distinct $QP = \{1, 4, 7, 10, 13, 16, 19, 22, 25, 28, 31, 34, 37, 40, 43, 46, 49, 51\}$. The test sequences are *Foreman*, *Silent*, *Container* and *Mother and Daughter*. Their frame rate are $30f/s$.

video sequences are encoded at a frame rate $30f/s$ by distinct $QP = \{1, 4, 7, 10, 13, 16, 19, 22, 25, 28, 31, 34, 37, 40, 43, 46, 49, 51\}$. 20 random frames are selected for each test

sequence and the Intra-encoded results are averaged. The video sequences are *Foreman*, *Silent*, *Container* and *Mother and Daughter*. It is clear that the proposed rate model can provide a good estimation to the number of bits per frame. The model accuracy can further be improved if a proper method is used to update corresponding model coefficients during encoding. The QP of the starting frame of each GOP is thus given by

$$QP_{i,1} = \begin{cases} \min(51, \max(\frac{\max(0, (\log f(n_{G,1}) - \mu_0))}{\mu_1}, 1)), & \text{if } f(n_{G,1}) > 10000 \\ \min(51, \max(\frac{\max(0, (\log f(n_{G,1}) - \nu_0))}{\nu_1}, 1)), & \text{otherwise} \end{cases} \quad (6.12)$$

In the equation, $QP_{i,1}$ is quantization parameter of the starting frame of each GOP.

6.3.4 Estimation of number of header bits

As the H.264 video bitstream consists of not only the luminance and chrominance coefficients, but also MB header, MV and so on. The header bits means the bits for MB header, MV and delta QP. In JM6R, the number of bits for a basic unit is given by f_{rb}/N_{ub} , where f_{rb} denotes the number of remaining bits for the non-coded basic unit and N_{ub} denotes the number of non-coded basic unit [119]. The number of bits for the luminance and chrominance are given by

$$B_{l,c} = f_{rb}/N_{ub} - H \quad (6.13)$$

where $B_{l,c}$ is the number of bits for luminance and chrominance coefficients. H is the number of header bits, i.e., bits for MB header, delta QP, MVs and so on.

In JM6R, the number of header bits is predicted by a recursive method. However, the recursive method in [119] cannot provide a good estimation of the number of header bits because the number of header bits especially MVs is bursty. One of typical examples is shown in Figure 6.5. It is observed that the number of bits for MVs per MB varies greatly

depending on chosen PM. The video sequence *Foreman* is encoded at a frame rate $15f/s$

Figure 6.5: The number of bits for MVs per MB vs. prediction mode. The test sequence *Foreman* is encoded by the H.264 encoder with constant $QPs = 10, 20, 35$. The frame rate is $15f/s$.

by the H.264 video encoder with three different $QP = 10, 20, 35$. It is obvious that the MB encoded with PM Inter8X8 requires 2-6 times number of bits to encode MVs as compared with the MBs encoded with PM Inter16X16. In some cases, the MB encoded with PM Inter8X8 requires more than 20 times number of bits for MVs than the MB encoded with Inter16X16. For example, Figure 6.6 gives the number of bits per MB for a lot of successive frames. The results are generated from the video sequence *Silent* at a frame rate $30f/s$. It is observed that the video encoder may need at most 140 bits for MVs of a MB. Suppose a QCIF-video sequence is encoded at a target bit rate 128kbps and a frame rate $30f/s$, and the number of total bits allocated to a MB is computed by using a simple uniform

Figure 6.6: The number of bits required by MVs per MB vs. MB index. The test sequence *Silent* is encoded by the H.264 encoder with constant $QP = 10$. The frame rate is $30f/s$.

allocation technique.

$$\begin{aligned} TotalBitsPerMB &= \frac{1}{99} \times \frac{128 \times 10^3}{30} \\ &= 43 \end{aligned} \quad (6.14)$$

In the above equation, $TotalBitsPerMB$ denotes the number of total bits for a MB. There are total 99 MBs within a QCIF frame. It is obvious that, in this scenario, the MVs of a MB may use up all the bits allocated to the MB. Therefore, it is very important to determine the number of header bits.

Because motion prediction mode plays an important role in the number of bits for MVs, a model is proposed as follows

$$\pi_7 = \frac{K}{Q_{ave}} + S \quad (6.15)$$

where K and S are coefficients, Q_{ave} is the average quantizer step size of a frame, π_7 denotes how many MBs are predicted by model Inter8X8 in a frame (The detailed definition is in chapter 5, equation(5.9)) . The initial values of K and S are set to be 1 and 0, respectively.

In previous chapter, it is observed that three motion prediction modes, Inter8X8, Inter16X16 and Skip, are the most often used motion prediction modes. At the same time, in previous section, it is observed that the MBs encoded with PMs Inter16X8 Inter8X16, and Inter16X16 require similar number of bits to encode their MVs. Therefore, the number of header bits is given as

$$H = (\tau - \varsigma) \times H_{Inter8X8} + (1 - \tau + \varsigma) \times H_{Inter16X16} \quad (6.16)$$

where H is the number of header bits of those remaining uncoded MBs, $H_{Inter8X8}$ is the average number of header bits of MBs that are encoded with PM Inter8X8, $H_{Inter16X16}$ is the average number of header bits of MBs that are encoded with PM Inter16X16, Inter16X8 and Inter8X16, τ is obtained using equation (6.15), ς is given by

$$\varsigma = n_{MB8}/n_{MBF} \quad (6.17)$$

where n_{MB8} denotes how many MBs is encoded with PM Inter8X8 among those already encoded MBs in the frame, n_{MBF} is the number of total encoded MBs in the frame.

6.3.5 Adaptive rate control based on optimized components

In this section, we summarize the proposed performance improved rate control algorithm. The new rate control algorithm employs the above logarithmic rate model, the improved MAD prediction technique and so on.

The rate allocation and the computation of QP of the staring frame of a GOP have been

discussed in detail in Section 6.3.3. In the following, we firstly present an optimal frame level rate control algorithm for the rest frames in GOP. Then, the computation of corresponding quantization parameters at MB level is discussed.

Frame layer rate control

Before current frame is encoded, the number of bits in the encoder buffer, B_c , is updated and is checked to see whether the following frame needs to be skipped. The number of bits in the encoder buffer is

$$B_c(G, j+1) = \max(B_c(G, j) + B' - R/F_r, 0) \quad (6.18)$$

where B' is the actual number of bits used to encode those previous frames, $B_c(G, j)$ is the previous buffer level, R and F_r are the channel bit rate and frame rate, respectively, G denotes GOP number and j means frame number. The number of bits allocated to the $j+1$ frame in the G th GOP is given

$$f(n_{G,j+1}) = \frac{R}{F_r} - \Delta, \quad (6.19)$$

$$\text{where } \Delta = \begin{cases} \frac{B_c(G,j+1)}{F_r}, & B_c(G, j+1) > Z \times M \\ B_c(G, j+1) - Z \times M, & \text{otherwise} \end{cases}$$

where $f(n_{G,j+1})$ is the number of bits assigned to the frame j in the G th GOP, Z is target buffer delay, M is the threshold for frame skipping. By default, we set $M = R/F_r$ and $Z = 0.5$. The frame level rate control algorithm tries to maintain the buffer fullness at the 50% level. If the buffer fullness is greater than the 50% level of the maximum fullness M , the number of bits for the next frame will be decreased.

