MULTIMEDIA STREAMING OVER WIRELESS CHANNELS

by

HÜLYA SEFEROĞLU

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfillment of
the requirements for the degree of
Master of Science

Sabanci University
August 2005

MULTIMEDIA STREAMING OVER WIRELESS CHANNELS

APPROVED BY

Assist. Prof. Dr. Özgür GÜRBÜZ ....
(Thesis Supervisor)

Assist. Prof. Dr. Özgür ERÇETİN ....
(Thesis Supervisor)

Assoc. Prof. Dr. Yücel ALTUNBAŞAK ....

Assist. Prof. Dr. Hakan ERDOĞAN ....

Assist. Prof. Dr. Tonguç ÜNLÜYURT ....

DATE OF APPROVAL: ....

To my family

Acknowledgments

I would like to thank everybody who has contributed to the realization of this thesis. Special thanks go to my supervisor Assist. Prof. Dr. Özgür Gürbüz and my co-supervisor Assist. Prof. Dr. Özgür Erçetin for their guidance and support all through my stay at the Department of Electrical Engineering and Computer Science. I am also grateful to Assoc. Prof. Dr. Yücel Altunbaşak for his invaluable contributions to this thesis.

I want to express my gratitude to my colleagues and members of Sabancı University for creating stimulating work environment. I am indebted to the members of the jury of my thesis: Assoc. Prof. Dr. Yücel Altunbaşak, Assist. Prof. Dr. Hakan Erdoğan, and Assist. Prof. Dr. Tonguç Ünlüyurt for reviewing my thesis and for their useful remarks.

Especially, I would like to thank my family for their endless support during all my studies. It would not have been possible to manage everything without them.

Abstract

The improvements in mobile communication systems have accelerated the development of new multimedia streaming techniques to increase the quality of streaming data over time varying wireless channels. In order to increase multimedia quality, error control schemes are indispensable due to time-varying and erroneous nature of the channel. However, relatively low channel capacity of wireless channels, and dependency structure in multimedia limit the effectiveness of existing error control schemes and require more sophisticated techniques to provide quality improvement on the streaming data. In this thesis, we propose sender driven multimedia streaming algorithms that incorporate error control schemes of FEC, ARQ, and packet scheduling by considering media and channel parameters such as packet importance, packet dependencies, decoding deadlines, channel state information, and channel capacity. Initially, we have proposed a multi-rate distortion optimization framework so as to jointly optimize FEC rate and packet selection by minimizing end-to-end distortion to satisfy a specified Quality of Service under channel capacity constraint. Minimization of end-to-end distortion causes computational complexity in the rate distortion optimization framework due to dependency in encoded multimedia. Therefore, we propose multimedia streaming algorithms that select packet and FEC rate with reduced computational complexity and high quality as compared with multi-rate distortion optimization framework. Additionally, protocol stack of a UMTS cellular network system with W-CDMA air interface is presented in order to clarify the relation between proposed multimedia streaming algorithms and UMTS system that is used in simulations. Finally, proposed algorithms are simulated and results demonstrate that proposed algorithms improve multimedia quality significantly as compared to existing methods.

Özet

Mobil iletişim sistemlerindeki gelişmeler, zamanla değişen kanallar üzerinden aktarılan verinin kalitesini artırmak amacıyla, yeni çoğul ortamlı duraksız işlem tekniklerinin gelişmesini hızlandırmştır. Çoğul ortamlı verinin kalitesini artırmak için, kanalın zamanla değişen ve gürültülü yapısından dolayı, hata kontrol tekniklerinin kullanılması kaçınılmazdır. Fakat, kanalın kapasitesinin düşük olması ve çoğul ortamlı verinin birbirine bağımlı olması, temel hata kontrol tekniklerinin verimliliğini kısıtlar ve iletilen verinin kalitesini artırmak amacıyla daha gelişmiş tekniklerin kullanılmasını gerektirir. Bu tezde, temel hata kontrol teknikleri olan gönderme hata düzeltimi, hata durumu yineleme ve paket zamanlama, çoğul ortam ve kanal parametreleri göz önüne alarak birleştirilmiş ve verici tarafında çalışan çoğul ortam duraksız işlem algoritmaları geliştirilmiştir. Çoğul ortam ve kanal parametreleri, gönderilen paketin önemi, paketler arasındaki bağımlılık, paketlerin çözülebilmesi için son zaman, kanal durumu ve kapasitesi olarak tanımlanır. İlk olarak, kanal kapasite kısıtlamaları altında belli bir servis kalitesine ulaşmak için, uçtan uca bozunum miktarını enküçülterek, gönderme hata düzeltim oranını ve paket seçimini birlikte eniyilemek için çoklu-hız bozunum eniyileme tekniği geliştirilmiştir. Uçtan uca bozunum enküçültmesi, kodlanmş çoğul ortam verisinin içerisindeki bağımlılıktan dolayı, hesaplama karmaşıklığına sahiptir. Bu sebeple, düşük hesap karmaşıklığı olan ve önerilen çoklu-hız bozunum eniyileme tekniğine yakın servis kalitesine sahip gönderme hata düzeltim oranı ve paket seçimini yapan çoğul ortam duraksız işlem algoritmaları geliştirilmiştir. Bunlara ek olarak, benzetim sürecinde kullandığımız W-CDMA hava arayüzüne sahip hücresel ağ sistemi olan UMTS sistemi ile önerilen çoğul ortam duraksız işlem algoritmaları arasındaki bağlantıyı göstermek amacıyla UMTS sisteminin protokol yığını açıklanmıştır. Sonuçta, önerilen algoritmalar UMTS yapısı içerisinde kullanılmştır ve sonuçlar önerilen tekniklerin var olan tekniklere göre çoğul ortam veri kalitesini önemli ölçüde arttırdığını göstermiştir.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

In the third generation mobile communication systems, there is an increasing demand for multimedia services such as real-time or near real-time audio and video streaming. Multimedia compression schemes such as H.264 and MPEG-4 [3], [4] have made possible the provision of such multimedia services in the current wireless channels. These schemes remove temporal and spatial redundancy found in the consecutive video frames so that the required bit rate necessary to transfer the video successfully is reduced. However, removing the redundancy found in the video frames makes the data more vulnerable to channel errors. Furthermore, such schemes usually result in dependencies among the consecutive frames. Consequently, an error occurring in a frame may propagate to the subsequent frames and cause significant distortion in the audio or video.

Another challenge in transferring multimedia data over wireless channels is the sensitivity of the multimedia data to packet delays. Note that multimedia data can be considered in terms of frames representing a portion of the audio or video stream. Each frame has its own deadline before which must be received, decoded and displayed in order to be useful. A frame received after its deadline cannot be combined with the preceding frames and becomes obsolete. Thus, each packet missing its deadline results in distortion in the video.

The final challenge that needs to be overcome is the adverse effects of the limited capacity and time varying quality of wireless channels. Unlike wireline channels, wireless channels experience random packet losses due to fading, shadowing and interference. The causes of random losses may change in time due to the mobility of the user, number of users in the area, etc. Thus, transmission and error correction techniques developed should be adaptive to the changes in the channel.

In this thesis, in order to address the above challenges, we develop a cross-layer transport protocol. The protocol has main parts working jointly: (1) A packet scheduler, which selects the most appropriate packet to transmit in each instance from an incoming stream by taking into account the packet deadlines, their dependencies and past history of unsuccessful transmissions; (2) Forward Error Correcting (FEC) encoder, which selects the FEC rates according to the packet to be transmitted, the quality of channel and the number of previous unsuccessful attempts to transmit the packet.

1.2 Objectives

One objective in this thesis is to provide a rate distortion optimized packet and FEC selection scheme by modifying the rate distortion optimization framework proposed in [1], and [2]. The framework in [1] optimizes packet selection by minimizing end-to-end distortion. The distortion function is defined by using packet deadlines, importances, dependencies, and channel state information. The same distortion function is used in [2] that proposes an incremental redundancy scheme in addition to packet scheduling. However, the time varying and error prone nature of the wireless medium necessitates the use of FEC in addition to packet scheduling. Therefore, we formulated multi-rate distortion optimization (MRDO) framework by using the similar distortion function, and similar formulation as in [1], [2]. MRDO jointly optimizes packet and FEC rate.

The end-to-end distortion function in MRDO is highly complicated due to the dependency structure between frames in the encoded multimedia. Hence, the optimization is computationally complex, and this complexity makes the optimization algorithm inappropriate for real time applications. Another objective of this thesis is, therefore, to propose transport algorithms performing packet and FEC selection with reduced computational complexity, and high multimedia quality as compared with MRDO. With this purpose, we proposed two reduced complexity (RC) algorithms that incorporate packet deadline, importance, and channel state information in order to determine best packet and FEC rate.

Our final objective is to insert the proposed transport algorithms to the third generation cellular system, Universal Mobile Telecommunications System (UMTS) with wide-band code division multiple access (W-CDMA) air interface. The end-to-end protocol stack of the UMTS system is examined and a new system model that works with the transport algorithms is presented.

1.3 Organization of the Thesis

The thesis is organized as follows: Chapter 2 contains the background information on current multimedia compression standards, basic and advanced methods for error control and concealment, third generation mobile communication systems, and wireless channel modeling. Chapter 3 explains the system model composed of a gateway, mobile, and the wireless channel together with problem formulation. Problem intended to be solved in this thesis is also formulated in Chapter 3. Chapter 4 presents the formulation of the multi-rate distortion algorithm and its solution method. Chapter 5 describes reduced complexity algorithms. The simulation model and simulation results of the proposed transport algorithms are presented in Chapter 6, and finally Chapter 7 summarizes our conclusion in this thesis.

Chapter 2

Background

2.1 Multimedia Compression

In order to use current technology efficiently data compression is needed while data is stored or transmitted. Multimedia compression is an important issue due to enormous amount of raw data and increasing demand for multimedia services. This thesis focuses on video transmissions, so video formats and compression schemes are briefly explained in the following paragraphs.

Recorded continuous video data is sampled in time and space, and each sample is quantized. Quantization is a process to define the color level at the sampled point. Definition of color level varies according to the choice of color space type [5]. For example, RGB is a color space that represents an image sample with three numbers that show the proportions of Red, Green, and Blue, since any color is represented by using these colors. However, human visual system is more sensitive to brightness than color. Therefore, another color space, YUV, is proposed. In this color space, a sampled point is represented by luminance sample Y, and two color components(chrominance); U and V.

There are three sampling patterns for Y, U, and V that are supported by current video compression standards, [5]. 4:4:4 sampling means that three components (Y, U, and V) have the same resolution and hence a sample of each component exist at every pixel position. The numbers indicate the relative sampling rate of each component in the horizontal direction. In 4:2:2 sampling, the chrominance components have same the vertical resolution as the luma but half the horizontal resolution. More popular sampling pattern is 4:2:0 where chrominance components have half the vertical resolution of luma. 4:2:0 sampling is widely used for video conferencing, digital television and digital versatile disk.

Video compression standards can compress various video formats, [5]. These are common intermediate format (CIF), quarter common intermediate format (QCIF), sub-QCIF, and 4CIF. CIF and QCIF are used widely especially for video conferencing and mobile multimedia applications. CIF video formats have sizes 352×288 and 1216512 bits per frame when 4:2:0 sampling format is used while QCIF videos have sizes 176×144 and each frame has 304128 bits when again 4:2:0 sampling format is used.

Video compression standards define representation of encoded data and a method of decoding syntax to reconstruct visual information. Standardization process is evolved in two branches; MPEG and H.26L compression standards.

2.1.1 MPEG

Moving picture experts group (MPEG), a working group of the International Organization for Standardisation (ISO) published MPEG group of compression standards. Historically, MPEG-1 [6] was developed for video storage and playback on compact disks. MPEG-2 [7] is a standard for digital broadcasting of television. Finally MPEG-4 [3] has been developed. MPEG-4 attempts to increase the coding gain using the video objects of any shape in the video signal. The objective is to divide a video frame into discrete objects which will be coded separately. For example, in the scene where a car moves, there are at least two objects, [8]. These objects are compressed independently. This phenomenon provides us to differentiate objects depending on the content of the video and increase the quality of compression of objects according to these objects.

2.1.2 H.264

The video coding experts group of the International Telecommunication Union Telecommunication Standardisation Sector (ITU-T) develops standards known in general H.26L.

H.261 [9] is the first widely used standard for videoconferencing over ISDN circuit switched networks, [5]. These networks operate at multiples of 64 kbps and the standard was designed to offer computationally simple video conferencing for these bitrates. In order to improve compression performance of H.261, H.263 was proposed. H.263 operates over a wide range of circuit and packet switched networks by providing 30 kbps bit rate with basic video quality, [10]. Finally, H.264 provide compression by using a more efficient, robust, and practical way as compared to previous standards. In the following paragraphs, H.264 is briefly discussed.

H.264 has been proposed for applications using low bit rate and requiring low picture resolution. This makes H.264 more suitable candidate for mobile communication and video conferencing. H.264 video compression standard can be examined in two parts. The first part is the video coding layer (VCL) and the other is the network abstraction layer (NAL).

2.1.2.1 Video Coding Layer

Video coding layer (VCL) receives digitized video in the form of consecutive pictures (frames). VCL layer consists of spatial and temporal prediction, and transform coding, [5]. That means each frame is predicted by temporal and spatial prediction, and transform coded. However, the first frame and frames coming after specific duration are not coded by using temporal prediction. This coding is called intra-coding, and intra-coded frames are called I frames. The frames between two I frames are inter-coded where temporal prediction in addition to spatial prediction and transform coding. There are two types of inter-coded frames depending on the direction of temporal prediction where P frames are only predicted from previous frames and B frames use bi-directional temporal prediction. The temporal prediction induces dependency among coded frames. The dependency structure for a video containing I, P, and B frames is shown in Fig. 2.1.

The compression process is performed as the following. Each color component of every frame is divided into macroblocks. These macroblocks are predicted by using temporal and spatial prediction.

Temporal prediction removes the redundancy in the repeated information between successive frames. Motion compensation is used for temporal prediction. For

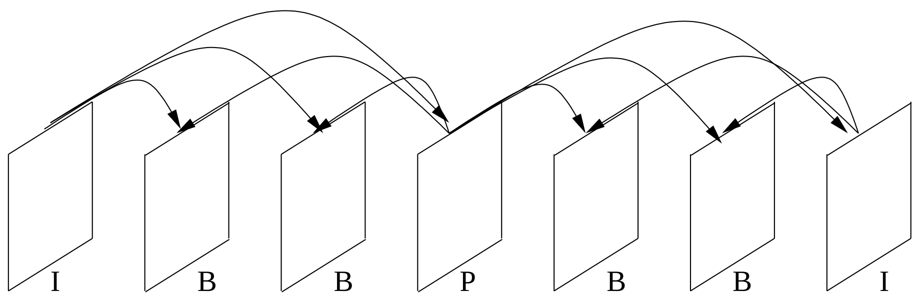

Figure 2.1: Dependency structure

each block in the current frame a search is done in the previous frame (reconstructed frame) to determine the best match. If a match is found, it is then subtracted from the currently considered block. If no match is found, the block is transmitted without temporal prediction. Each block is therefore accompanied with a motion vector to enable the decoder to find the appropriate block in the previously reconstructed frame.

Spatial prediction removes the redundancy among blocks in the same frame. VCL reads one block at a time in a frame. The residual between current and previous block is calculated and this residual is used for transform coding. Each color component of the prediction residual is subdivided into smaller blocks. Each block is quantized with separable integer transform similar to discrete cosine transform (DCT), in order to further reduce temporal redundancy. DCT is a popular transform since it achieves significant energy compaction, [8]. However, the transform coefficients of DCT are floating point numbers that may result errors. Therefore, similar integer transform is used. Then the transform coefficients are quantized by using scalar quantization. The goal of quantization is to represent a very large set of values with a much smaller set, and provide the use of fewer bits, [8]. The quantized transform coefficients are entropy coded. Entropy coding assigns a codeword to each symbol of transform coefficients. The entropy coded data is transmitted with the side information for either Intra-frame or Inter-frame prediction.

As mentioned video is divided into macroblocks before compression. A fixed number of macroblocks in a same frame compose a slice. One frame may be composed of more than one slice. Depending on the type of coding of the frame, slices are called as I, P, or B slices. In addition SP and SI slices are also possible where they provide switching between coded streams. Structure of a slice is shown in

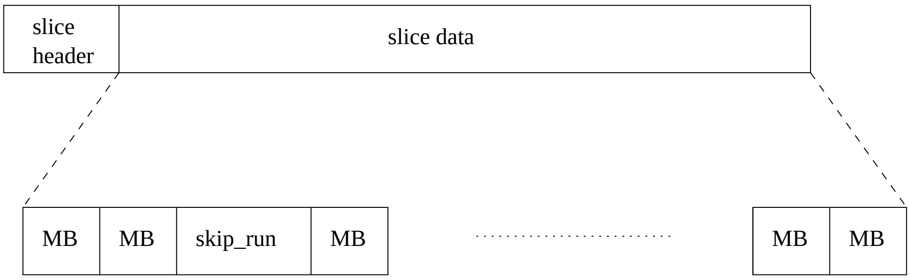

Figure 2.2: Slice syntax

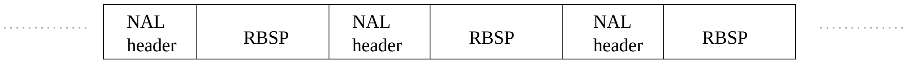

Figure 2.3: Sequence of NAL units

Fig. 2.2 that is received from [5].