MB level rate control

The MB level rate control algorithm computes quantization parameters for all of MBs in a frame according to the proposed logarithmic rate model, so that the sum of the bits of the frame is close to frame target bits $f(n_{G,j})$. The steps are given as follows:

Step A: GOP Initialization

Let $N_t = N_{MBF} = 99$ and $QP_{prev} = QP_0$, where QP_0 is quantization parameter of the starting frame in this GOP. If this is the first GOP, set $K = 1$, $S = 0$, $H_{Inter8x8} = 20$ and $H_{Inter16x16} = 10$. Otherwise, set K , S , $H_{Inter8x8}$ and $H_{Inter16x16}$ to the values obtained after encoding the previous GOP.

Step B: Pre-motion estimation/compensation

Perform motion estimation and compensation for the frame. Compute MAD of MBs using the method described in the previous chapter. Let $\sigma_{all} = \sum_{s=0}^{N_t} \sigma_s^2$, where σ_s is the MAD of the s th MB.

Step C: Frame Initialization

Set $N = N_t$, $n_{MB8} = n_{MBF} = 0$ and $i = 0$, where i is the MB number which has been encoded. The number of bits for the frame is $f(n_{G,j})$. Set $H_{s1} = 0$, $H_{s2} = 0$, $s1 = 0$ and $s2 = 0$.

Step D: Allocate number of bits to the luminance and chrominance coefficients

Compute τ using equation (6.15). The average quantizer step size of previous coded MBs in the frame is used. The average number of bits for header, H , is then predicted through (6.16). The number of bits allocated to the luminance and chrominance coefficients is equal to

$$B_{l,c} = \frac{(f(n_{G,j}) - N \times H) \times \sigma_i^2}{\sigma_{all}} \quad (6.20)$$

Step E: Compute quantization parameter

If $B_{l,c} > 0$, compute the optimized quantizer step size using the following equation

$$Q_i = \begin{cases} \frac{\sigma_i}{(\frac{B_{l,c}}{e^{\alpha(1)}})^{\frac{1}{\beta(1)}}}, & \text{if } B_{l,c} \geq 150 \\ \frac{\sigma_i}{(\frac{B_{l,c}}{e^{\alpha(2)}})^{\frac{1}{\beta(2)}}}, & \text{else if } B_{l,c} \geq 60 \\ \frac{\sigma_i}{(\frac{B_{l,c}}{e^{\alpha(3)}})^{\frac{1}{\beta(3)}}}, & \text{else} \end{cases} \quad (6.21)$$

QP is then found by looking up Table 6.1 with computed Q_i . Otherwise, we are running out of bits, so we set $QP = QP_{prev} + 10$.

Step F: Compute appropriate QP to encode MB

Put a simple constraints on computed QP in order to avoid fast quality change.

$$QP = \max \{1, QP_{prev} - Dq, \min(QP, QP_{prev} + Dq, 51)\} \quad (6.22)$$

where Dq is given as follows

$$Dq = \begin{cases} 10, & \text{if } 1 \leq QP_{prev} < 21 \\ 3, & \text{else if } 21 \leq QP_{prev} < 31 \\ 1, & \text{else} \end{cases} \quad (6.23)$$

The MB is encoded with obtained QP .

Step G: Update model coefficients

Collect the number of generated texture bits B_i , the number of header bits H_g , the actual MAD $\bar{\sigma}_i$ and prediction mode (PM). Set $QP_{prev} = QP$, $f(n_{G,j}) = f(n_{G,j}) - (B_i + H_g)$ and $\sigma_{all} = \sigma_{all} - \sigma_i^2$. Update linear MAD prediction model, FPE prediction model and logarithmic rate model.

Update the average number of bits for header. If prediction mode is Inter8X8, then

$$H_{s1} = \frac{H_{s1} \times s1 + H_g}{s1 + 1} \quad (6.24)$$

$$s1 = s1 + 1 \quad (6.25)$$

$$H_{Inter8X8} = \frac{H_{s1}(i + 1) + H_{Inter8X8}(N_t - 1 - i)}{N_t} \quad (6.26)$$

If prediction mode is Inter16X16, Inter16X8 and Inter8X16, then

$$H_{s2} = \frac{H_{s2} \times s2 + H_g}{s2 + 1} \quad (6.27)$$

$$s2 = s2 + 1 \quad (6.28)$$

$$H_{Inter16X16} = \frac{H_{s2}(i + 1) + H_{Inter16X16}(N_t - 1 - i)}{N_t} \quad (6.29)$$

π_A is also updated according to prediction mode.

Step H: Continue

Set $i = i + 1$, $N = N - 1$, $n_{MB8} = s1$ and $n_{MBF} = i$. If $N > 0$, go to step 4. Otherwise, set $N_{MBF} = n_{MBF}$. Update K and S using linear regression method. Go to frame layer rate allocation.

In the proposed scheme, the rate control is done in two layers, i.e. frame layer and MB layer. In frame layer, the number of bits for each frame is computed using current buffer level, channel bandwidth and frame rate. At MB layer, the quantization step size is derived from logarithmic rate model. The iterative steps are performed at the video encoder to achieve a target bit rate at MB level. The complexity of the proposed rate control is reasonable because each MB is encoded only once and RDO is performed only once. Nevertheless, our technique is slightly more complex than JM6R rate control because the encoder needs do pre-encoding (ME/MC) and more model are employed in the proposed rate control algorithm.

The computational complexity of the proposed method arises from logarithm rate model, as MAD prediction methods adds no additional computation to existing video encoder. The complexity of logarithmic model has been discussed in the Chapter 4, and we know that it is still low as compared with ME/MC in the encoding process. Therefore, the complexity of the proposed rate control is low as compared with H.264 video encoding process.

6.4 Simulation results

To show the effectiveness of the proposed rate control algorithm, the rate control is implemented in the public H.264 video encoder, i.e., JM6.1. The configuration parameters of H.264 video encoder are the same as we have used in the previous chapter 5.3. For completeness, we re-state them as follows:

MV resolution	1/4
Hadamard	on
RD optimization	on/off
Search Range	32
Restrict Search Range	2
Reference Frames	1
Symbol Mode	CABAC
GOP structure	I P P Intra period= 0
Frames to be encoded	300

Table 6.2: The configuration parameters of H.264 video encoder. MV resolution is 1/4 means that motion vectors may have the non-integral values, which are derived from original pixel values using bilinear interpolation. The entropy coding is Context-Adaptive Arithmetic Coding (CABAC). The number of reference frame for ME/MC is 1, and the search range in the reference frame is 32. The first frame is intra-coding, and the rest frames are inter-coding.

The logarithmic rate model and rate control algorithm are added into the H.264 video encoder to compute quantization parameter of each MB and the number of frames that needs to be skipped. In the following, the proposed rate control algorithm is called logarithm

rate control (LR) as the rate control is based on logarithmic rate model. The video sequences include *Foreman*, *News*, *Silent*, *Mother and Daughter* in QCIF format (176×144 pixels/frame). The JM6R rate control is employed to provide a performance comparison anchor. It is assume that the first frame is Intra-coded and the rest of frames are Inter-coded.

In Table 6.4, the video sequence, target bit rates, frame rates, and names assigned to each of test are described. Those video sequences are all well known, and we encode 100 frames.

Test Name	Video Sequence	Frame rate F_r (fps)	Target Bit rate R (kbps)
fmn128	foreman	30	128
fmn96	foreman	10	96
news256	news	30	256
sil100	silent	30	100
sil60	silent	15	60

Table 6.3: Description of the experimental configuration: Names, video sequence, frame rate F_r and target bit rate R . Each name corresponds to a given video sequence, frame rate and target bit rate.

6.4.1 Results for stored video

At first, we test the performance of the proposed rate control for stored video. The stored video has less constraints on the video buffer size as the H.264 video bitstream can be stored into harddisk directly after compression. Therefore, the objective is, given a total bit budget, to achieve average PSNR as high as possible. The buffer level after encoding a frame is a less important factor and the encoder cares more about the final bit budget and average PSNR. Figure 6.7 shows comparison of PSNR when using the proposed LR algorithm and JM6R. In the experiment, it is observed that both rate control algorithms do not skip frame and all the frames in the test sequence are encoded. It is clear that LR rate control provides a lower PSNR for the first frame as compared with JM6R. The reason

Figure 6.7: PSNR Comparison of JM6R and the proposed LR method. The video sequence is *Foreman*, total bit budget is $R * 100 / F_r$, where $R = 128kbps$ and $F_r = 30f/s$.

is that, in the proposed LR rate control algorithm, QP of the starting frame in GOP is derived from a measurement of buffer level and a given channel transmission bit rate and the resulted QP is smaller. However, for the next several frames, LR outperforms than JM6R in PSNR. The average PSNR of the proposed LR algorithm and JM6R are 35.56dB and 35.81dB, respectively. The results show that the proposed LR algorithm achieves a comparable performance as compared with JM6R to meet a given total bit budget. The number of bits per frame and buffer level after encoding each frame are compared in Figure 6.8-6.9. In Figure 6.8, it can be observed that JM6R requires much more bits to encode the first frame than LR. Therefore JM6R needs a very large buffer to store coded frame. From Figure 6.8, it is easy to understand that why there are maybe overflow in the buffer. In Figure 6.9, the buffer level which is less than 0 means that buffer underflows (However, the buffer level can not be less than 0 in practical situation, so in the following we use 0 to denote buffer underflow). It is observed that if there is a finite buffer restriction for the video encoder, JM6R would suffer from severe underflow and overflow. The reason is that

Figure 6.8: The number of bits per frame of JM6R and the proposed LR algorithm. The video sequence is *Foreman*, total bit budget is $R * 100 / F_r$, where $R = 128kbps$ and $F_r = 30f/s$.

Figure 6.9: Buffer level after encoding each frame of JM6R and the proposed LR algorithm. The video sequence is *Foreman*, total bit budget is $R * 100 / F_r$, where $R = 128kbps$ and $F_r = 30f/s$.

JM6R consumes too many bits on the first frame, which can be observed in Figure 6.8. Rather, the proposed LR rate control maintains a rather good buffer level at a cost of low PSNR for the starting frame.

Although the performance (PSNR) of the proposed rate control algorithm is similar to that of JM6R, LR has more models than JM6R and thus more model coefficients needs to be updated during encoding, it is therefore a little more complex and time-consuming than JM6R.

6.4.2 Results for low delay communication

However, in practical situation, there are always some buffer constraints for the video encoder. In the following, we compare the performances of the proposed LR rate control and the standard rate control of H.264 video encoder when the video encoder has buffer constraints. For a given transmission channel bandwidth, the smaller buffer size will lead to a lower delay. If a limited buffer size M is defined and accumulated number of bits is greater than M , the video buffer overflows and the next several frames needs to be skipped. On the other hand, if accumulated number of bits is less than the bit rate the channel provides, the buffer underflows. The objective of rate control is to avoid both overflow and underflow while maximizing average PSNR of the sequence. In the following experiments, it is assumed that the buffer size at the video encoder is $M = R/F_r$ for all video test sequences, because too large buffer size causes a large delay at the decoder, which is not desirable.

Figure 6.10-6.12 shows plots of the number of bits per frame, the number of bits in the encoder buffer after coding each frame and the resulted PSNR when using LR rate control and JM6R rate control. The video test sequence *Foreman* is encoded at a target bit rate $R = 128kbps$. The frame rate F_r is 30f/s and the buffer size at the video encoder

Figure 6.10: The number of bits per frame comparison of JM6R and LR algorithm. The video sequence is *Foreman*, and target bit rate $R = 128kbps$, frame rate $F_r = 30f/s$, $M = R/F_r$.

is $M = R/F = 4266bits$. If there is more than $0.8M$ bits in the video buffer, both rate control schemes skip frames until the buffer level is below $0.8M$. It is decided that when a frame is skipped, the previously encoded frame should be used to compute PSNR because the decoder would displays the previous frame until the next encoded frame comes to the decoder. It can be know from Figure 6.10 that the QP calculation of the first frame works effectively, so that video encoder generate a moderate number of bits. The buffer level results in Figure 6.11 shows that LR rate control avoids underflow and overflow successfully. It also can observe that LR schemes leads to a smaller buffer fluctuation than JM6R. When JM6R is used by the video encoder, the video encoder suffers from buffer underflow 2 times and skips total 12 frames, 6 of which is skipped very near the first frame. It means that QP which is obtained by using JM6R enables the video encoder to produce too many bits to be stored at the video buffer, and the JM6R rate control for the latter frames is also not good enough as compared with LR algorithm. The proposed LR rate control algorithm

Figure 6.11: Buffer level comparison of JM6R and LR algorithm. The buffer level is measured after encoding each frame. The video sequence is *Foreman*, and target bit rate $R = 128kbps$, frame rate $F_r = 30f/s$, $M = R/F_r$.

Figure 6.12: PSNR comparison of JM6R and LR algorithm. The video sequence is *Foreman*, and target bit rate $R = 128kbps$, frame rate $F_r = 30f/s$, $M = R/F_r$.

does not skip any video frames and it therefore outperforms JM6R. The PSNR results in 6.12 show that the proposed LR rate control provides a rather constant picture quality and a higher average PSNR than JM6R. JM6R has a higher starting PSNR, and the picture quality degrades slowly.

Figure 6.13-6.15 provides additional plots of the number of bits per frame, the number of bits in the buffer after encoding each frame and the resulted PSNR when using our rate control and JM6R rate control. The sequence *Foreman* now is encoded at a target bit rate

Figure 6.13: The number of bits per frame comparison of JM6R and the proposed LR algorithm. The video sequence is *Foreman*, bit rate $R = 96kbps$, frame rate $F_r = 10f/s$, $M = R/F_r$.

$R = 96kbps$, frame rate $F_r = 10f/s$. The available encoder buffer size is $M = 9600bits$. It is obvious that the encoder with JM6R rate control skips some frames and the buffer underflows. The proposed LR algorithm outperforms than JM6R in term of average PSNR and reduces number of skipped frames.

Figure 6.14: Buffer level comparison of JM6R and the proposed LR algorithm. The buffer level is measured after encoding each frame. The video sequence is *Foreman*, bit rate $R = 96kbps$, frame rate $F_r = 10f/s$, $M = R/F_r$.

Figure 6.15: PSNR comparison of JM6R and the proposed LR algorithm. The video sequence is *Foreman*, bit rate $R = 96kbps$, frame rate $F_r = 10f/s$, $M = R/F_r$.