Slice header as shown in Fig. 2.2, contains slice type and the coded picture to which the slice belongs. Slice data consists of a series of coded macroblocks and/or skipped (not coded) macroblocks. A macroblock consists of 16×16 sample region of the corresponding frame. This 16×16 region includes all luma and chroma samples. The slices are the output of the video coding layer, and they are passed to network abstraction layer.

2.1.2.2 Network Abstraction Layer

Network abstraction layer (NAL) is the interface between the video coding layer and the outside world. Specifically, VCL encoder produces slices. The NAL encoder encapsulates the slice output of the VCL encoder into NAL units which are suitable for transmission over packet networks or use in packet oriented environments. A NAL unit consists of a header and the payload string as shown in Fig. 2.3.

In Fig. 2.3, raw byte sequence payload (RBSP) is a set of data corresponding to coded video data or header information. An example for a sequence of RBSP is given in Fig. 2.4.

Each of the units in Fig. 2.4 is transmitted in a separate NAL unit by adding a

Figure 2.4: Example sequence of RBSP elements

0				8		16 ... 31
V	P	X	CC	M	Payload Type	Sequence Number
Time Stamp						
SSRC Identifier						
CSRC Identifier						

Figure 2.5: RTP header

NAL header to each of them. It is clear from Fig. 2.4 that each RPSB element and hence each NAL unit conveys different type of information. For example, sequence parameter set contains global parameters for all pictures in the stream. Picture parameter set, on the other hand, contains information about the picture. I slices and P slices are coded bitstreams. Picture delimiter puts a bound between two consecutive frames.

The NAL unit header in Fig. 2.3 is a description to separate the NAL units and/or identify the properties of the NAL unit. NAL unit encapsulation is performed by using Annex B or real time protocol (RTP) encapsulations, [4].

Annex B, integral part of the H.264 standard, is a simple method for describing boundaries between consecutive NAL units. Basically, NAL units are separated by a simple 4-byte sequence, 00 00 00 01, [12]. Finding a NAL unit boundary is therefore as simple as searching this byte sequence.

A more complicated NAL unit encapsulation method is RTP. The RTP header indicates the type of the NAL unit, the presence of bit errors or syntax violations in the NAL unit payload, and information regarding the relative importance of the NAL unit for the decoding process. Therefore, RTP provides more robust encapsulation than Annex B. In particular RTP header for H.264 video is constructed as in Fig. 2.5,[13].

Some parameters in Fig. 2.5 are not set for H.264 encoded video, [14]. These parameters are version (V) that shows the version of RTP. Padding (P) is used to indicate that packet contains one or more additional padding bytes that are not part of the payload. When the extension (X) field is set, the fixed header must be followed by exactly one header extension. CSRC count (CC) specifies the number of CSRC identifiers that follow the fixed header. Payload type (PT) identifies the format of RTP payload. SSRC is the randomly chosen number used to distinguish synchronization sources within the same RTP session. CSRC specifies the contributing sources for the payload contained in the packet. The other parameters shown in Fig.b.1. is set for using H.264. Marker bit (M) is set and used for Sequence number (SN) is set and used for determining the decoding order for the NAL unit. Timestamp is set to show the sampling timestamp of the content.

2.2 Error Control and Concealment

In the literature, there has been significant previous work on error control, recovery, and concealment techniques that discuss different approaches to increase the quality of streaming multimedia. The purpose of the error control and recovery techniques such as Forward Error Correction (FEC), and Automatic Repeat reQuest (ARQ) is to provide error free transmission for the related data. Error concealment techniques, on the other hand, attempt to suppress the negative effects of the errors that are not recovered by FEC or ARQ. There are many techniques for error concealment. However, the general idea is to use the characteristics of the source or encoder, and the channel so as to increase quality of the multimedia as much as possible.

Basic error control and recovery techniques are discussed in more detail in Section 2.2.1. In order to increase the effectiveness of these basic methods, more sophisticated ones that incorporate the properties of the encoded data and the channel conditions use a combination of more than one error control and/or concealment techniques. These methods are grouped into source and transport level error control and concealment techniques in this section by following the same classification as in [15].

2.2.1 Basic Methods

In this section, basic error control and concealment methods essential for the current sophisticated techniques are explained.

2.2.1.1 Forward Error Correction

Forward error correction (FEC) is a method for error control and recovery for data transmission where the receiver has the capability to detect and correct fewer than a predetermined number or bits or symbols corrupted by transmission errors [16]. FEC is accomplished by adding redundancy to the transmitted data using a predetermined algorithm. Adding redundancy with FEC coding reduces data bandwidth of the channel since a portion of the channel bandwidth is used for parity bits of the FEC code. FEC techniques, however, are indispensable for channels with high error rates, e.g. wireless channels.

The two main categories of FEC are block coding and convolutional coding. Block codes operate on fixed-size blocks of bits or symbols of predetermined size, while convolutional codes consider bit or symbol streams. Block and convolutional codes are frequently combined in concatenated coding schemes in which the convolutional code does most of the work and the block code detects any errors made by the convolutional decoder.

There are many types of block codes, but the most important one is Reed-Solomon coding that is explained in the following sub-section. Hamming, Bose-Chaudhuri-Hocquenghem (BCH), and Golay codes are other examples of block codes. Hamming codes are a family of single-bit error correcting codes which are the first major class of linear block error correcting codes, [17]. An important property of these codes is that their construction is very easy. BCH codes are important class of cyclic codes. BCH codes can be designed to correct any number of errors. One bit error correcting BCH codes are the Hamming codes. The disadvantage of the BCH codes is the growing complexity when the number of errors to be corrected increases. Golay codes is another type of block codes that are more powerful than the Hamming codes but their code rate is lower and decoders are more complex, [16].

One important class of convolutional codes is punctured convolutional codes, such as rate compatible punctured convolutional codes (RCPC), and rate compatible punctured turbo coding (RCPT). Punctured convolutional codes provide various code rate opportunities. The most recent development in error correction is turbo coding that can perform near to the Shannon limit.

Reed Solomon Codes

Reed Solomon (RS) codes are an example of block codes that works on fixed size blocks of symbols. A symbol is composed of m bits, where m is any positive integer having a value greater than 2, [16]. RS (n, k) coded data are composed total of n symbols, where k of them are data symbols. There is a relation between symbol size m with n and k as in the following inequality;

$$0 < k < n < 2^m + 2. \tag{2.1}$$

The difference between n and k is called parity, and the relation between error correcting capability of the code and the parity is given by $n - k = 2t$ where t represents error correcting capability of the code. This relation originates the minimum distance of the RS codes achieving the largest possible minimum distance. The minimum distance is defined as the number of symbols which the sequences differ. The minimum distance;

$$d_{min} = n - k + 1. \tag{2.2}$$

Therefore, the code is capable of correcting any combination of t or fewer errors, where t can be expressed as in [16];

$$t = \lfloor \frac{d_{min} - 1}{2} \rfloor = \lfloor \frac{n - k}{2} \rfloor. \tag{2.3}$$

Eq.(2.3) shows that RS codes correcting t symbol errors requires no more than $2t$ parity symbols. That means, for each error, one redundant symbol is used to locate the error, another redundant symbol is used to find its correct value. In short, Reed Solomon codes can correct any combination of errors when they are less than t.

2.2.1.2 Automatic Repeat-reQuest

Automatic Repeat-reQuest(ARQ) is an error control mechanism for data transmission in which the receiver detects transmission errors in a message and automatically requests a retransmission from the transmitter, or transmitter does not receive an acknowledgement (ACK) from the receiver and decides on retransmission depending on the type of ARQ. A few types of ARQ protocols such as stop-and-wait-ARQ, go-back-N ARQ, selective repeat ARQ, type-I, and type II hybrid ARQ are examined in the remainder of this section.

Stop-and-Wait ARQ

In Stop-and-Wait ARQ scheme transmitter and receiver work on the delivery of one packet at a time, [18]. The transmitter sends a packet to the receiver, then stops transmission and waits for ACK. If ACK is received, another packet is transmitted. If ACK is not received, the transmitter retransmits the packet, and begins waiting for ACK again. This process continues until ACK for the packet is received by the transmitter, or the limit for the maximum number of transmissions is reached.

Stop and Wait ARQ performs better for channels with low propagation channel.

Go-Back-N ARQ

The inefficiency of Stop-and-Wait ARQ due to delay is defeated by using Go-Back-N ARQ. This method sends packets continuously, and at the same time waits ACKs for previously transmitted packets. Hence, the channel is used upto its capacity.

Go-Back-N ARQ works as the following, [18]: transmitter sends packets continuously to the receiver, and monitors the backward channel for acknowledgement packets. If one ACK packet is not received for a packet, this packet and all packets coming after this packet is transmitted again. N is the difference between the packet number of the packet that is to be transmitted in the next transmission time and the number of packet whose ACK is to be waited by the next transmission time.

In this protocol some packets are sent to the receiver more than once although previous transmission are successful, and this reduces the throughput of the protocol.

Selective Repeat ARQ

In channels that have high error rates, the Go-Back-N ARQ is inefficient because of the need to transmit the packet whose ACK is not received and all the subsequent packets, [18].

Selective Repeat ARQ transmits packets continually as in Go-Back-N ARQ scheme. However, if one packet is corrupted, and ACK is not received for this packet, only the corrupted packet is retransmitted to the receiver. Therefore, the receiver accepts error-free packets but these packets may be out of order. The receiver rearranges the received packets according to their frame numbers.

Type-I Hybrid ARQ

Type-I hybrid ARQ scheme includes parity bits for both error detection and error correction in every transmitted packet [19]. If the number of erroneous bits in a received packet is within the error correction capability of the code, the errors are corrected and the decoded packet is accepted by the receiver. If the packet is corrupted, the packet is rejected and a retransmission is requested. The transmitter sends the original packet again.

Appropriately designed error correction codes can correct errors in the transmitted packets, hence they reduce the number of retransmissions compared to ARQ schemes that do not include error correcting codes. Since wireless channels are time varying, some type-I hybrid ARQ algorithms have been developed to adopt the code rate in each (re)transmission to the channel conditions. A disadvantage of type-I hybrid ARQ scheme is that the corrupted packets are discarded by the decoder even if they might contain some useful data.

Type-II Hybrid ARQ

Type-II hybris ARQ scheme includes parity bit for every packet that is (re)transmitted as in type-I hybrid ARQ scheme, [19]. However, the corrupted packets are not discarded by the receiver.

In this method, a packet is coded with a rate compatible code such as rate compatible punctured convolutional codes or rate compatible punctured turbo codes. Rate compatible codes provide different levels of error protection depending on the parity used. The parity size is changed with puncturing. Type-II hybrid ARQ sends a packet coded with rate compatible code with very high rate. This packet is transmitted to the receiver, if packet is not decoded, the supplemental bits, which were deleted by puncturing is transmitted to the receiver. The receiver combines the corrupted packet and its the supplemental bits. This results in a coded packet with a lower rate error correcting code. The decoder then tries to decode this packet. If packet is not recovered again a more parity bits are transmitted from the transmitter, or the packet itself is transmitted again.

2.2.1.3 Interleaving

Interleaving is an error control technique that re-organizes consecutive packets in time in order to avoid consecutive packet corruptions. In most of the multimedia compression schemes, consecutive packets are dependent on each other. This causes more reduction in quality when successive packets are lost. Therefore, interleaving is especially beneficial for channels having long burst of errors.

2.2.2 Advanced Methods

Error control and concealment techniques are divided into two depending on the level these techniques applied. Source level techniques operate in application layer and includes their source coders to handle error control and concealment process. Transport level techniques work in transport protocol, and use the properties of the encoded data. Both source level and transport level techniques may use one or more of the basic methods described in Section 2.2.1 in addition to packet scheduling that decides which packet is chosen for transmission among several packets in the transmitter.

2.2.2.1 Source Level Error Control and Concealment

Layered Coding with Transport Prioritization

Layered coding is an effective scheme for providing error resilience in a multimedia streaming system. Multimedia data is partitioned into multiple layers that differ depending on their importance. In general, layers are grouped into base layers and enhancement layers. The base layers provide sufficient information to be able to decode video stream with an acceptable quality. Enhancement layers provide higher quality improvement over base layer video quality.

Layered coding with transport prioritization give higher importance to base layers, by providing a higher degree of error protection in order to satisfy a certain quality of service. If an enhancement layer is received correctly while its base layer is not received, the information contained in the enhancement layer is not used. Therefore, providing higher error protection to base layers (unequal error protection) is reasonable. One example for unequal error protection is given in [20] that divides the H.263 encoded video into two layers. The base layer is channel coded with a fixed FEC rate more robust than the fixed FEC rate used for protection of the enhancement layer. Another example is [21] that proposes a method to divide MPEG-4 video bitstream into several layers, and observes the increase in the quality when number of layers is increased. However, these two methods only consider source data where channel condition is also an important parameter for error control. Unequal error protection can be achieved by applying higher reliable ARQ scheme on more important data as in [22] that proposes and ARQ type selection scheme depending on the priority. H.263 encoded video stream is divided into three layers of priority: the first layer consists of high sensitive data composed of headers of group of blocks or macroblocks, the second layer is medium sensitive data including the transform coefficients of intra-coded frames, and the third layer contains low sensitive data, transform coefficients of inter-coded frames. ARQ is performed according to sensitivity level of data by considering the fact that corrupted packet with low sensitive video data will result less visual distortion and no ACK is required while high sensitive video data necessitates the use of ARQ. A similar technique is proposed in [23] where source signal is divided into multiple priority layers and ARQ is modified regarding these layers. However, ARQ is decided as a result of an optimization process where optimum packet that minimizes delay distortion function is chosen for transmission. Optimization is performed over packets that are not transmitted before and transmitted but ACKs for them are not received.

Using only FEC or ARQ for unequal error protection is not sufficient for time varying and/or error prone channels. Therefore, it is common to use both FEC and ARQ in layered coding with transport prioritization. In most of the previous work, encoded data is divided into layers and unequal error protection for these layers is performed, and an ARQ scheme that may take into account the layers' importances is also used. For example, the work in [24] divides the total FEC bandwidth for different type of layers that varies depending on the contained information such as header, IP payload, shape motion, or texture, and performs priority based ARQ that decides re-transmission of packets according to their priority coming from layer information, and whether the packet is received before its deadline or not. Another unequal error protection scheme using both FEC and ARQ is proposed in [25]. This work uses similar unequal error protection technique as the technique in [24], but considers the channel condition to decide which FEC rate is proper for the current channel condition. One more improvement in this paper is to add dynamic packet length adjustment to the unequal error protection technique in order to improve the throughput of the channel.

Multiple Description Coding

As described in Section 2.2.1, layered coding can offer error resilience when the base layer is received successfully by using FEC and/or ARQ. In some applications, however, (re)transmitting without error a part of a signal with different FEC rate may not be guaranteed, or feasible. If this part is a base layer, quality of the decoded signal is reduced significantly. Therefore, another approach for transmitting layered data is to use multiple description coding, [15]. Multiple description coding supposes that there are multiple independent channels between transmitter and receiver. The channels may be physically distinct channels, or virtual channels that are defined by using time or frequency division.

Multiple description coding transmits coded packets over separate channels. Each packet stream in each channel is given a description, and each description contains information necessary for decoding. That means, all channels should carry a part of same information that reduces the throughput of the system.

Joint Source and Channel Coding

In layered and multiple description coding there is no effect of channel coder to the source coder. In other words, given the source information, channels coder uses different FEC, or retransmits corrupted packets. However, joint source channel coding methods propose that source coder can be modified depending on the channel coder, or channel conditions, [15]. Specifically, designing the quantizer and entropy coding in joint source and channel coding depends on the channel error characteristics to minimize the effect of transmission errors.

Robust Waveform - Entropy Coding

The objective of multimedia compression schemes is to remove temporal and spatial redundancy as much as possible and increase the compression gain. Removing redundancy makes the encoded data to be more vulnerable to channel errors, and makes the error concealments more difficult at the decoder. Therefore, robust waveform and entropy coding approaches are proposed. In these approaches, the idea is to add or keep some redundancy at the source coder, [15]. Robust waveform coding method keeps redundancy at waveform coding stage while redundancy is kept at enteropathy coding stage in robust entropy coding method.

Selective Encoding for Error Concealment

The multimedia encoders use predictive coding by removing temporal redundancy so that the size of encoded data is further reduced. The decoder uses prediction, hence corruption errors can propagate in the decoded data. In order to prevent this phenomenon selective encoding is used, [15]. Selective encoding stops predictive coding for a moment if the receiver locates the places of errors, and sends a request for interrupting predictive coding. This method reduces compression gain, but error propagation is reduced.