In Table 6.4, we show a summary of achieved bit rate, the number of skipped frames, and the average PSNR when using our rate control algorithm and JM6R rate control. 300

Test Name	JM6R bit rate(kbps)	LR bit rate(kbps)	JM6R PSNR(dB)	LR PSNR(dB)	JM6R skipped	LR skipped
fmn128	128.5	128.1	34.68	35.56	12	0
fmn96	96.3	96	35.42	36.58	6	0
news256	257.1	256	42.81	43.39	7	0
sil100	100.6	100.1	35.55	36.64	15	0
sil60	60.1	60.1	35.86	36.47	10	0

Table 6.4: Comparison of achieved bit rate and average PSNR for JM6R rate control and the proposed LR.

frames of each video test sequence are encoded by the H.264 video encoder. The proposed LR algorithm and JM6R are employed to control quantization parameter of the video encoder, respectively. The PSNR of each frame is collected to compute average PSNR of the sequence, which is interpreted as a measure of overall picture quality. The results show that LR provides a 1-2dB higher average PSNR as compared with JM6R. At the same time, the number of skipped frames is greatly reduced when the proposed LR rate control is employed by the video encoder. In general, the new proposed rate control achieves a bit rate closer to the target bit rate, while the JM6R needs slightly more bits.

6.5 Conclusions

In this chapter, the performance improved rate control algorithm for the new emerging H.264 video encoder is studied. Because new techniques, such as various motion prediction modes and RDO, are applied to the H.264 video encoder, most of current existing rate control algorithms cannot be used directly. The proposal of the standard rate control algorithm for the H.264 video encoder represents a major advance in the area of rate control. However, the standard rate control for H.264 video encoder cannot maintain a good performance when the encoder has a finite buffer size. The rate control for low delay H.264

communication is studied in this chapter since low delay impose a more tight constraint on the buffer size, and therefore requires a more accurate rate control algorithm.

The logarithmic rate model, which has been studied in the chapter 4, is applied into the new proposed rate control algorithm to derive quantization parameter. Furthermore, the improved MAD prediction technique enables the proposed rate control algorithm obtain the characteristics of the video frame more accurate and faster. The computation of the starting quantization of each GOP is based on the buffer level and channel bandwidth, so the proposed rate control algorithm avoids spending too many or few bits on the very first frame of the GOP.

In comparison to standard rate control of H.264, the new proposed rate control algorithm achieves a better computation on the quantization parameter, and therefore smooths buffer level effectively as well as avoids underflow and overflow. Experimental results show that the proposed rate control achieves an average 1-2dB PSNR improvement as compared with standard rate control for the H.264. The number of skipped frames is reduced greatly because of rather constant buffer level as compared with standard rate control of H.264. It is observed that sometimes the standard rate control skips 15 frames and the proposed rate control skips 0 frame. The proposed rate control algorithm is therefore suitable for low delay H.264 video communication.

Chapter 7

Conclusions and Recommendations

7.1 Conclusions

Media rich services over the Internet and wireless networks are expected to be the dominant applications in the near future. However, the high error rates and limited bandwidth constraints in wireless networks still hamper ubiquitous media-rich applications. Therefore, providing robust compressed video transmissions as well as efficient multimedia rate control is the key to enable these networked and wireless media rich services. In this thesis, we have investigated several rate control techniques for video streaming.

We have proposed a novel joint source channel rate allocation framework, where the compressed video bitstream is protected in such a way that the end-to-end distortion is minimized. The key contribution is that, since PSNR of a video frame depends on BER in that frame, the optimal rate allocation method should optimize BER in the decoded frame. An decoded BER based adaptive joint source channel rate control method is therefore proposed. The error protection scheme targets for a specified decoded BER, and the optimal rate allocation is derived after deriving FEC code rates. The concept is straightforward and easy to be implemented. Furthermore, UEP protection technique is employed to maximize

utilization efficiency of transmission bandwidth. The experimental results show that the quality of the video frames can be improved by 1-3 dB as compared to fixed UEP-based rate control approach. The proposed rate control method is also compared with an operational rate distortion based rate allocation (BM) scheme. The experimental results show that the proposed method is comparable with BM scheme. Nonetheless, the complexity of the proposed scheme is much lower as compared with BM scheme. In BM scheme, in order to obtain universal rate distortion characteristics, each frame needs to be encoded in a number of times, and the compressed bitstream is protected by a number of FEC schemes. On the other hand, the proposed scheme does not need to encode a frame iteratively and channel encoded video bitstream iteratively.

If an accurate rate model is used in a rate control algorithm, the derivation of encoding parameters can be simple and the derived parameters are effective. As the number of bits of a frame grows non-linearly as quantizer step size of the frame grows, a novel rate model using multiple logarithmic functions is proposed. The idea is to use different logarithmic functions to represent different parts of actual rate curve. In this way, the logarithmic rate model fits an actual rate curve in a combination method. The previously encoded frames are used to derive the logarithmic model parameters and optimize boundary, therefore, they are adaptive to statistics of previously encoded frames. The simulation results show that logarithmic rate model outperforms some existing rate models in term of bits number estimation accuracy. The accuracy can be improved about 80% while the computational complexity remains the same level.

The rate control scheme can derive a better and more accurate quantization parameter if it knows MAD or MSE clearly. Nonetheless, the new emerging H.264 video coding standard uses RDO and other techniques, which improves compression efficiency greatly yet sets an obstacle for the video encoder to obtain MAD and MSE . We have proposed a FPE com-

ponent to perform MAD and MSE computation for H.264 encoder. The idea is inspired by the facts that correlations exists between different prediction modes. The proposed MAD/MSE prediction method is implemented in standard H.264 video encoder. The proposed method is compared with an existing linear MAD prediction model. The computer simulation results show that FPE component-based rate control scheme demonstrates a PSNR improvement 0.1-0.5 dB as compared with linear MAD prediction model-based rate control scheme, while the achieved bit rate of the proposed scheme is closer to the target bit rate than linear MAD prediction model based rate control scheme.

Since there is no report on rate control for low-delay H.264 video communication system, we have proposed a rate control scheme to achieve it. Firstly, The previously studied logarithmic rate model is used to derive quantization parameter for each MB. Next, The FPE component is employed to obtain MAD of the encoding MB. In additional that, a novel method is proposed to compute the quantization parameter of the starting frame of each GOP, and it is based on measurement of the video encoder buffer level and the channel transmission rate. We also propose a new method to predict the number of headers bits which includes MB header and MVs. The novel method is based on prediction mode and the average number of headers bits which is derived from previously encoded frames. The novel rate control scheme is therefore designed specially for low delay H.264 communication. The experimental simulation results show that our rate control scheme reduces the number of skipped frames greatly as compared with standard H.264 rate control scheme. The results show that the proposed rate control scheme can effectively prevent buffer overflow and underflow and improve the quality of the decoded video frames by 1-2dB as compared to standard H.264 rate control scheme.

7.2 Recommendations for further research

There are a number of areas studied in this thesis where further work could be pursued.

1. Exploiting a model to compute optimal error probability in the decoded video packet

The decoded SER is employed to derive FEC code rates and control rate allocation between the video encoder and channel encoder. In this thesis, the target SER is derived using pilot slices, which requires an additional pilot channel. A better solution is to construct a model which can derive target SER based on several parameters such as data packet size, channel BER and so on. The model can use statistics of the channel and previously decoded video packets to update corresponding parameters. However, how to keep up with the change of the video frames and channel statistics, and how to mitigate distortions caused by incorrect estimation of the video frames or channel statistics are important issues that have to be addressed.

On the other hand, the symbol error rates between two priority-based video bitstream may exist some correlations as both the video bitstream are transmitted over the same physical channel. It is therefore interesting to investigate how the correlation will affect the overall distortion D_{s+c} . As far as the author knows, there is a few reports available to tackle the problem.

2. Joint source channel video coding for new emerging H.264 video coding standard

The new emerging H.264 video coding standard provides a more flexible network interface, so it is easier for it to adopt a joint source channel video coding scheme to improve bandwidth utilization efficiency. Now H.264 supports three sub-bitstreams with different priorities, and H.264 video encoder/decoder can process those sub-bitstreams directly. It is possible that more benefits could be gained from those standardized priority-based sub-bitstreams. But as more flexibility is provided in H.264 video encoder, it is more difficult to achieve a specified bit rate.

3. Improving efficiency and reducing complexity of rate control of H.264

The standard H.264 rate control scheme enables the video encoder provide a better picture quality than the encoder which does not use rate control scheme, while the bit rate remains the same. The proposed rate control scheme outperforms the standard rate control for low-delay H.264 video communication system. Nevertheless, a number of models are employed in the proposed rate control scheme, and their coefficients needs to be updated after coding each frame. It is possible to reduce computational complexity of the proposed rate control scheme by using simpler model.

4. Using subjective method to measure video picture quality

In this thesis, objective methods are used to measure how is the picture quality. However, subjective methods have gained importance for scalable video coding. It is possible that more benefits could be gained by adopting subjective methods, as subjective methods reflect the picture quality from user's point view.

Author's Publications

This chapter shows the list of publications arising from the research work reported in this thesis.

Journal Papers

1. J. Wei, B. H. Soong and Z. G. Li, "A New Rate-Distortion Model For Video Transmission Using Multiple Logarithmic Functions", IEEE Signal Processing Letters, pp. 694- 697, Vol. 11, Issue, 8, Aug, 2004.
2. J. Wei, Z. G. Li and B. H. Soong, "Unequal Error Protection and Adaptive Source Channel Rate Control for MPEG-4 Wireless Transmission" submitted to Image Communication.
3. J. Wei, B. H. Soong and Z. G. Li, "Rate Control in H.264 Video Coding for Low Delay Communications", submitted to IEEE Trans. Circuits Syst. Video Technology.

Conference Papers

1. J. Wei, Z. G. Li and B. H. Soong, "An Adaptive Video Coding System Over Wireless Channel," *2003 International Conference on Software, Telecommunications and Computer Networks (SoftCom)*, Split, Dubrovnik (Croatia) Ancona, Venice (Italy), October, 2003.

2. J. Wei and B. H. Soong, "Multiple States Transcoding Proxy for Wireless Video Streaming," *2003 International Conference on Software, Telecommunications and Computer Networks (SoftCom)*, Split, Dubrovnik (Croatia) Ancona, Venice (Italy), October, 2003.
3. J. Wei, B. H. Soong and Z. G. Li, "Rate Control for H.264 Video Transmission," *2003 Wireless Networking Symposium (WNCG)*, Austin, TX, USA, October, 2003.

Bibliography

- [1] JVT. Jvt test model version 6.1. [Online]. Available: ftp://standards.polycom.com/jvt-experts/2003_03_Pattaya/JVT-G012r1_software.zip
- [2] W. B. Pennebaker and J. L. Mitchell, *JPEG still image data compression standard*. New York: Kluwer Academic Publishers, 1993.
- [3] ISO/IEC 14496-2 JTC1/SC29/WG11, "Information technology: coding audio-visual objects," Pisa, Italy, Jan. 2001.
- [4] H. Holma and A. Toskala, Eds., *WCDMA for UMTS, 2nd Edition*. John Wiley and Sons, 2002.
- [5] ITU-T Rec. H.264/ISO/IEC 11496-10, "Advanced video coding, final committee draft," in *JVT-G050*, Mar. 2003.
- [6] B. W. Abeysundara and A. Kamal, "High-speed local area networks and their performance: a survey," *ACM Computing Surveys (CSUR)*, vol. 23, pp. 221–264, June 1991.
- [7] ATM forum. Atm beginner overview. [Online]. Available: <http://www.atmforum.com/aboutatm/guide.html>
- [8] Standard video test sequences. [Online]. Available: <http://ise.stanford.edu/video.html>

- [9] M. T. Sun and A. R. Reibman, Eds., *Compressed video over networks*. New York: Marcel Dekker, Inc, 2002.
- [10] C.-Y. Hsu, A. Ortega, and M. Khansari, "Rate control for robust video transmission over burst-error wireless channels," *IEEE J. Select. Areas Commun.*, vol. 17, pp. 756 – 773, May 1999.
- [11] Z. H. He, J. F. Cai, and C. W. Chen, "Joint source channel rate-distortion analysis for adaptive mode selection and rate control in wireless video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, pp. 511–523, June 2002.
- [12] F. Wu, S. P. Li, and Y. Q. Zhang, "A framework for efficient progressive fine granularity scalable video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, pp. 332 – 344, Mar. 2001.
- [13] S. Aramvith, I.-M. Pao, and M.-T. Sun, "A rate-control scheme for video transport over wireless channels," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, pp. 569 – 580, May 2001.
- [14] X. Y. Tang and A. Zakhor, "Matching pursuits multiple description coding for wireless video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, pp. 566 – 575, June 2002.
- [15] J. García-Frías and P. M. Crespo, "Hidden Markov models for burst error characterization in indoor radio channels," *IEEE Trans. Veh. Technol.*, vol. 46, pp. 1006–1019, Nov. 1997.
- [16] T. S. Rappaport, *Wireless communications: principles and practice*. Prentice Hall, 2000.

- [17] B. Girod, N. Farber, and E. Steinbach, "Performance of the H.263 video compression standard," *J VLSI Signal Process: Syst Signal, Image, Video Tech.*, vol. 17, pp. 101–111, Nov. 1997.
- [18] L. M. Wang, "Bit rate control for hybrid DPCM/DCT video codec," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 4, pp. 509 – 520, Oct. 1994.
- [19] D. Reiminger and D. Raychaudhuri, "Bit-rate characteristics of a VBR MPEG video encoder for ATM networks," in *ICC 93*, May 1993, pp. 517–521.
- [20] Z. C. Zhang and V. Li, "Layered multicast with forward error correction (FEC) for Internet video," in *GLOBECOM '02*, vol. 2, Nov. 2002, pp. 1465–1469.
- [21] T.-W. Lee, S.-H. Chan, Q. Zhang, W. W. Zhu, and Y. Q. Zhang, "Allocation of layer bandwidths and FECs for video multicast over wired and wireless networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, pp. 1059–1070, Dec. 2002.
- [22] P. Frossard and O. Verscheure, "Joint source/FEC rate selection for quality-optimal MPEG-2 video delivery," *IEEE Trans. Image Processing*, vol. 10, pp. 1815–1825, Dec. 2001.
- [23] Y. Wang and Q.-F. Zhu, "Error control and concealment for video communication: A Review," *Proc. IEEE*, vol. 86, pp. 974 – 997, May 1998.
- [24] Y. Wang, S. Wenger, J. T. Wen, and A. K. Katsaggelos, "Error resilient video coding techniques," *IEEE Signal Processing Mag.*, vol. 17, pp. 61 – 82, July 2000.
- [25] J.-W. Suh and Y.-S. Ho, "Error concealment techniques for digital TV," *IEEE Trans. Broadcast.*, vol. 48, pp. 299 – 306, Dec. 2002.
- [26] Y. Wang, Q. F. Zhu, and L. Shaw, "Maximally smooth image recovery in transform coding," *IEEE Trans. Commun.*, vol. 41, pp. 1544 – 1551, Oct. 1993.