2.2.2.2 Transport Level Error Control and Concealment

After multimedia data is encoded, the output of the source coder is assembled into transport packets in such a way that when a packet is lost, the other packets can be still useful because the header and coding mode information is embedded into successive packets.

A packet often contains data from several blocks. To prevent the loss of contiguous blocks because of a single packet loss, interleaved packetization can be used, by which successive blocks are put into nonadjacent packets. This way, a packet loss will affect blocks in an interleaved order, which will ease the error concealment task at the decoder.

Packets vary according to their importances depending on the content of blocks that they consist of. Inter- and intra-coded frames includes several packets. There is a dependency structure between frames, hence there is an indirect dependency among these packets. There are many works for error control and concealment on the transport level that are discussed in the following paragraphs.

In transport layer error control and concealment scheme, FEC coding is performed at the transport layer using packet importances as in [26]. This work proposes an optimal FEC rate selection algorithm by using importances, deadlines, remaining times of H.263 encoded video packets, and the condition of the wireless channel. Using FIFO rule as a packet scheduling and not using ARQ are a disadvantages of this technique.

Some transport layer error control and concealment techniques use only ARQ for protection. A method in [27] is a ARQ decision method where packet transmission or retransmission decisions are made using perceptual and temporal importance of packets encoded with H.264 encoder. However, the effect of channel to quality is not considered in this paper as well as FEC rate selection which is important especially for wireless channels. An alternative method for ARQ is proposed in [28] where low delay interleaving scheme and conditional retransmission is based on concealment error and channel condition.

Packet interleaving can be performed independently from other error control and concealment techniques as in [29]. In this work, an interleaving scheme that minimizes distortion function subject to delay constraint is proposed.

Transport level packet scheduling techniques are presented in several works in the literature. One of these techniques proposes an optimal packet scheduling framework that minimizes total distortion function which is formulated by using channel statistics, transmission history, and channel feedback, [30]. Another technique, [31], schedules packets according to packet importance and deadline. One more technique, [32], performs scheduling by smoothing the video data in the transmit buffer. The idea is constructing importance levels for the packets belonging to the highest importance level, and transmitting them first. A further technique combines packet scheduling and ARQ in rate distortion optimized manner, [1]. The objective is to minimize the distortion function by using the packet importances, dependencies, and channel condition. To this point all the discussed techniques for error correction and concealment at transport level lack the use of robust error recovery methods such as FEC and ARQ.

There are several transport level error control and concealment methods that use both the advantages of FEC and ARQ. One method using both FEC and ARQ is to use lookup tables that contains channel condition and FEC pairs [33],[34]. These tables reconstructed before real bit stream begins. The idea is to compose the best FEC, channel state pair that guarantees the recovery of the packet under given channel condition and FEC pair. ARQ mechanism in these methods is similar to hybrid-I ARQ. Another approach is using hybrid FEC/ARQ scheme as in [19] and [35]. These methods proposes a hybrid FEC/ARQ scheme that combines the advantages of hybrid-I and hybrid-II ARQ schemes. In these works, certain error correction capability is provided in each (re)transmitted packet, and the information can be recovered from each transmission or retransmission alone if the errors are within error correction capability (type-I hybrid ARQ), and the retransmitted packet consists of redundancy bits which, when combined with the previously transmitted packet, result in a more powerful code to recover information if error correction has failed for every single packet (type-II hybrid ARQ). A disadvantage of this techniques is that they decide on (re)transmissions by not considering the packet importance or channel condition. A similar approach uses hybrid II ARQ scheme with packet scheduling by using rate distortion optimization, [2]. In this technique, information or parity packets are chosen for transmission by minimizing a distortion function.

2.3 Third Generation Mobile Communication Systems

Third generation mobile communication systems provide a global mobility with wide range of services including enhanced multimedia (voice, data, video), usability of all popular modes (cellular telephone, e-mail, paging, video conferencing, and web paging), and broad bandwidth and high speed (upto 2 Mbps). International Telecommunication Union (ITU) started the process defining the standard for third generation systems, referred to as International Mobile Telecommunications 2000 (IMT-2000). The Europe European Telecommunications Standard Institute (ETSI) was responsible of UMTS standardization process. Our main interest is on the UMTS systems, and more information about IMT-2000 can be found [36].

The Universal Mobile Telecommunications System (UMTS) is the third generation mobile communication system that allow the use of a new frequency spectrum. For UMTS, two schemes for radio access such as wideband code-division multiple access (W-CDMA), which adopts frequency division duplexing (FDD), and time division CDMA (TD-CDMA), which is based on time division duplexing (TDD) are proposed.

In general, the transmission part of UMTS system consists of binary information source, channel encoder (convolutional or Turbo coding) with puncturing or unequal repetition, interleaving, frame assembler for cyclic redundancy check (CRC) and tail bit insertion, multiplexing of dedicated control channel with dedicated channel, modulation, spreading and scrambling, and raised cosine filter for inter symbol interference reduction, [37].

The transmission process is performed as the following. The information coming from source is divided into transport blocks. Each of these blocks is added with Cyclic Redundancy Check (CRC). The size of CRC can be 24, 16, 12, 8, or 0 and it is learned from higher layers what CRC size should be used for each transport block. Then, according to the size of the code used for channel coding, transport blocks are concatenated or segmented, and code blocks are constructed. Then, code blocks are delivered to the channel coding block, and convolutional coding, turbo coding or no coding is applied to the code blocks. The encoded code blocks are padded with bits in order to ensure that the resultant data is segmented into same size blocks. Each transport block is interleaved, and segmented into fixed size blocks. After rate matching is also performed, transport blocks are combined by transport channel multiplexing. Finally, bit scrambling physical channel segmentation and physical channel mapping is performed.

In UMTS, users will be provided with data rates up to: 144 kbps, in macro-cellular environments, 384 kbps, in micro-cellular environments, and up to 2 Mbps in indoor or pico-cellular environments. However, the radio interface is characterized by great flexibility and a variety of different physical and logical channel types. In the downlink, there are different types of channels for data and control packets. These are dedicated channels (DCH) to transmit data packets, dedicated control channels (DCCH) to transmit pilot symbols, power control bit, downlink shared channels (DSCH) for carrying dedicated control or traffic data, forward access channel (FACH), random access channel for control information (RACH), paging channel (PCH), and broadcast channel to broadcast system and cell specific information (BCH), [38]. Depending on the design of the system one or more of these channel types may not be used or unrelated with the data transmission. For example, CRC

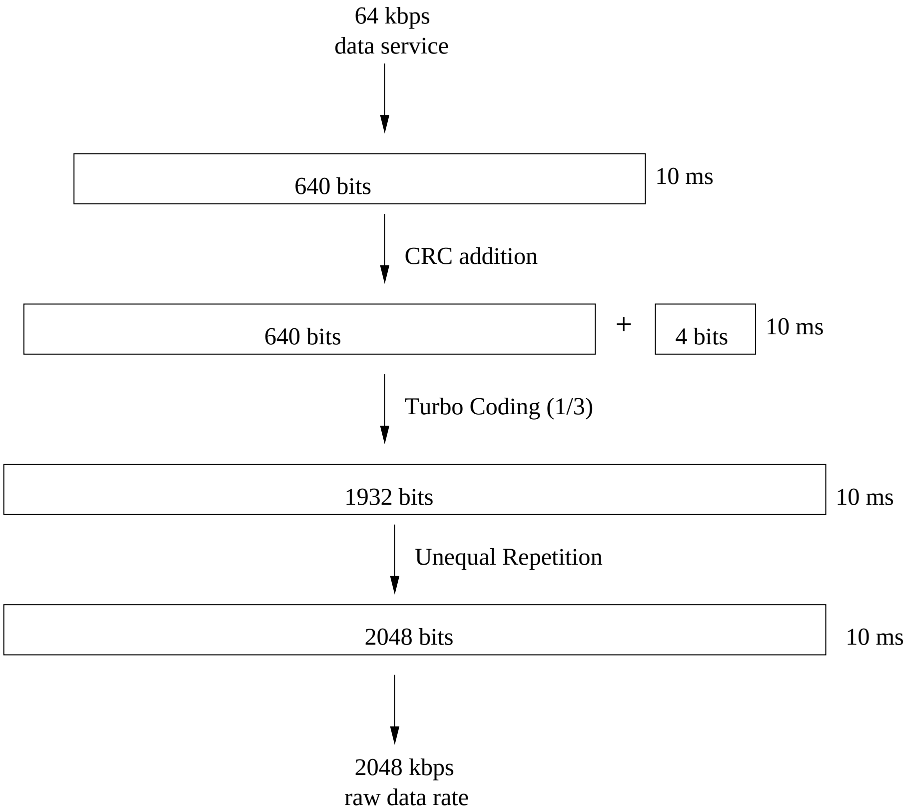

Figure 2.6: CRC and channel coding for 64 kbps service rate on the dedicated channel

addition and channel coding part of the transmission process for user data rate 64 kbps is shown in Fig. 2.6. As shown in Fig. 2.6, the raw data rate allocated for data transmission is 204.8 kbps. In this user data rate, there is a dedicated control channel in addition to data channel. The raw data rate allocated for dedicated control channel is 512 kbps, [37].

2.4 Wireless Channel Modeling

The propagation characteristics of wireless channels vary according to the type of the communication system. Terrestrial and cellular wireless communications can be given as systems having different propagation characteristics.

In terrestrial communication, propagation is point to point and only channel noise is added to the received signal. This type of communication systems are modeled with additive white gaussian noise (AWGN) channel model where zero-mean noise having a Gaussian distribution is added to the signal, [39]. On the other hand, cellular systems have multiple propagation paths due to the obstructing materials in the environment. Multipath propagation causes fading in the received signal. Fading is divided as slow fading and fast fading. Slow fading is the result of large reflectors and diffracting objects along the transmission path, [39]. However, fast fading is the rapid variation of signal levels when the receiver moves in short distances, [39].

In a typical multipath environment, the envelope of the received signal is represented with Rayleigh distribution where received instantaneous signal to noise ratio (SNR) γ is distributed exponentially with probability density function, [40];

$$p(\gamma) = \frac{1}{\gamma_0} exp(-\frac{\gamma}{\gamma_0}), \tag{2.4}$$

where γ_0 is the average SNR.

2.4.1 Gilbert-Elliott Channel Model

Rayleigh fading channel can be represented by using Markov models. A classical approach is two-state Gilbert-Elliott model, [41], [42]. In this model, the channel can be at one of two states at a time. One state shows noisy case (bad state), while the other shows near-noiseless case (good state). The probability of being in a state is called steady state probability, and moving from one state to the other is called transition probabilities. The objective of the Gilbert-Elliott model is to calculate steady state and transition probabilities depending on the movement speed of the mobile receiver and calculate bit-error rates occurring at each state.

2.4.2 Finite-State Markov Channel Model

Gilbert-Elliot model is a useful model to an extent, because it does not consider interim conditions of the channel. That means channel does not behave always as good or bad. Therefore, finite state Markov channel (FSMC) model is used for more exact representation of the channel. FSMC is considered in the works, [40], [43] where the general objective is the same as the Gilbert-Elliott channel model. In this thesis, FSMC model derived from the following formulations, [40], [43] is used.

The fading characteristics of the signal envelope are determined by the Doppler frequency due to the motion of a mobile terminal. The maximum Doppler frequency f_m is calculated as;

$$f_m = \frac{2v_m f_c}{c}, \tag{2.5}$$

where v_m is the speed of mobile, f_c is the communication frequency, and c is the velocity of propagation.

In order to find steady state and transition probabilities of the Markov model, level crossing rate of the instantaneous SNR process γ is calculated. It is the average number of times per unit interval that a fading signal crosses a given signal level Γ. The crossing rate $N(\Gamma)$ depends on the maximum Doppler frequency f_m, Γ, and average SNR of the channel γ_0;

$$N(\Gamma) = \sqrt{\frac{2\pi\Gamma}{\gamma_0}} f_m exp(-\frac{\Gamma}{\gamma_0}). \tag{2.6}$$

If the FSMC is composed of K states, there are $K+1$ level crossing rates. Initial level is $\Gamma_1 = 0$, and final level is $\Gamma_{K+1} = \infty$. The aim is to find the other crossing levels by solving the following equation that relates the duration of each state and the upper and lower crossing rates of the corresponding state;

$$\overline{\tau_k} = \frac{Prob\{\Gamma_k \leq \gamma \leq \Gamma_{k+1}\}}{N(\Gamma_k) + N(\Gamma_{k+1})}, \tag{2.7}$$

where $\overline{\tau_k}$ is the average time duration of the state k, and can be represented as $\overline{\tau_k} = c_k T_p$. In this relation c_k is a constant which should be larger than 1, and T_p is the average packet duration. If c_k constants are taken as the same for all k where $k = 1, 2, ..., K$, and if K is specified, the Eq.(2.7) returns to K equations with K unknowns. When this equation is solved, the results; $\Gamma_1, \Gamma_2, ...\Gamma_K, c_k$ are found. By using these results, steady state probabilities and state transition probabilities are as the following where π_k's are steady state probabilities, and $P_{k,k+1}$ and $P_{k,k-1}$ are transition probabilities;

$$\pi_k = exp(-\frac{\Gamma_k}{\gamma_0}) - exp(-\frac{\Gamma_{k+1}}{\gamma_0}), \tag{2.8}$$

$$P_{k,k+1} \approx \frac{N(\Gamma_{k+1})T_p}{\pi_k}, \; ... \; k = 1, 2, ..., K-1 \tag{2.9}$$

$$P_{k,k-1} \approx \frac{N(\Gamma_k)T_p}{\pi_k}, \ \ldots \ k = 2, 3, ..., K \tag{2.10}$$

The final information that should be defined is the bit error rate (BER) at each state. BER is calculated by using crossing rates and steady state probabilities. More information can be found in [43]. Now that all steady state, state transition probabilities, and BER values at each state is found, there is a complete wireless channel model at hand. This model can be used to asses the performance of some systems on simulation environment, and provide information for future behavior of the channel in real time simulations.

Chapter 3

System Model and Problem Formulation

In this thesis, we consider the wireless last hop of a generic network where a gateway streams multimedia data to a wireless client. As depicted in Fig. 3.1, the gateway receives a continuous stream of encoded real-time multimedia data from a media server via a high capacity and virtually error-free link. The data is buffered at the gateway until it is successfully transmitted to the user or its deadline expires. At each transmission opportunity from the gateway to the user, a transport algorithm is run over the packets available in the buffer. Transmission opportunities are given to the gateway when the transmission channel is idle and there is at least one packet in the transmit buffer. The packet selected as the outcome of the algorithm is encoded with an appropriate FEC code (also dictated by the algorithm) and transmitted via a bi-directional wireless link between the gateway and the user. Between the two transmission opportunities, gateway listens to the backward channel to receive the ACKnowledgements (ACKs) for the previously transmitted packets.

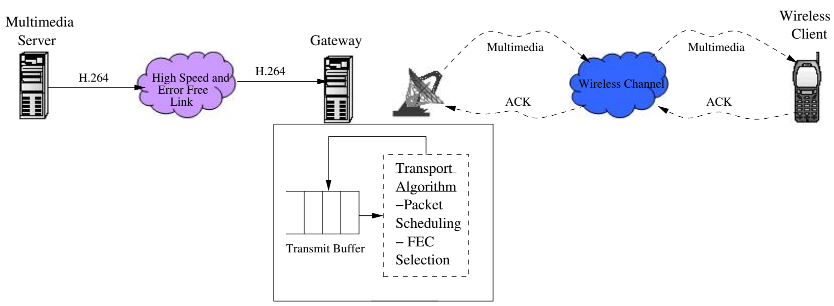

Figure 3.1: Multimedia streaming system

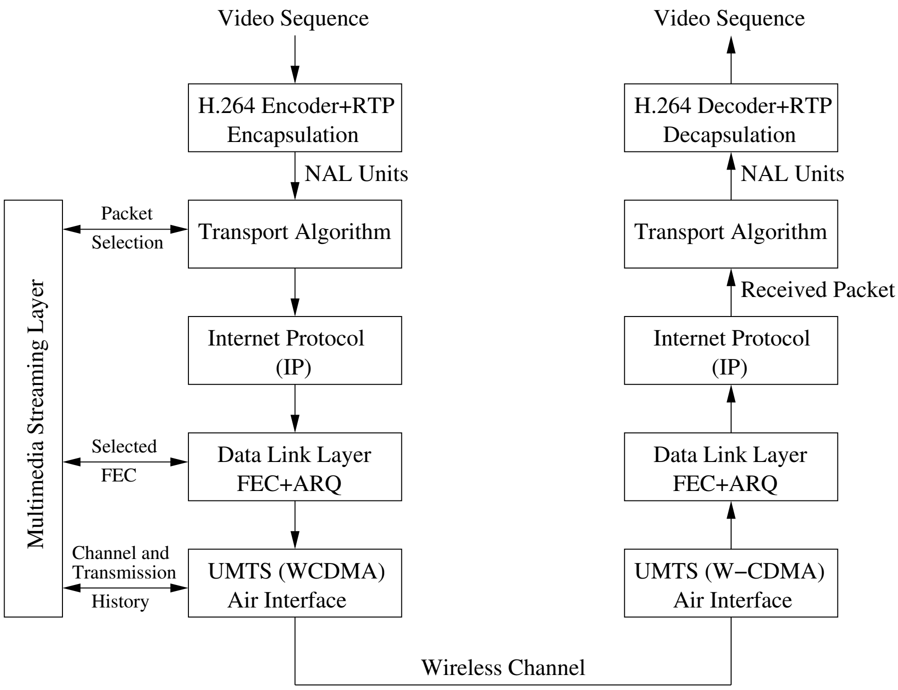

Figure 3.2: Protocol stack of the proposed system

In the following, we discuss the operational framework shown in Fig. 3.2 for our cross-layer algorithms in 3G mobile communication systems. We first discuss interactions among application, transport, datalink and physical layers on the gateway side. Second, we discuss the wireless channel model used. The wireless channel is modeled with finite-state Markov model. Finally, we mention the operational model for the wireless client.