- [27] H. Sun and W. Kwok, "Concealment of damaged block transform coded images using projections onto convex sets," *IEEE Trans. Image Processing*, vol. 4, pp. 470 – 477, Apr. 1995.
- [28] G. B. Unal and A. E. Cetin, "Restoration of error-diffused images using projection onto convex sets," *IEEE Trans. Image Processing*, vol. 10, pp. 1836 – 1841, Dec. 2001.
- [29] R. Talluri, "Error-resilient video coding in the ISO MPEG-4 standard," *IEEE Commun. Mag.*, vol. 36, pp. 112 – 119, June 1998.
- [30] S. Fukunaga, T. Nakai, and H. Inoue, "Error resilient video coding by dynamic replacing of reference pictures," in *GLOBECOM '96*, Nov. 1996, pp. 1503–1508.
- [31] C.-C. Liu and S.-C. S. Chen, "Providing unequal reliability for transmitting layered video streams over wireless networks by multi-arq schemes," in *ICIP 99*, Oct. 1999, pp. 100–104.
- [32] D. I. Recommendation H.223/Annex C, "Extensions of recommendation h.223 for highly error-prone environment," 1998.
- [33] H. Liu and M. E. Zarki, "Performance of H.263 video transmission over wireless channels using hybrid ARQ," *IEEE J. Select. Areas Commun.*, vol. 15, pp. 1775 – 1786, Dec. 1997.
- [34] Q. Q. Zhang and S. A. Kassam, "Hybrid ARQ with selective combining for fading channels," *IEEE J. Select. Areas Commun.*, vol. 17, pp. 867 – 880, May 1999.
- [35] A. Majumda, D. G. Sachs, I. V. Kozintsev, K. Ramchandran, and M. M. Yeung, "Multicast and unicast real-time video streaming over wireless LANs," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, pp. 524 – 534, June 2002.

- [36] B. Masnick and J. Wolf, "On linear unequal error protection codes," *IEEE Trans. Inform. Theory*, vol. it-13, pp. 600 – 607, July 1967.
- [37] M. G. Martini and M. Chiani, "Wireless transmission of MPEG-4 video: Performance evaluation of unequal error protection over a block fading channel," in *Proc. Vehicular Technology Conference, 2001*, Rhodes, Greece, May 2001, pp. 2056–2060.
- [38] —, "Proportional unequal error protection for MPEG-4 video transmission," in *ICC*, June 2001, pp. 1033–1037.
- [39] L. H. Xu, "Resource-efficient delivery of on-demand streaming data using UEP codes," *IEEE Trans. Commun.*, vol. 51, pp. 63 – 71, Jan. 2003.
- [40] M. H. Le and R. Liyana-Pathirana, "Unequal error protection codes for wavelet image transmission over W-CDMA, AWGN and Rayleigh fading channels," in *ICT 2003*, Mar. 2003, pp. 1140–1146.
- [41] J. Kim, R. M. Mersereau, and Y. Altunbasak, "Error-resilient image and video transmission over the Internet using unequal error protection," *IEEE Trans. Image Processing*, vol. 12, pp. 121 – 131, Jan. 2003.
- [42] D. S. Park, H. W. Lee, and Y. K. Kim, "Efficient QoS control using unequal error protection (UEP) in physical layer," in *VTC Spring 2002*, May 2002, pp. 250–254.
- [43] A. R. Reibman, H. Jafarkhani, Y. Wang, M. T. Orchard, and R. Puri, "Multiple-description video coding using motion-compensated temporal prediction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, pp. 193 – 204, Mar. 2002.
- [44] D.-M. Chung and Y. Wang, "Multiple description image coding using signal decomposition and reconstruction based on lapped orthogonal transforms," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 895 – 908, Sept. 1999.

- [45] J. Kim, R. M. Mersereau, and Y. Altunbasak, "An integrated multiple-substream unequal error protection and error concealment algorithm for Internet video applications," in *Proceedings., International Conference on Image Processing 2002*, June 2002, pp. 693–696.
- [46] W. Xu and S. S. Hemami, "Efficient partitioning of unequal error protected MPEG video streams for multiple channel transmission," in *2002 International Conference on Image Processing*, Sept. 2002, pp. 721–724.
- [47] C.-S. Kim and S.-U. Lee, "Multiple description coding of motion fields for robust video transmission," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, pp. 999 – 1010, Sept. 2001.
- [48] M. Gallant and F. Kossentini, "Rate-distortion optimized layered coding with unequal error protection for robust Internet video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, pp. 357 – 372, Mar. 2001.
- [49] M. Ghanbari, "An adapted H.261 two-layer video codec for ATM networks," *IEEE Trans. Commun.*, vol. 40, pp. 1481 – 1490, Sept. 1992.
- [50] Y. Wang, S. Panwar, S. Lin, and S. Mao, "Wireless video transport using path diversity: multiple description vs layered coding," in *2002 International Conference on Image Processing*, Sept. 2002, pp. 21–24.
- [51] M. van der Schaar and H. Radha, "Unequal packet loss resilience for fine-granular-scalability video," *IEEE Trans. Multimedia*, vol. 3, pp. 381 – 394, Dec. 2001.
- [52] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, pp. 379 – 423, 1948.
- [53] —, "Coding theorems for a discrete source with a fidelity criterion," *IRE Nat. Conv. Rec.*, pp. 142 – 163, Mar. 1959.

- [54] M. J. Ruf and J. W. Modestino, "Operational rate-distortion performance for joint source and channel coding of images," *IEEE Trans. Image Processing*, vol. 8, pp. 68–84, Mar. 1999.
- [55] R. E. van Dyck and D. J. Miller, "Transport of wireless video using separate, concatenated, and joint source-channel coding," *Proc. IEEE*, vol. 87, pp. 1734 – 1750, Oct. 1999.
- [56] K. Stuhlmuller, N. Farber, M. Link, and B. Girod, "Analysis of video transmission over lossy channels," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 1012–1032, June 2000.
- [57] A. Ortega and M. Khansari, "Rate control for video coding over variable bit rate channels with applications to wireless transmission," in *Proceedings., International Conference on Image Processing 1995*, Oct. 1995, pp. 388–391.
- [58] J. C. Kwon and J. K. Kim, "An adaptive rate allocation to source-channel coding for Internet video," in *GLOBECOM '02*, Nov. 2002, pp. 1985–1988.
- [59] M. Bystrom and T. Stockhammer, "Dependent source and channel rate allocation for video transmission," *IEEE Trans. Wireless Commun.*, vol. 3, pp. 258–268, Jan. 2004.
- [60] M. Gallant and F. Kossentini, "Rate-distortion optimal joint source/channel coding for robust and efficient low bit rate packet video communications," in *Proceedings. Image Processing, 2000*, Sept. 2000, pp. 355–358.
- [61] H. Song and C.-C. Kuo, "Rate control for low-bit-rate video via variable-encoding frame rates," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, pp. 512 – 521, Apr. 2001.

- [62] H.-J. Lee, T. H. Chiang, and Y. Q. Zhang, "Scalable rate control for MPEG-4 video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, pp. 878 – 894, Sept. 2000.
- [63] J. I. Ronda, M. Eckert, F. Jaureguizar, and N. Garcia, "Rate control and bit allocation for MPEG-4," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 1243 – 1258, Dec. 1999.
- [64] M. Bystrom and J. W. Modestino, "Combined source-channel coding schemes for video transmission over an additive white Gaussian noise channel," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 880–890, June 2000.
- [65] G. Cheung and A. Zakhor, "Bit allocation for joint source/channel coding of scalable video," *IEEE Trans. Image Processing*, vol. 9, pp. 340 – 356, Mar. 2000.
- [66] R. Zhang, S. L. Regunathan, and K. Rose, "Video coding with optimal inter/intra-mode switching for packet loss resilience," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 966 – 976, June 2000.
- [67] G. F. Elmasry, "Joint lossless-source and channel coding using automatic repeat request," *IEEE Commun. Mag.*, vol. 37, pp. 953 – 955, July 1999.
- [68] B. He, G. F. Elmasry, and C. N. Manikopoulos, "Joint lossless-source and channel coding using ARQ/go-back-(N, M) for image transmission," *IEEE Trans. Image Processing*, vol. 12, pp. 1610 – 1617, Dec. 2003.
- [69] G. Cote and F. Kossentini, "Optimal intra coding of blocks for robust video communication over the Internet," *Image Commun.*, vol. 15, pp. 25–34, Sept. 1999.
- [70] J. Ribas-Corbera and S. Lei, "Rate control in DCT video coding for low-delay communications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 172 – 185, Feb. 1999.

- [71] Z. Li, C. Zhu, N. Ling, X. Yang, G. Feng, S. Wu, and F. Pan, "A unified architecture for real-time video-coding systems," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, pp. 472–487, June 2003.
- [72] R. Puri, K. Ramchandran, and A. Ortega, "Joint source channel coding with hybrid FEC/ARQ for buffer constrained video transmission," in *1998 IEEE Second Workshop on Multimedia Signal Processing*, Dec. 1998, pp. 567–572.
- [73] T. Stockhammer, H. Jenkac, and C. Weiss, "Feedback and error protection strategies for wireless progressive video transmission," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, pp. 465 – 482, June 2002.
- [74] W. Ding and B. Liu, "Rate control of MPEG video coding and recording by rate-quantization modeling," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 12 – 20, Feb. 1996.
- [75] A. Vetro, H. F. Sun, and Y. Wang, "MPEG-4 rate control for multiple video objects," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 186 – 199, Feb. 1999.
- [76] L.-J. Lin and A. Ortega, "Bit-rate control using piecewise approximated rate-distortion characteristics," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, pp. 446 – 459, Aug. 1998.
- [77] J. Choi, "Distortion policy of buffer-constrained rate control for real-time VBR coders," *IEEE Trans. Image Processing*, vol. 8, pp. 537 – 547, Apr. 1999.
- [78] Z. He, Y. Kim, and S. K. Mitra, "Low-delay rate control and smoothing for video smoothing for video coding via ρ -domain source modeling," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, pp. 928–940, Aug. 2001.

- [79] F. Pan, Z. Li, K. Lim, and G. Feng, "A study of MPEG-4 rate control scheme and its improvements," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, pp. 440 – 446, May 2003.
- [80] L. J. Lin, A. Ortega, and C.-C. J. Kuo, "Cubic spline approximation of rate and distortion functions for MPEG video," in *Visual Communication and Image Processing*, Mar. 1996, pp. 100–104.
- [81] A. Ortego and K. Ramchandran, "Rate-distortion methods for image and video compression," *IEEE Signal Processing Mag.*, vol. 15, pp. 23 – 50, Nov. 1998.
- [82] J. J. Chen and D. W. Lin, "Optimal bit allocation for coding of video signals over ATM networks," *IEEE J. Select. Areas Commun.*, vol. 15, pp. 1002 – 1015, Oct. 1997.
- [83] C. Y. Hsu and A. Ortega, "A Lagrangian optimization approach to rate control for delay-constrained video transmission over burst-error channels," in *ICASSP '98*, May 1998, pp. 2989–2992.
- [84] A. H. Sadka, Ed., *Compressed video communications*. New York: John Wiley and Sons, Ltd, 2002.
- [85] J. Youn, M. T. Sun, and C. W. Lin, "Motion estimation for high performance transcoding," *IEEE Trans. Consumer Electron.*, vol. 44, pp. 649 – 658, Oct. 1998.
- [86] K. T. Fung, Y. L. Chan, and W. C. Siu, "Low-complexity and high quality frame-skipping transcoder," in *ISCAS 2001*, May 2001, pp. 29–32.
- [87] S. J. Wee, J. G. Apostolopoulos, and N. Feamster, "Field-to-frame transcoding with spatial and temporal downsampling," in *ICIP 99*, Oct. 1999, pp. 271–275.
- [88] D. S. Turaga and T. Chen, "Classification based mode decisions for video over network," *IEEE Trans. Multimedia*, vol. 3, pp. 41 – 52, Mar. 2001.

- [89] T. G. Venkatesh and S. Srinivasan, "A pruning based fast rate control algorithm for MPEG coding," in *ICCIMA '99*, Sept. 1999, pp. 403–407.
- [90] Y. M. Huang, J. L. Wu, and C. L. Chang, "A generalized output pruning algorithm for matrix-vector multiplication and its application to compute pruning discrete cosine transform," *IEEE Trans. Signal Processing*, vol. 48, pp. 561 – 563, Feb. 2000.
- [91] R. Dugad and N. Ahuja, "A fast scheme for image size change in the compressed domain," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, pp. 461 – 474, Apr. 2001.
- [92] B. Girod, "Psychovisual aspects of image communication," *Signal Processing*, vol. 28, pp. 239 – 251, Sept. 1992.
- [93] F. Pereira and T. Alpert, "MPEG-4 video subjective test procedures and results," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 32–51, Feb. 1997.
- [94] ITU, "ITU-T recommendation H.263 (draft): Video coding for low bitrate communication," Nov. 1998.
- [95] Z. Li, N. Ling, G. N. Feng, C. Zhu, X. K. Yang, S. Wu, and F. Pan, "Packetization algorithms for MPEG-4 fine granularity over internet," in *3rd IEEE Workshop and Exhibition on MPEG-4 (WEMP4)*, San Jose, California, USA, June 2002, pp. 17–20.
- [96] G. Wade, "Coding techniques, an introduction to compression and error control," Palgrave, Jan. 2000.
- [97] T. P. Chen and T. Chen, "Fine-grained rate shaping for video streaming over wireless networks," in *ICASSP '03*, Apr. 2003, pp. 688–691.
- [98] U. Horn, K. Stuhlmuller, M. Link, and B. Girod, "Robust Internet video transmission based on scalable coding and unequal error protection," *Image Commun.*, vol. 15, pp. 77–94, Sept. 1999.