3.1 Wireless Gateway

3.1.1 H.264 Encoder and RTP Encapsulation

The main application considered in this thesis is video streaming. It is assumed that the video sequence is encoded with an H.264 encoder, stored in a media server and when requested streamed to the wireless gateway. We chose H.264 as the video encoder, since it appears as a good choice for current and next generation wireless system applications with its high compression efficiency.

As mentioned in Section 2.1.2, H.264 encoder uses intra- and inter-coding depending on the trade off between the required Quality of Service (QoS) and the

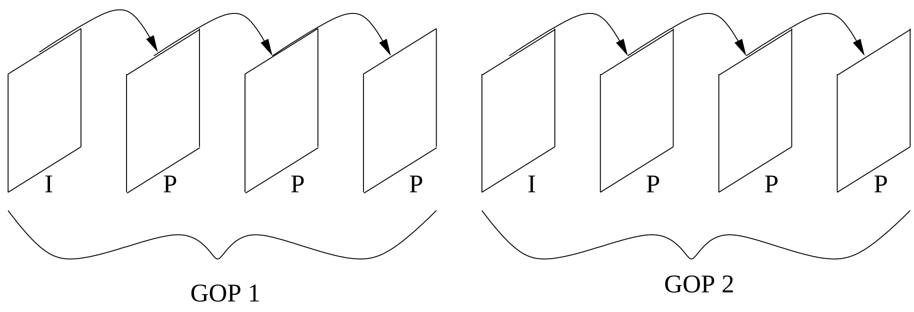

Figure 3.3: Dependency structure of H.264 encoded video

compression efficiency. In a general setting, H.264 encoder produces a sequence of Group of Pictures (GoPs) as the outcome of an input video stream. We assume that, each GoP consists of an intra-coded frame (I frame) and a certain number of successive frames called inter-coded frames (P frames). In a GoP, each P frame is dependent on the preceding P frame. The dependency structure of GoPs is depicted in Fig. 3.3. It is assumed that an error occurring in a GoP does not propagate to another GoP. However, an error that occurs in a frame in a GoP propagates in the worst case until the last frame in the GoP.

Frame coding and GoP fragmentation are performed by the video coding layer (VCL) of H.264 encoder. VCL also divides frames into smaller units called slices. The slices are composed of macroblocks, which contain the motion vectors and the transform coefficients representing the characteristics of the corresponding macroblocks. Slices also provide resynchronization points for the decoder to decode the received video bit-stream even when one or more slices are lost or corrupted. The VCL encoder passes slices to the network abstraction layer (NAL) of the H.264 encoder.

Network abstraction layer encapsulates each slice into a NAL unit. In our system, RTP encapsulation is employed since RTP encapsulation is more appropriate for multimedia streaming over wireless networks and it provides robust header compression.

Once NAL units are generated, they are transferred from the media server to the gateway via a high capacity and virtually error free link.

3.1.2 Transport and IP Layers

In the conventional protocol stack, transport layer is responsible for the reliable end-to-end transmission. In our system model, the transport layer is responsible for selecting the NAL unit to be transmitted in the next transmission opportunity according to the importance of the NAL unit (e.g., type of the frame the NAL unit belongs to), its position in the dependency structure, its size, decoding deadline, channel state, and capacity information. The importance of the NAL unit, its position in the dependency structure and the decoding deadline are given by the higher layers, whereas the channel state and capacity information is received by the physical layer.

Once a NAL unit is selected to be transmitted in the next transmission opportunity, a packet is created by adding RTP and Internet Protocol (IP) headers to the NAL unit. IP layer adds the necessary addressing information to the packet and passes the packet to the data link layer. In the current standards, the size of the RTP header is 16 bytes and the size of the IP header is 20 bytes. However, the header can be compressed by using Robust Header Compression (RoCH) protocol to 3 bytes [44].

3.1.3 Data Link and UMTS Layers

Data link layer is responsible to add redundancy according to an Error Correction Code (ECC). Once the packets are ECC encoded, they are sent over the W-CDMA air interface of UMTS.

UMTS views the incoming packets as a byte stream, and divides the byte stream into transport blocks. UMTS standard leaves it optional to use Cyclic Redundancy Check (CRC) and convolutional codes with rates 1/2 and 1/3 [38]. In this thesis, we do not use either CRC or convolutional codes. Instead, we use an ideal form of Reed-Solomon codes that can correct errors up to the half of the parity of the packet. Furthermore, the FEC rate is selected according to our algorithm discussed in the next section by taking into account the importance of the packet, its position in the dependency structure, decoding deadline, channel state, and channel capacity information.

In our simulations, we assume that the users subscribe to 64 kbps service rate. With our proposed algorithms, the selected FEC rate is often greater than the 1/3. Thus, the user can be served with data rate higher than 64 kbps which is guaranteed by UMTS. After FEC coding, radio frame equalization, bit interleaving, frame segmentation, rate matching, and bit scrambling are performed, and the data is mapped to the physical channel (wireless channel) and transmitted to the client.

3.1.4 Cross-Layer Multimedia Streaming

In this thesis, we modify the rate distortion optimization framework of proposed in [1] so that ARQ and FEC selection is optimized by minimizing end-to-end distortion that results when one or more packets are not received correctly by the client with subject to overall rate constraint. An optimal solution to multi-rate distortion optimization (MRDO) algorithm a packet and an FEC rate pair that minimizes the total distortion function, [45].

In this thesis, we also propose two low complexity multimedia streaming algorithms that can provide comparable performance with the optimal algorithm, [45]. The proposed transport algorithms are called reduced complexity (RC) algorithms that do not optimize end-to-end distortion. Instead, the most important parameters that are clarified by simulation results of MRDO algorithm are used to improve the quality. Specifically, first RC algorithm, Algorithm I, chooses the most suitable packet for transmission depending on packet importance and deadline. An FEC rate is selected for Algorithm I by using an FEC rate table that contains channel state-FEC rate pairs. This FEC rate table is constructed by using MRDO algorithm simulation results. The second RC algorithm, Algorithm II, selects most suitable packet and FEC rate jointly by minimizing a distortion metric that regards packet importance, deadline, as well as the effect of chosen FEC rate to reception probability of a packet before its deadline.

The proposed cross layer multimedia streaming algorithms are compared with a traditional FEC-ARQ algorithms such as fixed FEC, fixed FEC-ARQ, channel adaptive FEC, and channel adaptive FEC-ARQ. Fixed FEC scheme provides FEC rate capable of correcting occurs at the average BER of the channel. There is no packet scheduling or ARQ. The packet that entered the buffer first is transmitted, and removes from transmit buffer. Fixed FEC-ARQ scheme provides ARQ for corrupted packets in addition to FEC. Channel adaptive FEC is provides FEC rates that can correct transmission errors at the given channel state. Channel adaptive FEC-ARQ scheme also provides ARQ. The ARQ scheme used in fixed FEC-ARQ and channel adaptive FEC-ARQ is similar to type I hybrid ARQ scheme proposed in Section 2.2.1. However, in our scheme retransmissions are performed if an ACK is not received for a packet when some time is passed after transmission time instead of waiting negative acknowledgement.

3.2 Wireless Channel

In this thesis, we used cellular system structure, UMTS, for application area of the proposed transport algorithms. Therefore, it is possible to assume that the behavior of the channel is modeled with Rayleigh fading distribution. As mentioned in Section 2.4.2, Rayleigh fading channels can be modeled with finite state Markov channel (FSMC) models. For different channel conditions defined in Section 6.1.1, FSMC model is constructed and used in simulations.

3.3 Wireless Client

In the receiver part of the wireless client, W-CDMA air interface receives data stream from wireless channel, and extracts packets from this stream. Packets are passed to the data link layer. In the data link layer, the received packets are checked as to whether the packet is received correctly or not. If a packet is received correctly and before its deadline, the packet is sent to the transport protocol, and an ACK packet is generated as well as sent to the gateway over backward channel. If a packet is not received correctly or after deadline, the received packet is discarded, and no ACK is generated. In the air interface and data link layer, received packets are buffered before processing. The waiting time in these layers are modeled as an M/M/1 model. Therefore, the distribution of the waiting time is thought as exponential distribution. At the transport protocol, the packets are buffered and sequenced with respect to its frame number and the position in the frame. After

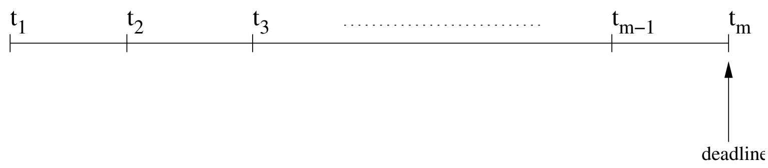

Figure 3.4: Transmission opportunities

play-out time of a frame is reached, the received packets composing the frame are sent to the H.264 decoder.

3.4 Problem Formulation

In this thesis, our aim is to select a packet from the transmit buffer in the gateway and choose an FEC rate in order to protect the selected packet. Assume that, there are L packets in the transmit buffer. One of these L packets and its FEC rate to be chosen at each transmission opportunity. Transmission opportunities occur when the gateway completes the transmission of a packet, and there are still packets waiting for transmission in the transmit buffer. Fig. 3.4 shows transmission opportunities where t_m is the last possible transmission opportunity for a packet since each packet has a deadline.

At each transmission opportunity, multimedia streaming algorithms choose a packet and and FEC rate considering the source and channel specific properties at that transmission opportunity. The source and channel specific properties for each packet k of frame f at transmission opportunity t_i may be the packet importance value α_k, transmission deadline $\theta_k(t_i)$, channel condition $\kappa(t_i)$, packet size β_k, and FEC rate $\upsilon_k(t_i)$. Considering these properties, packet and FEC rate are chosen to increase quality of received video. For example, for the case that a wireless channel is fading, transmission priority is given to important packets. Therefore, we should define a total distortion function that measures the possible distortion for all packets in the transmit buffer when a packet or FEC rate is chosen.

More formally, at each transmission opportunity, the packet k of frame f in the transmit buffer is assigned source and channel specific properties as shown in Eq.(3.1),

$$p_k(t_i) = [\alpha_k, \theta_k(t_i), \kappa(t_i), \beta_k, \upsilon_k(t_i)]. \tag{3.1}$$

Since there are L packets in the transmit buffer, we can define a vector in order to contain properties of all packets,

$$\boldsymbol{p}(t_i) = [p_1(t_i), p_2(t_i), ..., p_L(t_i)]. \tag{3.2}$$

As mentioned, our objective is to minimize a distortion function, $D(\boldsymbol{p}(t_i))$ that is defined by using the packet and channel properties in Eq.(3.1).In addition, the total rate function $R(\boldsymbol{p}(t_i))$ that depends on the chosen packet and FEC rate in current and future transmission opportunities should not be greater than the wireless channel capacity C. Therefore, our problem turns to the optimization problem in Eq.(3.3) and Eq.(3.4).

$$\min \quad D(\boldsymbol{p}(t_i)) \tag{3.3}$$

$$\text{s.t} \quad R(\boldsymbol{p}(t_i)) \leq C, \forall t_i \tag{3.4}$$

Chapter 4

Multi-Rate Distortion Optimization

In this chapter, our aim is to develop multi-rate distortion optimization framework that jointly optimizes packet selection and FEC using the framework proposed in [1] and [2]. More specifically, our objective is to optimally decide on the order and FEC rates of the packets sent in the impending transmission opportunities.

Assume that at a given time t, there are F_g frames from GoP g waiting for transmission in the transmit buffer of the gateway. Each frame f consists of N_f NAL units (as described in Section 2.1.2). In the following, we use the term "packet" to describe a NAL unit.

The transmission policy defines the packet transmitted at time instant t, and the FEC rate used in the transmission. Let $c_n(t)$ denote the FEC rate used for transmitting packet n at time t. For determining the best transmission policy for which rate-distortion is optimized, we consider a one-step look ahead strategy. For each packet n in the transmit buffer, we define a *rate vector* $\pi_{f,n}(t) = [c_n(t), c_n(t')]$, which shows the FEC rate of the packet n of frame f at the current time t, and at the subsequent transmission time t'. $c_n(t)$ is an element of a predefined set $C = \{\eta_0, \eta_1, ..., \eta_M\}$, where $c_n(t) = \eta_0$ represents the case when the packet n is **not** transmitted at time t. Let $\boldsymbol{\pi}(t)$ be the rate vector over all packets n and all frames f awaiting for transmission in the buffer. Note that each packet n represents a NAL unit in a frame f, where f is a frame in a particular GoP g.

Let $D(\boldsymbol{\pi}(t))$ be the total distortion observed with rate vector $\boldsymbol{\pi}(t)$ for all packets.

Distortion occurs due to the loss or delay of the packets in the wireless channel. Increasing the FEC rate reduces the loss of packets, but increases the cost of transmission. The rate vector $\boldsymbol{\pi}(t)$ is associated with a cost $R(\boldsymbol{\pi}(t))$. Our objective is to determine the optimal rate vector $\boldsymbol{\pi}(t)$ that minimizes the total distortion subject to a channel capacity constraint. We obtain the optimal solution by deriving the first-order conditions for the Langrangian function $J(\boldsymbol{\pi}(t), \lambda)$:

$$J(\boldsymbol{\pi}(t), \lambda) = D(\boldsymbol{\pi}(t)) + \lambda R(\boldsymbol{\pi}(t)). \tag{4.1}$$

In line with the discussion in [1], we define the distortion under the rate vector $\boldsymbol{\pi}(t)$ as the difference between the maximum distortion, $D_0(t)$, when none of the packets are received correctly or within their deadlines, and the total improvement delivered by the packets delivered correctly with rate vector $\boldsymbol{\pi}(t)$. Defined formally,

$$D(\boldsymbol{\pi}(t)) = D_0(t) - \sum_{g=1}^{G} \sum_{f=1}^{F_g} \Delta D_f \prod_{i=1}^{f} \left(1 - \frac{\sum_{n=1}^{N_i} \triangle D_{i,n} \varepsilon(\pi_{i,n}(t))}{\sum_{n=1}^{N_i} \triangle D_{i,n}} \right), \tag{4.2}$$

where $\triangle D_f$ is the relative importance of frame f measured as the reduction in SNR when this frame is received correctly and on time, $\triangle D_{i,n}$ is the relative importance of packet n in frame i defined in a similar fashion, and $\varepsilon(\pi_{i,n}(t))$ is the probability of failure of packet n of frame i(due to channel errors and delay) under the rate vector $\pi_{i,n}(t)$. Frame f is said to be successfully decoded if f is received correctly and on-time and all frames in the GoP that frame f depends on are correctly decoded. Thus, the product in Eq. (4.2) represents the probability that all frames that appear before frame f in dependency chain are correctly received.

Incidentally, the correct reception of each frame f depends on the correct reception of its N_f packets. However, frame f can still be decoded even one or more packets are not received successfully. Therefore, we define the probability of correct reception of frame f as the weighted average of the probabilities of correct reception of the packets in frame f as given in Eq.(4.2).

In Eq.(4.2), the failure probability, $\varepsilon(\pi_{i,n}(t))$, of packet n of frame i is calculated according to the success of previous transmissions. The transmission of packet n is said to be successful if at least one of the re-transmissions of packet n is received correctly by the client and also ACK is correctly received by the gateway before the deadline of the packet expires. Thus, $\varepsilon(\pi_{i,n}(t))$ depends on the forward trip time (FTT) and round trip time (RTT) of the wireless channel, $p_n(t)$ is the number of previous transmissions of packet n until time t, t_j is the time of j^{th} transmission of packet n, and t_{DTS} is the deadline of the packet. FTT is the difference between the time the packet is transmitted and the time the packet is received by the data link layer of gateway. Similarly, RTT is the difference between the time packet is transmitted by the gateway and the time ACK is received back at the gateway.

If packet n is transmitted at the current time t with rate $c_n(t) \neq \eta_0$ but not at the next transmission time instant t', i.e., $c_n(t') = \eta_0$, the probability of failure of packet n is given by:

$$\varepsilon(\pi_{i,n}(t)) = \prod_{j \leq p_n(t),\, c_n(t_j) \neq \eta_0} P\{FTT > t_{DTS} - t_j | RTT > t - t_j\}. \tag{4.3}$$

Thus, $\varepsilon(\pi_{i,n}(t))$ in Eq.(4.3) is equal to the probability that no ACK from previous transmissions of packet n is received until the deadline of the packet expires. Note that this event is equivalent to having a FTT *for each transmission* exceeding the deadline of the packet given that no ACK is received within the deadline.

In a similar fashion, if packet n is transmitted in the next transmission instant t' with rate $c_n(t') \in C - \{\eta_0\}$, then $\varepsilon(\pi_{i,n}(t))$ is defined as follows:

$$\varepsilon(\pi_{i,n}(t)) = \prod_{j \leq p_n(t),\, c_n(t_j) \neq \eta_0} P\{FTT > t_{DTS} - t'\} P\{FTT > t_{DTS} - t_j | RTT > t - t_j\}. \tag{4.4}$$

In this case, we say that the transmission of packet n fails when none of the transmissions until current time t is received correctly (given that there is no ACK received by the packet deadline), and also the transmission in the next transmission time cannot be received correctly.

In order to calculate the failure probabilities in Eq.(4.3) and Eq.(4.4), we need to determine the probability distribution functions of forward trip time (FTT) and round trip time (RTT). FTT depends on the packet size, FEC rate, waiting time in the air interface, propagation delay and the processing time in the data link layers. Let $p_F(t)$ be the probability density function describing the delay observed in the forward channel. In the literature, $p_F(t)$ is usually modeled as exponential distribution [1]. Moreover, RTT unlike FTT, also depends on the size of the ACK and the delays occurring in the backward channel. Thus, the channel delay in RTT is the sum of delays in the forward and backward channels. Since delay in forward channel, $p_F(t)$, is exponentially distributed random variable, the channel delay in RTT can be modeled by second degree gamma distribution. Then, probability of dropping a packet at the wireless client due to the packet delay or corruption can be calculated as follows:

$$P\{FTT' > \tau\} = \int_{\tau}^{\infty} p_F(t)dt + \left(1 - \int_{\tau}^{\infty} p_F(t)dt\right) \varepsilon_F(s,r), \tag{4.5}$$

where τ is the remaining time for the packet, and $\varepsilon_R(s,r)$ is the probability of packet corruption when channel is in state s and FEC rate is r. For Reed-Solomon codes that can correct errors less than the half of the parity, $\varepsilon_R(s,r)$ is calculated as:

$$\epsilon_F(s,r) = 1 - \sum_{k=0}^{\lceil B_{i,n}(r-1)/2 \rceil} \begin{pmatrix} B_{i,n}r \\ k \end{pmatrix} (\delta_s)^k (1-\delta_s)^{(B_{i,n}r-k)}, \tag{4.6}$$

where $B_{i,n}$ is the size of the nth packet of ith frame, and δ_s is the symbol error rate observed when the channel is at state s.

Similarly, ACK for a transmitted packet is said to fail, if ACK is not received until the current time or ACK is received with uncorrectable errors occurring in the backward channel. Then, the probability of ignoring an ACK packet in the wireless gateway due to delay or corruption of the ACK is calculated as follows:

$$P\{RTT' > \tau\} = \int_{\tau}^{\infty} p_R(t)dt + \left(1 - \int_{\tau}^{\infty} p_R(t)dt\right) \varepsilon_R(s,r), \tag{4.7}$$

where τ is the time difference between the current time and the transmission time of related packet, $p_R(t)$ is the probability density function of second degree gamma distribution, and $\varepsilon_R(s,r)$ is the probability of corruption in the round-trip channel. A corruption occurs in the round-trip channel, when either the packet or ACK is corrupted in the forward or backward channel, respectively. The probability of corruption in the round-trip channel, $\varepsilon_R(s,r)$ is defined as:

$$\varepsilon_R(s,r) = 1 - (1-\varepsilon_F(s,r))(1-\varepsilon_B(s',r)),$$

where $\varepsilon_B(s',r)$ is the corruption of ACK packet in the backward channel when the backward channel is in state s' and the same FEC rate r is used to encode ACK. $\varepsilon_B(s', r)$ is also calculated using Eq.(4.6).

Above, we defined all the quantities necessary to calculate the distortion function given in Eq. (4.2). Now, we define the total transmission cost function $R(\boldsymbol{\pi}(t))$. $R(\boldsymbol{\pi}(t))$ is defined as the sum of the estimated size of all packets at current and next transmission times. More formally, $R(\boldsymbol{\pi}(t))$ is given as:

$$R(\boldsymbol{\pi}(t)) = \sum_{g=1}^{G} \sum_{i=1}^{F_g} \sum_{n=1}^{N_i} B_{i,n} \rho(\pi_{i,n}(t)) \tag{4.8}$$

where G is the number of GoPs in the buffer, F_g is the number of frames in GoP f, N_i is the number of packets in frame i, and $\rho(\pi_{i,n}(t))$ is the average code rate used in transmitting packet n.

$$\rho(\pi_{i,n}(t)) = c_n(t) + c_n(t')(1 - P_{ACK,n}(t)), \tag{4.9}$$

where $P_{ACK,n}(t)$ is the probability of successfully receiving an ACK before the next transmission instant t'. Note that ACK may be received for any of the packet transmissions occurring on and before time t

Chapter 5

Reduced Complexity Algorithms

Multi-rate distortion optimization algorithm presented in Chapter 4 provides optimal scheduling decisions along with optimal FEC rates for encoded video streams. However, due to its complexity, MRDO algorithm is too time consuming and therefore not appropriate for implementation in real-time streaming applications. In this chapter, we present two new algorithms with reduced complexity while offering similar SNR performance with MRDO. In both algorithms, reduced complexity metrics for assessing distortion are devised by considering the system parameters that affect the SNR performance of MRDO through long-term simulations. Such parameters have been identified as packet importance, packet remaining time, and the state of the wireless channel. Packet importance is the measure of the decrease in SNR value when a packet is not received correctly or in time, packet remaining time is the allowed time until the packet's delivery deadline, and wireless channel state is assumed to be estimated at the sender or feedback by the receiver.

5.1 Algorithm I

In Algorithm I, we consider packet selection and FEC rate assignment separately: First a packet is selected from the buffer considering packet importance and remaining times, then the best FEC rate is assigned according to the channel state using a table that contains FEC rate-channel state pairs.

Simulations with the MRDO algorithm have shown that the impact of packet importance on SNR improvement is higher than the effect of packet remaining time. To reflect this property, we propose the following scheduling metric, d_I, for packet selection:

$$d_I = c_1^{\triangle D_{f,n}}(t_{DTS} - t) \tag{5.1}$$

where $\triangle D_{f,n}$ is the importance of packet n of frame$f$, t_{DTS} is the packet's deadline, t is the current time, and c_1 is a positive constant chosen as $0 < c_1 < 1$.

Fig. 5.1 depicts the flow diagram of Algorithm I. The metric d_I is computed for all of the packets in the transmit buffer and the packet that minimizes this metric is selected for transmission. Next, the FEC rate for the selected packet is determined from the FEC rate table for the current channel state. If a given state is experienced for the first time, i.e. the given state does not exist in the FEC rate table MRDO algorithms is employed. FEC rate, channel state pair is recorded and Algorithm I is employed in the next transmission opportunity. After a packet's transmission, the packet is placed in a wait buffer, and a timer is started while waiting for the acknowledgement and the packet's metric is not computed until its timer expires. If the packet is not delivered successfully, it will again be passed to the transmit buffer and considered in the optimization as a candidate packet as if it was not transmitted before.

The FEC rate table is constructed from MRDO decisions obtained in different channel states. Our experiments with long term simulations have indicated that the variance of the probability of selecting an FEC rate at a given state is small. Therefore, instead of long term measurements, short term decisions can be sufficient. Therefore, in the learning stage, the FEC rate table is constructed during actual packet transmissions with MRDO and the transmitter switches between MRDO and Algorithm I as shown in Fig. 5.1.

5.2 Algorithm II

In Algorithm II, we propose packet and FEC rate selection to be performed jointly with a simplified distortion metric that considers the impact of packet importance on video SNR together with the probability of discarding a packet (which could be due to channel errors or delay). The new distortion metric, d_{II}, is defined in the following:

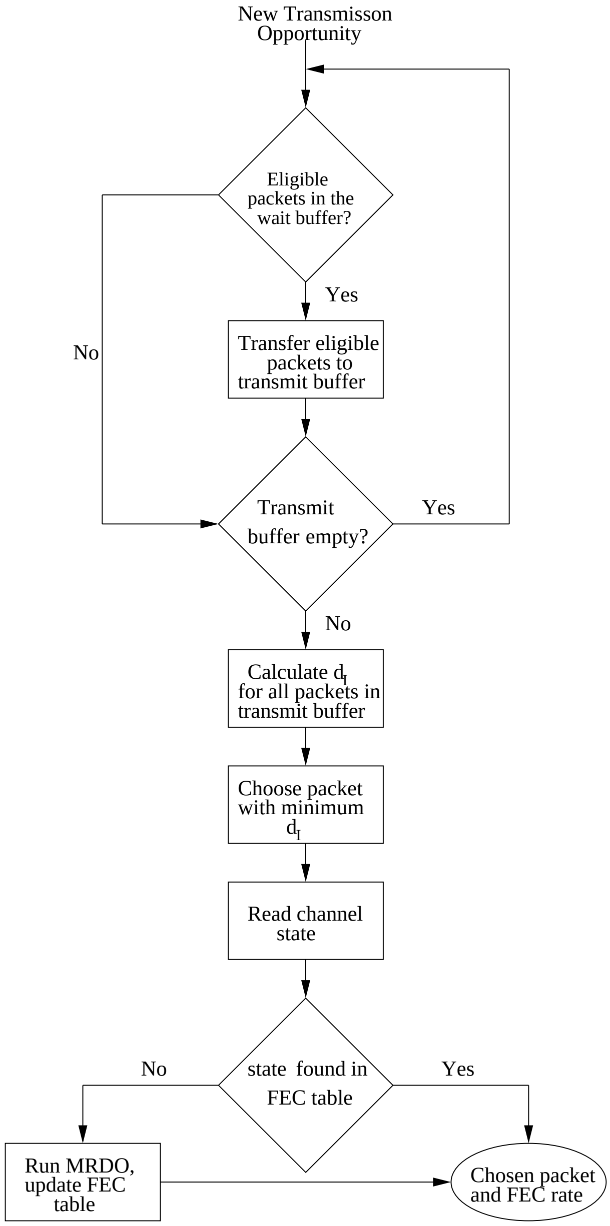

Figure 5.1: Flow diagram of Algorithm I

$$d_{II} = \epsilon(\pi_{f,n}(t))c_2^{\triangle D_{f,n}} \tag{5.2}$$

where $\triangle D_{f,n}$ is the importance of the packet n of frame f, c_2 is a constant in the interval $0 < c_2 < 1$, and $\epsilon(\pi_{f,n}(t))$ is the probability of discarding the packet n of frame f as defined in Eq.(4.3). $\epsilon(\pi_{f,n}(t))$ includes the all transmissions of the packet before its deadline expires. To further simplify the distortion metric, previous transmissions can be neglected and the reduced distortion metric obtained as:

$$d_{II} = P\{FTT' > t_{DTS} - t\}c^{\triangle D_{f,n}} \tag{5.3}$$

The reduced distortion metric accounts for the distortion when the estimated forward trip time exceeds the deadline for packet n as well as its importance. The first probability term is calculated as in Eq.(5.3) with the mean value of the exponential probability density function varied according to the selected FEC rate, considering the changes in packet transmission time with different FEC rates. Hence, delay effect of a given FEC rate is incorporated into the reduced distortion metric. Eventually, the packet and its FEC rate that minimizes the distortion metric are determined before transmission. A flow chart for Algorithm II is given in Fig. 5.2.

After the selected packet is transmitted, the packet is buffered in a sent buffer, and a timeout is started to wait for the acknowledgement. Before its timer expires, this packet is not included in the optimization. When the timeout expires, the packet passed to transmit buffer, and again becomes a candidate for the transmission.

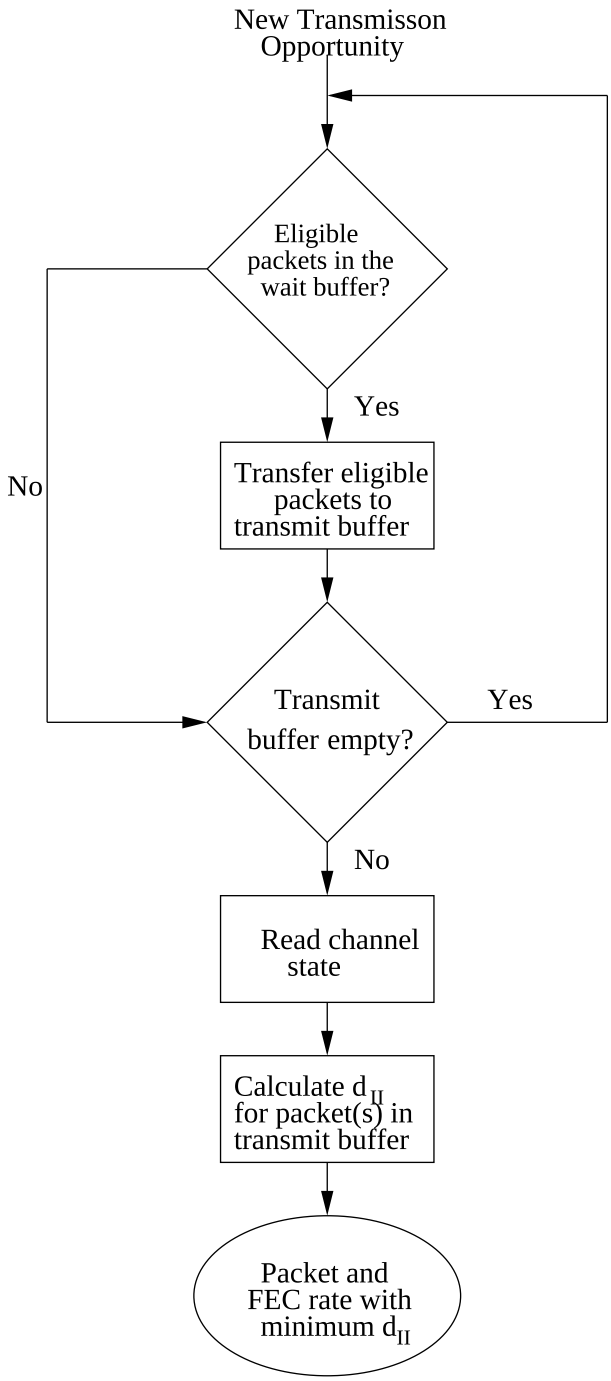

Figure 5.2: Flow diagram of Algorithm II

Chapter 6

Performance Analysis

In this chapter, we present the simulation results to evaluate the performance of the proposed multi-rate distortion optimization and the reduced complexity algorithms. The proposed transport algorithms are compared with traditional error control schemes such as fixed FEC, fixed FEC-ARQ, channel adaptive FEC and channel adaptive FEC-ARQ mentioned in Section 3.1.4.

6.1 Simulation Models

In our simulations we have modeled the system shown in Fig. 3.2, including video application and transport layers together with the parameters of UMTS, data link, physical layers as well as the wireless channel.

6.1.1 H.264 Encoder

For the application, JM.8.3 H.264 encoder and decoder distributed in [46] are used to compress video sequences, decode transmitted data, and measure the quality of the received video sequence through video SNR. In simulations, H.264 encoded QCIF and CIF size video sequences [47] mentioned in Section 6.2.1 and 6.2.2 were used. The video data rate is set to 70 kbps and 140 kbps for QCIF and CIF sequences respectively. The frame rate has been set to 10 frames per second and GoP size is selected as 10 frames (1 I frame and 9 P frames). Each frame is divided into slices that contain macroblocks and carry a totals average of 250 bytes of video data. In the network abstraction layer, the slices are encapsulated into NAL units where they become available as candidate packets for the transport algorithm.

6.1.2 Multimedia Streaming

The transport algorithm in multimedia streaming layer determines which packet should be selected from the transmit buffer, and which FEC rate will be used to protect this packet. This decision is made considering packet importance, packet dependencies, packet size, decoding deadline, channel state information, and channel capacity using the proposed cross-layer schemes, namely MRDO and RC algorithms. These algorithms are compared with traditional error control schemes such as fixed FEC, fixed FEC-ARQ, channel adaptive FEC and channel adaptive FEC-ARQ.

For reduced complexity algorithms, distortion metrics defined in Eq.(5.1), and Eq.(5.3) requires the definition of constants c_1 and c_2, respectively. From sensitivity analysis shown in Section 6.3, we have see that taking these constants as $c_1 = 0.9$ and $c_2 = 0.9$ results maximum performance improvement.