- [99] G. Cote, S. Shirani, and F. Kossentini, "Optimal mode selection and synchronization for robust video communications over error-prone networks," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 952–965, June 2000.
- [100] Z. G. Li, L. Xiao, C. Zhu, and P. Feng, "A novel rate control scheme for video over the Internet," in *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2002, pp. 2065–2068.
- [101] J. Cai, Q. Zhang, W. Zhu, and C. W. Chen, "An FEC-based error control scheme for wireless MPEG-4 video transmission," in *WCNC*, Sept. 2000, pp. 1243–1247.
- [102] D. W. Redmill and N. G. Kingsbury, "The EREC: an error-resilient technique for coding variable-length blocks of data," *IEEE Trans. Image Processing*, vol. 5, pp. 565 – 574, Apr. 1996.
- [103] J. Chen, U. V. Koc, and K. J. R. Liu, Eds., *Design of digital video coding systems: a complete compressed domain approach*. New York: Marcel Dekker, Inc, 2002.
- [104] H. M. Hang and J. J. Chen, "Source model for transform video coder and its application - part i: fundamental theory," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 287–298, Apr. 1997.
- [105] G. J. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Processing Mag.*, vol. 15, pp. 74–90, Nov. 1998.
- [106] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. J. Sullivan, "Rate-constrained coder control and comparison of video coding standards," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, pp. 688– 703, July 2003.
- [107] R. J. Freund and W. J. Wilson, Eds., *Regression Analysis: Statistical Modeling of a Response Variable*. Academic Press, 1998.
- [108] ITU/SG15, "Video codec test model, TMN8," June 1997.

- [109] T. Chiang and Y. Q. Zhang, "A new rate control scheme using quadratic rate distortion model," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 246–250, Feb. 1997.
- [110] Y. Kim and H. Kim, "A new buffer control strategy for image data compression," *IEEE Trans. Consumer Electron.*, vol. 40, pp. 932 – 937, Nov. 1994.
- [111] H. S. Oh and H. K. Lee, "Adaptive rate control scheme for very low bit rate video coding," *IEEE Trans. Consumer Electron.*, vol. 42, pp. 974 – 980, Nov. 1996.
- [112] G. J. Sullivan, T. Wiegand, and T. Stockhammer, "Using the draft h.26l video coding standard for mobile applications," in *ICIP 2001*, Greece, Oct. 2001, pp. 388–391.
- [113] Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, "Joint final committee draft (JFCD) of joint video specification (ITU-T Rec. H.264 — ISO/IEC 14496-10 AVC)," in *JVT-D157*, July 2002.
- [114] R. Schafer, T. Wiegand, and H. Schwarz, "The emerging H.264/AVC standard," in *EBU Technical Review*, www.ebu.ch, Jan. 2003.
- [115] Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, "Working draft number 2, revision 0 (wd-2)," in *JVT-B118*, Jan. 2002.
- [116] T. Wiegand and B. Girod, "Lagrangian multiplier selection in hybrid video coder control," in *ICIP 2001*, Greece, Oct. 2001, pp. 388–391.
- [117] S. W. Ma, W. Gao, Y. Lu, and H. Q. Lu, "Rate control for advance video coding (AVC) standard," in *ISCAS '03*, May 2003, pp. 892–895.
- [118] Z. G. Li, F. Pan, K. P. Lim, G. N. Feng, X. Lin, and S. Rahardja, "Adaptive frame layer rate control for H.264," in *ICME 2003*, July 2003, pp. 581–584.

- [119] Z. G. Li, W. Gao, F. Pan, S. W. Ma, K. P. Lim, G. N. Feng, X. Lin, S. Rahardja, H. Q. Lu, and Y. Lu, "Adaptive rate control with HRD consideration," in *JVT-H014*, Geneva, Switzerland, May 2003.
- [120] M. Jiang, X. Yi, and N. Ling, "On enhancing H.264 rate control by PSNR-based frame complexity estimation," in *IEEE International Conference on Consumer Electronics (2005 ICCE)*, Las Vegas, Nevada, USA, Jan. 2005, pp. 231–232.
- [121] ITU/SG15, "Video codec test model, TMN5," Jan. 1995.
- [122] M. Jiang, X. Yi, and N. Ling, "Improved frame-layer rate control for H.264 using MAD ratio," in *Proceedings of the 2004 IEEE International symposium on circuits and systems (ISCAS)*, Vancouver, Canada, May 2004, pp. III–813–III–816.
- [123] F. Pan, Z. G. Li, K. Lim, and G. Feng, "A study of MPEG-4 rate control scheme and its improvements," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, pp. 440–446, May 2003.

Appendix A

In this appendix, we give a proof of linear regression technique. Suppose that

$$Y = H \times \tilde{\theta} \tag{1}$$

where $Y = [y_1, y_2, y_3, \dots, y_{n-1}, y_n]^T$, $\tilde{\theta} = [\tilde{m}, \tilde{b}]^T$, and H is given as follows

$$H = \begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ x_1 & 1 \\ . & . \\ . & . \\ x_{n-1} & 1 \\ x_n & 1 \end{bmatrix} \tag{2}$$

It is assumed that n is fixed and the reason is clarified in the chapter 2.

Proof

$$\therefore Y = H \times \tilde{\theta} \quad (3)$$

$$\begin{aligned} \therefore H^T \times Y &= H^T \times H \times \tilde{\theta} \\ &= (H^T H) \times \tilde{\theta} \end{aligned} \quad (4)$$

$$\begin{aligned} \therefore (H^T H)^{-1} \times H^T \times Y &= (H^T H)^{-1} \times (H^T H) \times \tilde{\theta} \\ &= \tilde{\theta} \end{aligned} \quad (5)$$

On the other hand,

$$\begin{aligned} H^T H &= \begin{bmatrix} x_1 & x_2 & x_3 & \cdot & \cdot & x_{n-1} & x_n \\ 1 & 1 & 1 & \cdot & \cdot & 1 & 1 \end{bmatrix} \times \begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ x_3 & 1 \\ \cdot & \cdot \\ \cdot & \cdot \\ x_{n-1} & 1 \\ x_n & 1 \end{bmatrix} \\ &= \begin{bmatrix} \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i \\ \sum_{i=1}^n x_i & n \end{bmatrix} \end{aligned} \quad (6)$$

and

$$H^T Y = \begin{bmatrix} x_1 & x_2 & x_3 & \cdot & \cdot & x_{n-1} & x_n \\ 1 & 1 & 1 & \cdot & \cdot & 1 & 1 \end{bmatrix} \times \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \cdot \\ \cdot \\ y_{n-1} \\ y_n \end{bmatrix} \quad (7)$$

$$= \begin{bmatrix} \sum_{i=1}^n x_i y_i \\ \sum_{i=1}^n y_i \end{bmatrix}$$

It is therefore can be derive $\tilde{\theta}$ from equations (3)-(7)

$$\therefore \tilde{\theta} = (H^T H)^{-1} H^T Y \quad (8)$$

$$\therefore \begin{bmatrix} \tilde{m} \\ \tilde{b} \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i \\ \sum_{i=1}^n x_i & n \end{bmatrix}^{-1} \times \begin{bmatrix} \sum_{i=1}^n x_i y_i \\ \sum_{i=1}^n y_i \end{bmatrix} \quad (9)$$

$$= \frac{1}{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2} \begin{bmatrix} n & -\sum_{i=1}^n x_i \\ -\sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 \end{bmatrix} \times \begin{bmatrix} \sum_{i=1}^n x_i y_i \\ \sum_{i=1}^n y_i \end{bmatrix}$$

$$= \frac{1}{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2} \begin{bmatrix} n \sum_{i=1}^n x_i y_i - \sum_{i=1}^n x_i \sum_{i=1}^n y_i \\ -\sum_{i=1}^n x_i \sum_{i=1}^n x_i y_i + \sum_{i=1}^n x_i^2 \sum_{i=1}^n y_i \end{bmatrix}$$