For fixed-FEC algorithm, it is required to use an FEC rate that can correct errors occur at average BER of the channel. For 5 dB channel FEC rate is chosen as 0.475, but this FEC rate cannot be supported by the limited capacity of the channel, so FEC rate is taken as 0.7 and for 10 dB channel FEC rate is 0.8225. These values are also used for fixed FEC-ARQ system. For channel adaptive FEC, the FEC rates that can correct average BER shown in Table 6.3 are in Table 6.1 for 5 dB and 10 dB channels. These values are also used for channel adaptive FEC-ARQ scheme.

The decoding deadline for every frame is 150 ms after the time the frame is generated. All the packets in the same frame have the same decoding deadline. FEC rates are selected from the set, $C_s = \{0, 1, 0.98, 0.94, 0.91, 0.83, 0.71, 0.625, 0.55, 0.5, 0.33\}$ where the ratio of un-coded to coded bits is indicated, with 0 being the decision of not sending packet. The size of the selected packet is incremented by 3 bytes to account for compressed RTP and IP headers, and then updated according to the selected FEC rate at the data link layer before the packet is passed to UMTS W-CDMA air interface.

Table 6.1: FEC rates for channel adaptive FEC (Average SNR = 5 dB, 10 dB)

	5 dB	10 dB
state1	no FEC	0.3335
state2	no FEC	0.4616
state3	0.3929	0.8865
state4	0.8901	0.9426
state5	0.9502	0.9433
state6	0.9433	0.9433
state7	0.9433	0.9433
state8	0.9433	0.9433
state9	0.9433	0.9433
state10	0.9433	0.9433
state11	0.9433	0.9433
state12	0.9433	0.9433

6.1.3 UMTS - Air Interface

At the W-CDMA air interface, the dedicated channel at raw rate 204.8 kbps is considered, which also defines the channel capacity for rate distortion optimization formulation. For this dedicated W-CDMA channel, the frame size is 10 ms, which contains 256 bytes. If the size of the coded packet exceeds 256 bytes, the remaining part of the packet is sent in the next UMTS frame; if the packet size is less than 256 bytes, the rest of the frame is used for the next selected and FEC coded packet. UMTS air interface is considered as a point-to-point link, which is a reasonable assumption since the users are assigned dedicated channels with different spreading codes, and the UMTS frame size can be used in a flexible manner.

6.1.4 Wireless Channel

As mentioned, we model the wireless channel as Rayleigh fading channel. The analytical model of the channel follows the finite state Markov model proposed in [40]. In this model, the finite state Markov model partitions the signal-to-noise ratio (SNR) into a finite number of states that is previously defined depending on the required accuracy of the channel model. The duration of each state and the crossing SNR levels between adjacent states are determined by solving a set of equation in [40] with the use of the information regarding the number of states, the maximum Doppler frequency, average packet time period, and average SNR value of the channel. After calculating the duration of each state and the SNR crossing levels of the states, transition probabilities, steady state probabilities, and bit-error-rate (BER) at each state are easily calculated. In summary, the received SNR is mapped to a state which corresponds to a different channel quality indicated by bit error-rate. Therefore, in a specific time, if we know SNR, we know the state of the channel and the average BER value. Several other methodologies can be used to map received SNR level to a specified state [41].

In our simulation framework, we employed the model in [40] assuming that the SNR level of the forward channel is available and monitored through the physical layer . Assuming a speed of 3 km/h for the wireless client, we obtained the 12-state Markov model of the channel for two channels, one with average SNR of 5 dB and the other with 10 dB. The BER and SNR levels of each state, steady state, and state transition probabilities were determined in the Markov model. For the two SNR levels, the same state transition probabilities have been obtained as shown in Fig. 6.1 and steady state probabilities are given in Table 6.2, but BER levels at each state are different as shown in Table 6.3. For the considered user speed and frequency, the coherence time of the channel is 21 ms, which is the average duration of stay in a state. This results in slow fading within a transmission opportunity, and the channel can be assumed to be static over a packet's transmission time considering all FEC rates.

Packet delay is the sum of packet's transmission time over the channel, waiting times in the air interface and the data link layer of the receiver. The forward and round trip times (FTT and RTT) has been modeled by the exponential distribution due to exponential waiting times, mean values computed according to transmission delay. Average FTT is calculated by adding mean values of transmission time of a packet (250 bytes on the average), selected FEC rate (depends on the channel), waiting time (2 ms) in the receiver. Average RTT is calculated in the same manner

Table 6.2: Steady state probabilities

	steady state probabilities
state1	0.0134
state2	0.0733
state3	0.1761
state4	0.2539
state5	0.2245
state6	0.1481
state7	0.0733
state8	0.0270
state9	0.0076
state10	0.0022
state11	0.0004
state12	0.0001

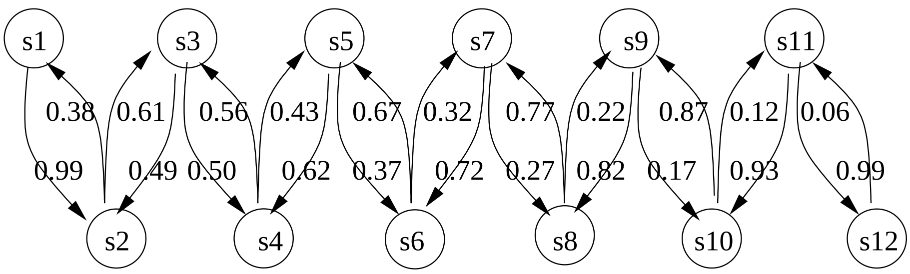

Figure 6.1: State transition diagram for wireless channel with (Average SNR = 5 dB, 10 dB)

Table 6.3: BER for wireless channels (Average SNR = 5 dB, 10 dB)

	5 dB	10 dB
state1	0.75836	0.57332
state2	0.25312	0.095787
state3	0.043367	0.004315
state4	$3.7686 * 10^{-3}$	$4.8503 * 10^{-5}$
state5	$1.562 * 10^{-4}$	$1.1875 * 10^{-7}$
state6	$2.8706 * 10^{-6}$	$5.4699 * 10^{-11}$
state7	$2.1341 * 10^{-8}$	$3.9684 * 10^{-15}$
state8	$5.6696 * 10^{-11}$	$3.5633 * 10^{-20}$
state9	$4.4899 * 10^{-14}$	$2.7743 * 10^{-26}$
state10	$7.9168 * 10^{-18}$	$1.0521 * 10^{-33}$
state11	$1.801 * 10^{-22}$	$6.5739 * 10^{-43}$
state12	$1.3386 * 10^{-28}$	$4.3903 * 10^{-55}$

as average FTT. The transmission time of an ACK packet on the backward channel and waiting time for ACK packet in the gateway should be added. The size of ACK packets is 16 bytes; the waiting time in the gateway is also assumed to be 2 ms.

6.2 Simulation Results

6.2.1 Experiments with QCIF Video Sequences

In this section, we present the performance of the proposed algorithms with QCIF size video sequences. The video rate is set to average 70 kbps so that at most 100 kbps of the total channel capacity will be used. Streaming of three sequences with different video characteristics, namely, *Foreman*, *Carphone*, and *Mother and Daughter* sequences were simulated using the proposed transport algorithms (MRDO, Algorithm I, and Algorithm II) and traditional error control schemes such as fixed FEC, fixed FEC-ARQ, channel adaptive FEC and channel adaptive FEC-ARQ . In order to compare the perceptual performance of each algorithm PSNR measurement

Table 6.4: PSNR in 5 dB channel for QCIF sequences

	Foreman (dB)	Carphone (dB)	Mother and Daughter (dB)
MRDO	26.56	27.04	33.86
Algorithm I	25.93	24.81	31.26
Algorithm II	25.26	24.45	30.36
channel adaptive FEC-ARQ	24.18	23.56	26.31
fixed FEC	23.24	22.91	23.91
fixed FEC-ARQ	22.36	21.52	24.07
channel adaptive FEC	21.59	21.45	22.14

is used. The recorded results for 5 dB and 10 dB channels are shown in Table 6.4 and 6.5, respectively.

As it can be seen in Table 6.4 and 6.5, the MRDO algorithm outperforms the channel adaptive FEC-ARQ by approximately 2-6 dB, which is a significant achievement range. The lower complexity algorithms, Algorithm I and Algorithm II perform fairly close to MRDO algorithm, outperforming channel adaptive FEC-ARQ by 1-5 dB, still proving major performance advantage with much lower complexity. The amount of SNR improvement by the proposed algorithms varies for different video sequences.

The reason for this difference is due to the rate of change in video scenes. For a fast moving video, the dependency between consecutive frames is small which reduces the SNR enhancement of the schemes that emphasize packet dependency.

Table 6.5: PSNR in 10 dB channel for QCIF sequences

	Foreman (dB)	Carphone (dB)	Mother and Daughter (dB)
MRDO	28.17	29.22	39.56
Algorithm I	27.83	28.63	38.54
Algorithm II	27.83	28.52	38.66
channel adaptive FEC-ARQ	26.77	27.48	33.94
fixed FEC	26.29	27.12	32.22
fixed FEC-ARQ	26.65	27.42	34.04
channel adaptive FEC	25.45	26.67	32.51

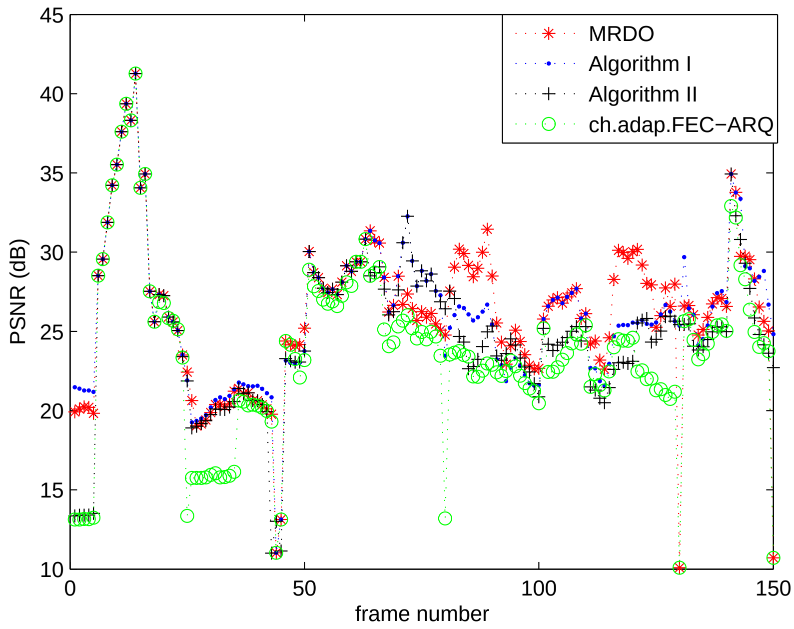

Figure 6.2: PSNR comparison of QCIF size *Foreman* sequence in 5 dB channel

Video SNR is improved more significantly as the sequence becomes more stable as in the *Mother and Daughter* sequence. The tables also indicate that the quality improvement by the proposed algorithms over channel adaptive FEC-ARQ is higher in the 5 dB channel create more room for improvement since the key principle of proposed algorithms is to reduce frame loss probability. In 10 dB channel however, frame loss probability is lower, hence its difficult to see high performance difference between proposed algorithms and channel adaptive FEC-ARQ algorithm, unless for the slow sequences as in *Mother and Daughter*.

In Fig. 6.2, 6.3, 6.4, 6.5, 6.6, and 6.7, the variation of peak SNR (PSNR) with time is depicted for the received *Foreman*, *Carphone*, and *Mother and Daughter* sequences for 5 dB and 10 dB channels, respectively. It can be seen in all figures that SNR decreases for some specific time intervals in all algorithms. While our proposed algorithms experience a smooth SNR decrease, traditional schemes show deep fades. This is due to the fact that some important or urgent packets are lost by the these algorithms during channel fades while these packets are specially treated and scheduled by the proposed algorithms, and the probability of loss is much lower in these algorithms.

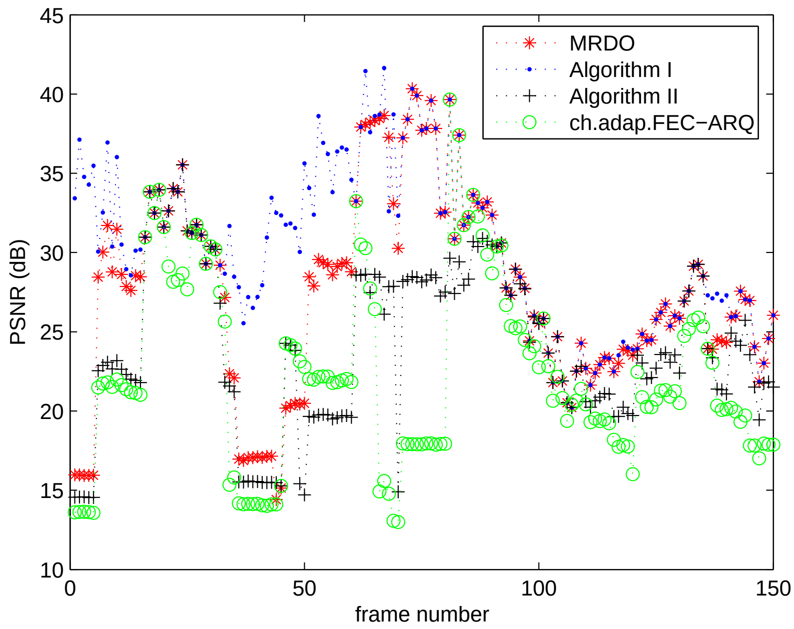

Figure 6.3: PSNR comparison of QCIF size *Carphone* sequence in 5 dB channel

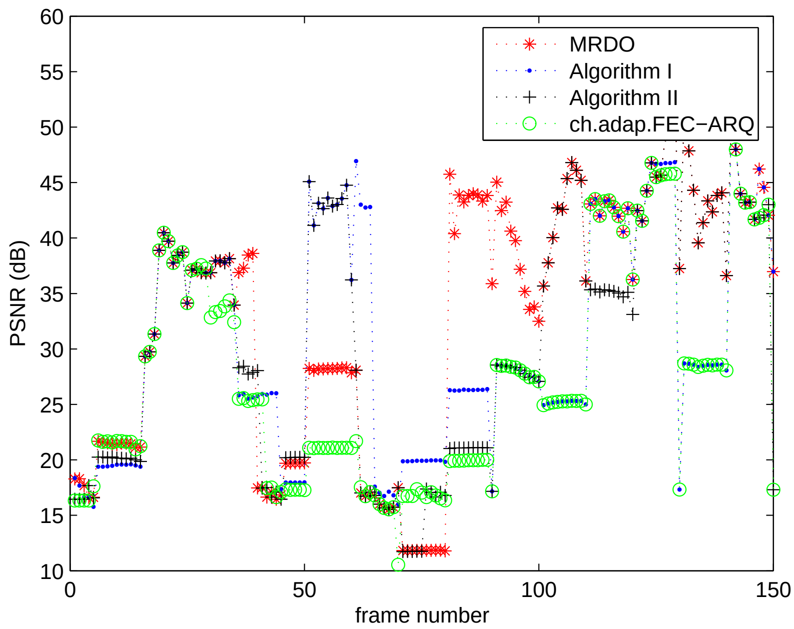

Figure 6.4: PSNR comparison of QCIF size *Mother and Daughter* sequence in 5 dB channel

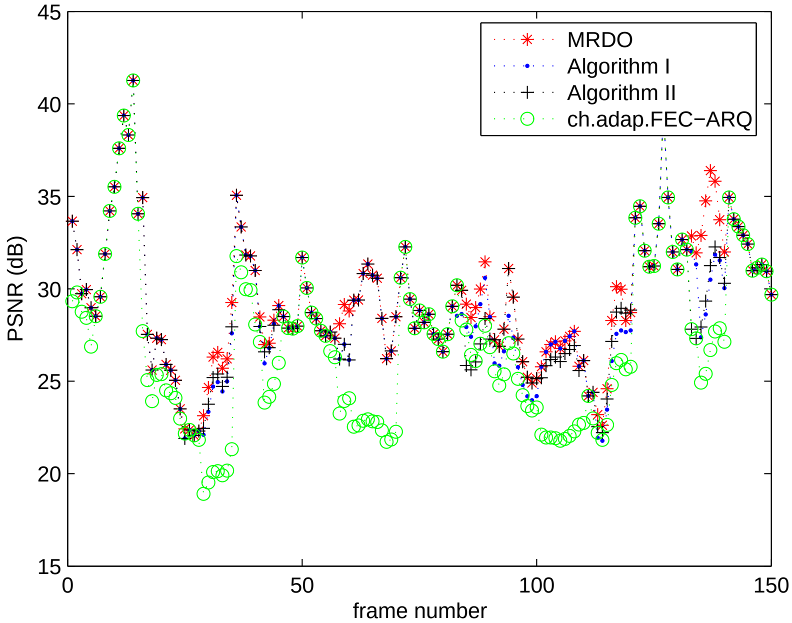

Figure 6.5: PSNR comparison of QCIF size *Foreman* sequence in 10 dB channel

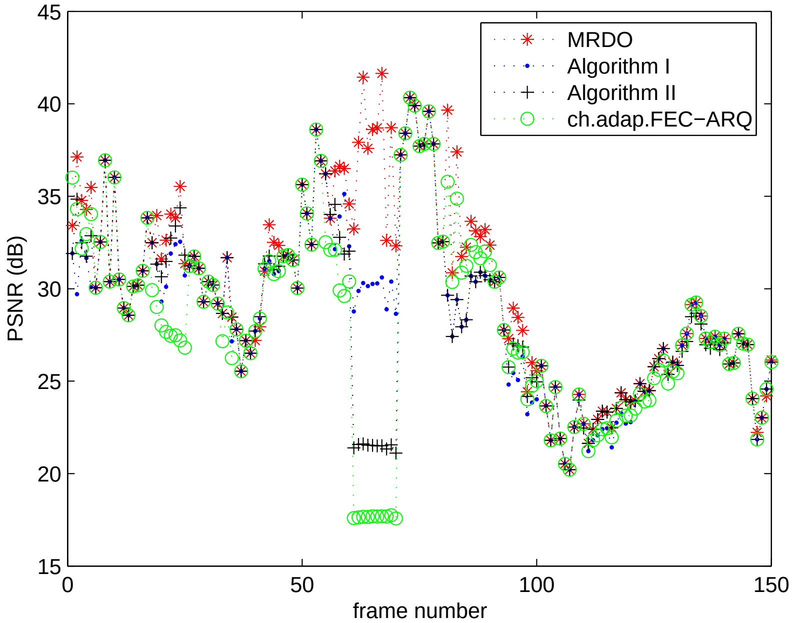

Figure 6.6: PSNR comparison of QCIF size *Carphone* sequence in 10 dB channel

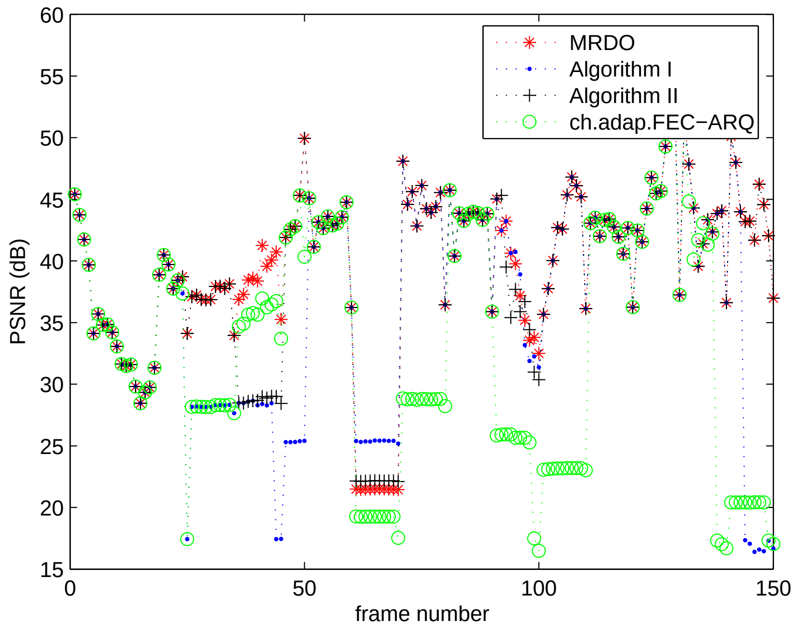

Figure 6.7: PSNR comparison of QCIF size *Mother and Daughter* sequence in 10 dB channel

6.2.2 Experiments with CIF Video Sequences

In this section, we present the performance of the proposed algorithms with CIF size video sequences in 5 dB and 10 dB channels. The video bit rate is set to 140 kbps. In this case, all of the channel capacity (204.8 kbps) will be used. The video sequences used in simulations are *Foreman*, *News*, and *Football*. Table 6.6 and Table 6.7 show PSNRs for each video sequence in 5 dB and 10 dB channels, respectively.

The above experiment prove that MRDO algorithm improves video quality by up to 4 dB as compared to traditional methods. Reduced complexity schemes are within 1 dB of the optimal algorithm achieving approximately 3 dB improvement over traditional ones. Table 6.6 and 6.6 also emphasize the mentioned property about videos with high rate of change. *Football* sequence is an example of a quickly changing video sequence where we recorded similar performance in all algorithms.

In Fig. 6.8, Fig. 6.9, Fig. 6.10, Fig. 6.11, Fig. 6.12, and Fig. 6.13, variation of PSNR with time is shown for the *Foreman*, *News*, and *Football* sequences at the receiver measured for 5 dB and 10 dB channels. These figures also show a similar behavior to the QCIF sequences, indicating that traditional methods have deeper fades as compared to proposed algorithms with the same reasoning.

Table 6.6: PSNR in 5 dB channel for CIF sequences

	Foreman (dB)	News (dB)	Football (dB)
MRDO	25.25	31.93	20.88
Algorithm I	24.74	29.74	20.84
Algorithm II	24.32	30.14	20.83
channel adaptive FEC-ARQ	22.40	24.04	20.77
fixed FEC	21.50	21.72	20.70
fixed FEC-ARQ	22.24	22.07	20.76
channel adaptive FEC	21.25	20.25	20.72

Table 6.7: PSNR in 10 dB channel for CIF sequences

	Foreman (dB)	News (dB)	Football (dB)
MRDO	26.77	35.37	20.90
Algorithm I	26.77	34.87	20.90
Algorithm II	26.51	34.82	20.90
channel adaptive FEC-ARQ	25.95	32.50	20.86
fixed FEC	25.44	29.46	20.85
fixed FEC-ARQ	25.63	32.03	20.86
channel adaptive FEC	24.47	26.12	20.82

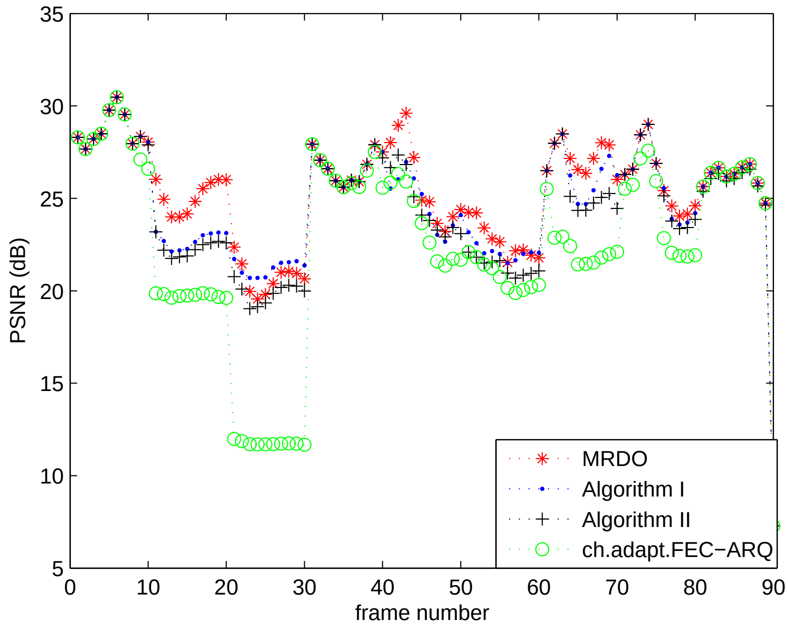

Figure 6.8: PSNR comparison of CIF size *Foreman* sequence in 5 dB channel

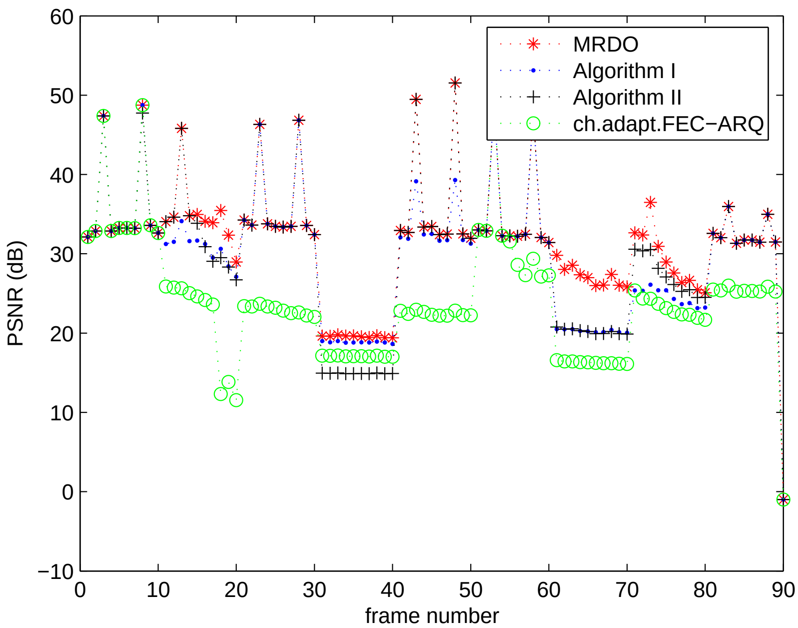

Figure 6.9: PSNR comparison of CIF size *News* sequence in 5 dB channel

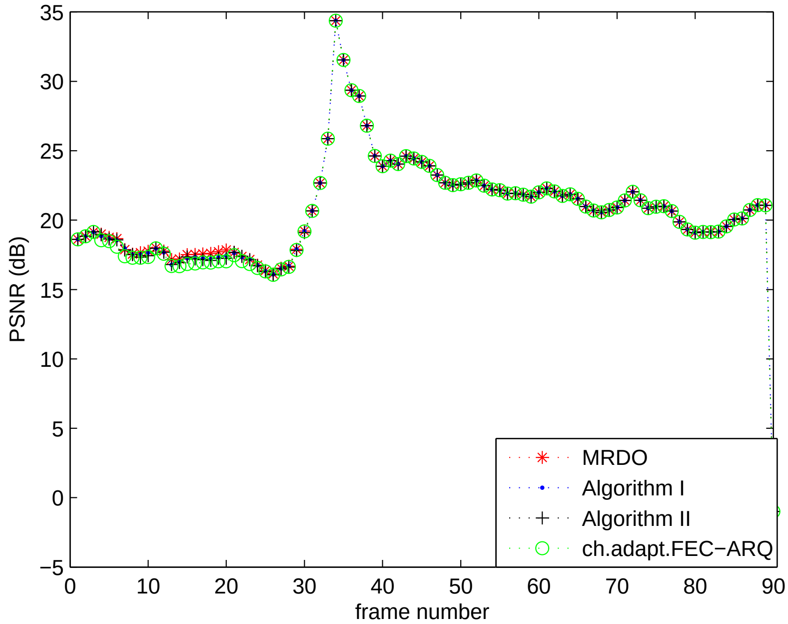

Figure 6.10: PSNR comparison of CIF size *Football* sequence in 5 dB channel

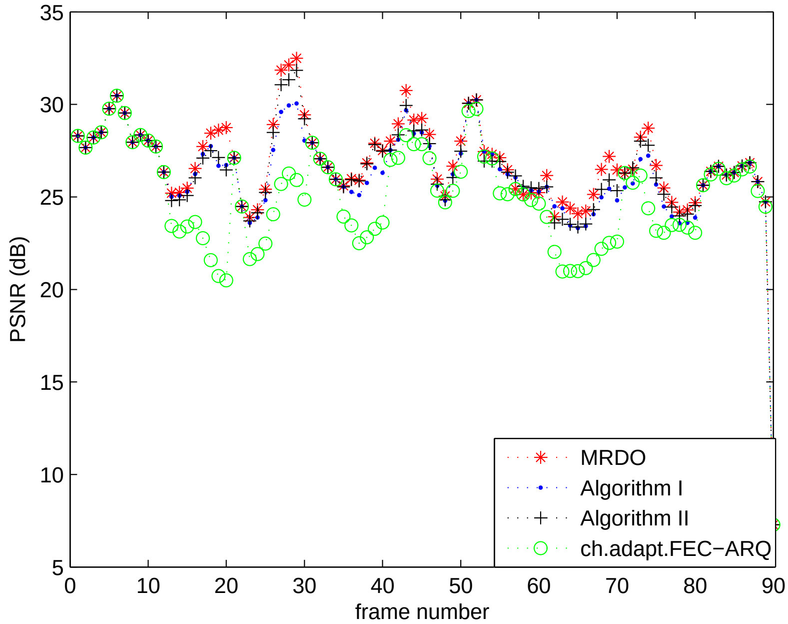

Figure 6.11: PSNR comparison of CIF size *Foreman* sequence in 10 dB channel

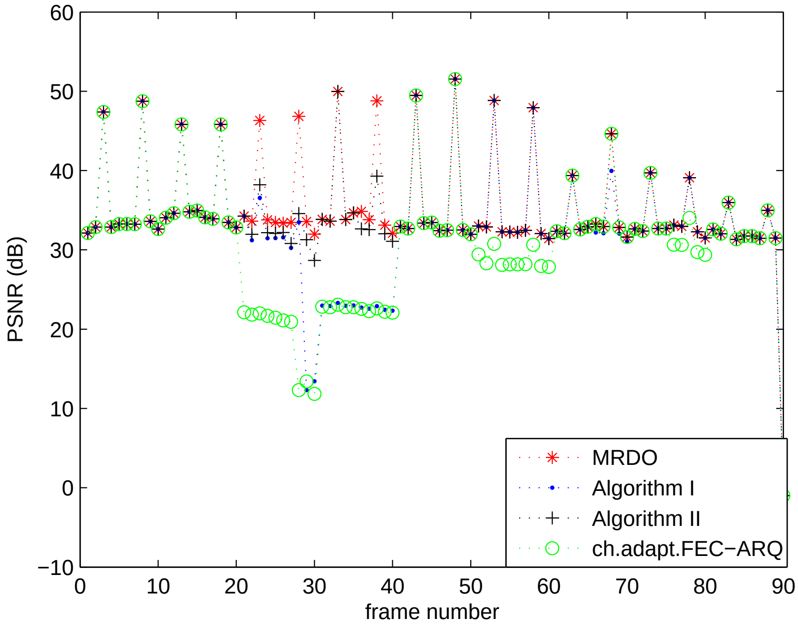

Figure 6.12: PSNR comparison of CIF size *News* sequence in 10 dB channel

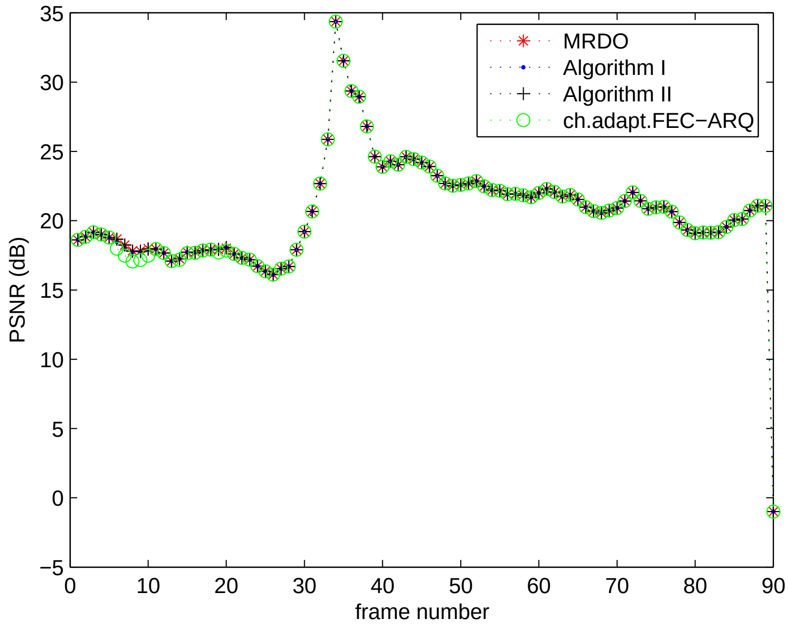

Figure 6.13: PSNR comparison of CIF size *Football* sequence in 10 dB channel

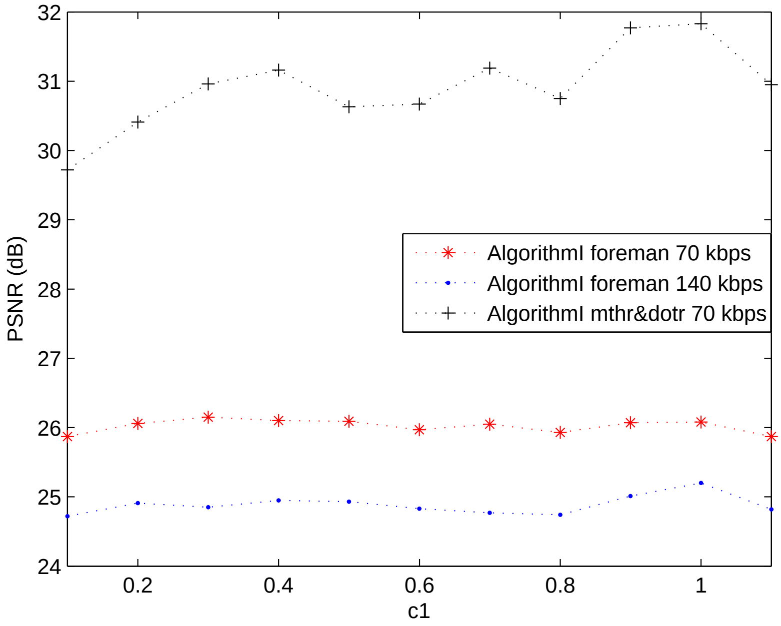

Figure 6.14: Algorithm I - PSNR versus c_1 in 5 dB channel

6.3 Sensitivity Analysis for Reduced Complexity Algorithms

Reduced complexity algorithms proposed in Chapter 5 use constant values c_1 and c_2 in order to calculate distortion metric. In order to choose a packet with higher importance, it is assumed that these constants should be $c_1 \leq 1$ and $c_2 \leq 1$. However, we do not give exact values for these two constants. Instead, we have performed simulations in order to decide which c_1 and c_2 values give more better results. We have simulated Algorithm I and Algorithm II in 5 dB and 10 dB channels with different video sequences and video rates. Fig. 6.14, Fig. 6.15, Fig. 6.16, and Fig. 6.17 shows PSNR change with respect to c_1 and c_2. From these figures, it is seen that there is not a drastic change in PSNR when $c_1 \leq 1$ and $c_2 \leq 1$. However, the highest quality is achieved when c_1 and c_2 take value around 0.9. Therefore, in all simulations whose results are shown in Section 6.2, we have taken the value of c_1 and c_2 as 0.9.

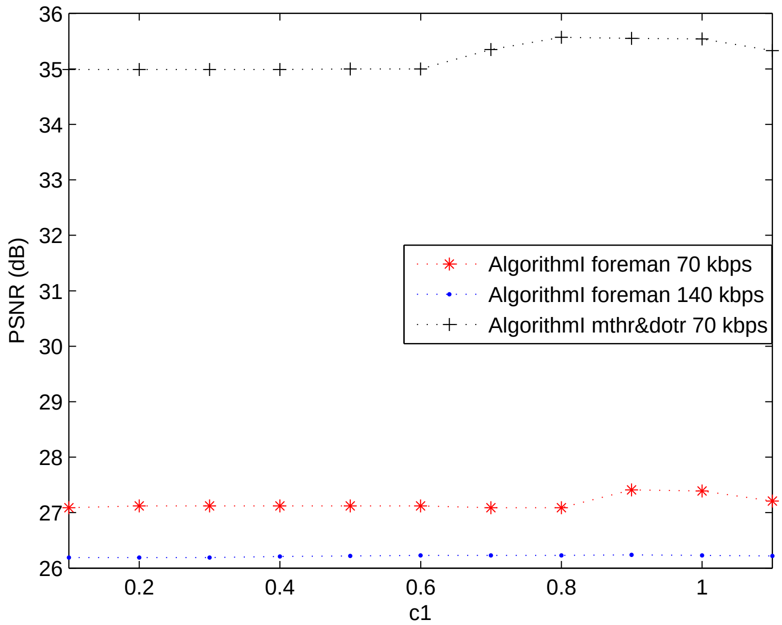

Figure 6.15: Algorithm I - PSNR versus c_1 in 10 dB channel

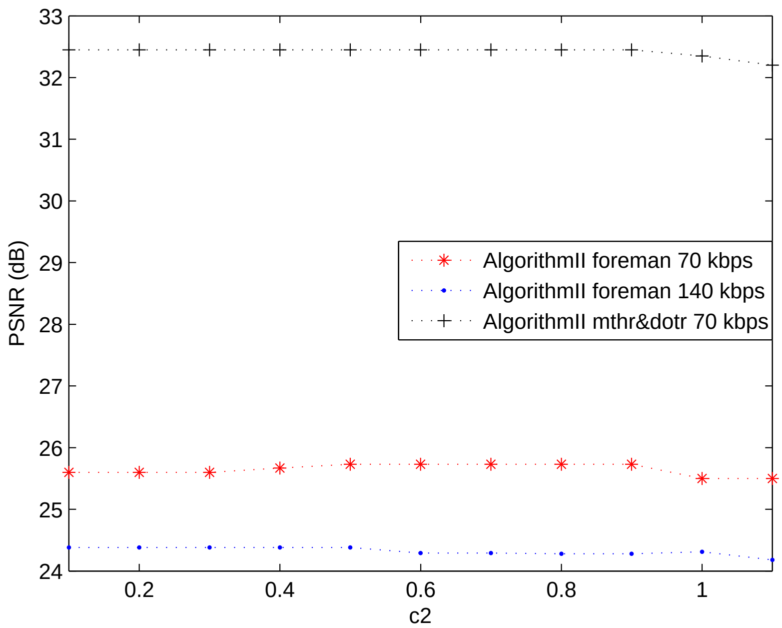

Figure 6.16: Algorithm II - PSNR versus c_2 in 5 dB channel

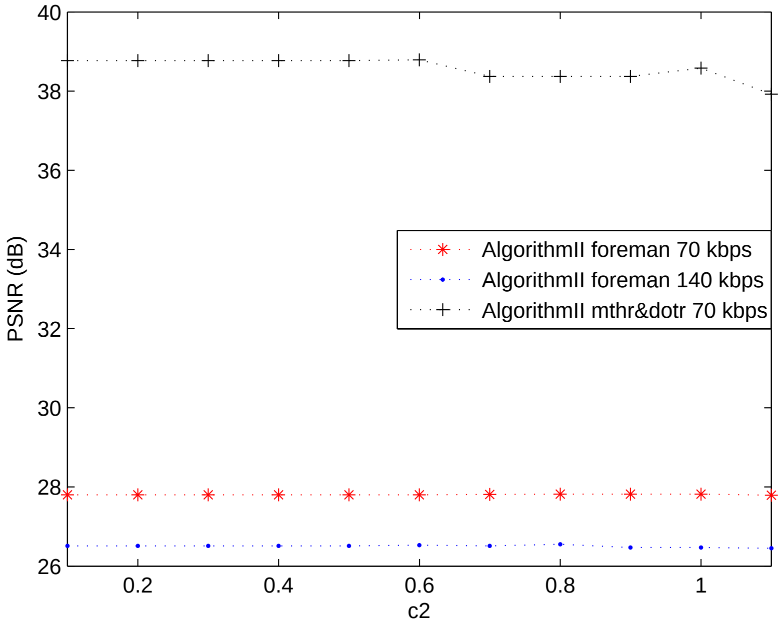

Figure 6.17: Algorithm II - PSNR versus to c_2 in 10 dB channel

6.4 Computational Complexity

In this section, we evaluate the complexity of the proposed algorithms and classical approach. We present the analytical complexity that measures and also provides practical results through simulation time of all algorithms.

The order of computational complexity for the rate distortion optimization algorithm in [1], is calculated in [48], and worst case computational complexity while finding optimal policy vector is found as $O(N2^N + L^2|\Gamma|^L)$ where N is the length of the transmission policy vector, L is the number of packets in the transmit buffer, and Γ is the set of convex hull transmission policies. This set is found with simulations. The same complexity analysis can be applied and extended for the MRDO algorithm resulting in a complexity in the order $O(N(M+1)^N) + L|\Gamma|^L$ where M is the size of the FEC rate set. Γ for MRDO algorithm is also convex hull transmission policies, and can be found by using similar simulations used to find the set for rate distortion optimization algorithm. It is clear that this set is larger than the set of convex hull transmission policies. Obviously, in both algorithms, the order of complexity increases exponentially with the number of packets in the buffer with a higher rate for MRDO due to increased data rates.

The order of computational complexity for RC algorithms is evident: For Algorithm I, if the FEC rate lookup table is already constructed the order of complexity is $O(L)$. Generalizing this analysis, including the learning stage, the order of complexity becomes $O(p(N(M+1)^N + L^2|\Gamma|^L) + (1-p)L)$ where p is the probability that the given state does not exist in the lookup table. For Algorithm II, the computational complexity is determined as $O(LM)$. The computational complexities of the proposed algorithms are compared with the channel adaptive FEC-ARQ scheme which is the simplest with complexity order $O(L)$.

From analytical results, it is realized that MRDO algorithm has high computational complexity than RC algorithms and channel adaptive FEC-ARQ. The simulation results support this realization. However, giving duration in seconds for each algorithm to complete a specific simulation is a relative measurement, because it is affected highly by simulation environment. Channel adaptive FEC-ARQ algorithm is known that it is a real-time applicable algorithm with very low computational complexity. Therefore, it is taken as a reference to compare other algorithms complexity. We have simulated 30 s streaming scenario for 70 kbps and 140 kbps video rates, and for 5 dB and 10 dB channels, and simulation duration of channel adaptive FEC-ARQ algorithm is taken as 1 unit. The durations for the other algorithms are compared with simulation time of channel adaptive FEC-ARQ. The results are given in Table 6.8 for sequence *Foreman*.

As it is seen from Table 6.8, MRDO algorithm is very complex in computation when compared with a real-time streaming system channel adaptive FEC-ARQ. Therefore, MRDO algorithm cannot be used in real time streaming systems although its high SNR efficiency. This result is known previously since rate distortion optimization algorithms require too many calculations and iterations to find solution. However, MRDO algorithm gives us insight about how one improves quality by using parameters used by Algorithm I and II with reduced complexity. In Section 6.2.1 and 6.2.2, it is seen that proposed reduced complexity algorithms achieves high SNR efficiency. Table 6.8 shows, computational complexity of these algorithms is low as compared with channel adaptive FEC-ARQ. This makes our algorithms very suitable to third generation mobile multimedia streaming services.

Table 6.8: Complexity comparison

Ch. Cap. / Average SNR	MRDO	Algorithm I	Algorithm II	ch.adap.FEC-ARQ
70 kbps / 5 dB	13.80	0.89	1.06	1
70 kbps / 10 dB	10.18	1.05	1.27	1
140 kbps / 5 dB	18.45	0.84	1.05	1
140 kbps / 10 dB	11.82	1.02	1.29	1

Chapter 7

Conclusions and Future Work

7.1 Conclusions

In this thesis, we analyzed multimedia streaming over wireless channels. The challenges of wireless multimedia streaming are overcome by using sophisticated transport algorithms with scheduling considering the properties of multimedia data and the wireless channel.

The first proposed transport algorithm is multi-rate distortion optimization which is an extended and modified version of the work in [1],[2], respectively. MRDO optimizes packet selection and FEC rate jointly by minimizing a distortion function that includes properties, deadlines, dependencies, and loss probabilities of current and future packets subject to the fact that total rate cannot be greater than physical channel capacity. Simulation results show that MRDO provides 1-5 dB SNR improvement depending on the characteristics of the simulation video sequence. Although MRDO achieves significant SNR improvement, it is not applicable to real-time streaming systems because of its high computational complexity.

To reduce computational complexity of multi-rate distortion optimization, two reduced complexity algorithms are proposed. Algorithm I is a technique that selects a packet from the transmit buffer according to the packet importance and deadline. An FEC rate is chosen from a lookup table that contains FEC rate, channel state pairs. This lookup table is constructed by running the MRDO first. If acknowledgement for a specific packet is not received within a specific time, the packet is retransmitted again. Algorithm II jointly performs packet selection and FEC rate by minimizing a distortion metric. This distortion metric includes packet delay, corruption probability, and packet importance. Since packet corruption probability is directly related with chosen FEC rate, FEC rate selection is also performed by minimizing this distortion metric. Same ARQ scheme as Algorithm II is used. Algorithm I and Algorithm II approaches to MRDO significantly. SNR difference between MRDO and Algorithm I, II is less than 1 dB. In addition, Algorithm I and II significantly outperform when compared with traditional error control techniques, especially channel adaptive FEC-ARQ. As it, is seen Algorithm I and II shows 1-4 dB better SNR quality than channel adaptive FEC-ARQ. Furthermore, computational complexity of Algorithm I and II is low enough to be implemented in real time streaming systems.

We applied MRDO algorithm and reduced complexity algorithms in the transport protocol of UMTS third generation mobile communications system. The flexibility of CRC and code rate design in UMTS provides us to modify the UMTS system in order to send selected packet from transmit buffer and use selected FEC rate.

7.2 Future Work

As a future work, the proposed transport algorithms may be applied to wireless local area networks (WLAN). Since UMTS is a cellular network, the wireless channel is modeled with Rayleigh fading distribution. However, in WLANs this assumption is not correct. More sophisticated channel modeling by considering the multi access channel and collision probabilities is necessary. Our transport algorithms can be used in the WLAN structure with the new channel model.

In this thesis, we focused on a point-to-point user communication. As mentioned, this assumption is true with static channel allocation for W-CDMA air interface of UMTS system. However, for other systems such as TD-CDMA air interface of UMTS and wireless local area networks, it is necessary to allocate data rate to multiple users. By using transport level multimedia properties and channel properties this allocation is performed optimally.

Bibliography

[1] P.A.Chou, Z.Miao, "Rate Distortion Optimized Streaming of Packetized Media", Technical Report, MSR-TR-2001-35, Microsoft Research, 2001.

[2] J.Chakareski, P.A.Chou, "Application Layer Error-Correction Coding for Rate Distortion Optimized Streaming to Wireless Clients", *IEEE Transactions on Communications*, Vol. 52, No.10, October, 2004.

[3] *Overview of the MPEG-4 Standard*, ISO/IEC JTC1/SC29/WG11 N4668, March, 2002.

[4] *Advanced Video Coding for Generic Audiovisual services*, ITU-T, 2003.

[5] I.E.G. Richardson, *H.264 and MPEG-4 Video Compression*, Wiley, 2004.

[6] *Information Technology - coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbps(MPEG1)*, ISO/IEC 11172, 1993.

[7] *Information technology: generic coding of moving pictures and associated audio information (MPEG-2)*, ISO/IEC 13818, 1995.

[8] A.Gibson, "H.264 Video Compression Standard", Master Thesis, Queen's University, 2002.

[9] *H.261 - Video CODEC for audiovisual services at p×64 kbps*, ITU-T, 1993.

[10] *H.263, Video coding for low bit rate communcaton*, Version 2, ITU-T, 1998.

[11] T. Stockhammer, M.M. Hannuksela, T. Wiegand, "H.264/AVC in Wireless Environments", *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 13, No. 7, July, 2003.

[12] M. Fiedler, "Implementation of a basic H.264/AVC Decoder", Seminar Paper, Chemnitz University of Technology, June, 2004.

[13] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, July, 2003.

[14] S.Wenger, M.M. Hannuksela, T.Stockhammer, M. Westelund, D. Singer, "RTP Payload Format for H.264 Video", RFC 3984, February, 2005.

[15] Y. Wang, Q. F. Zhu, "Error Control and Concealment for Video Communication: A Review", *Proceedings of IEEE*, Vol.86, No.5, May, 1998.

[16] B. Sklar, *Digital Communications, Fundementals and Applications*, NJ, Prentice Hall, 2000.

[17] R. B. Wells, *Applied Coding and Information Theory for Engineers*, NJ, Prentice Hall, 1999.

[18] A. L. Garcia, I. Widjaja, *Communication Networks, Fundemental Concepts and Key Architectures*, Mc-Graw Hill, 2004.

[19] H. Liu, M. E. Zarki, "Performance of H.263 Video Transmission over Wireless Channels Using Hybrid ARQ", *IEEE Journal on Selected Areas in Communicatins*, Vol. 15, No. 9, December, 1997.

[20] H. Liu, M. E. Zarki, "Transmission of video telephony images over wireless channels", Wireless Networks2, J.C. Baltzer AG, Science Publishers, 1996.

[21] J. Cai, Q. Zhang, W. Zhu, C. W. Chen, "An FEC-Based Error Control Scheme for Wireless MPEG-4 Video Transmission", *IEEE Wireless Communications and Networking Conference*, Vol.3, pp.1243-1247, 2000.

[22] C.C. Liu, S.C.S. Chen, "Providing Unequal Reliability for Transmitting Layered Video Streams over Wireless Networks by Multi-ARQ Schemes", *IEEE International Conference on Image Processing*, Vol.3, pp.100-104, October, 1999.

[23] M. Podolsky, S. McCane, M. Vetterli, "Soft ARQ for layered streaming media", *Tech. Rep UCB/CSD-98-1024*, University of California, Computer Science Division, Berkeley, 1998.

[24] Y. Shan, A. Zakhor, "Cross Layer Techniques for Adaptive Video Streaming over Wireless Networks", *IEEE International Conference on Multimedia Expo*, Vol. 1, pp. 277-280, 2002.

[25] X. Liu, W. Badawy, "A Novel Adaptive Error Control Scheme for Real-Time Wireless Video Streaming", *IEEE Information Technology: Research and Education*, pp.460-463, August, 2003.

[26] M.U.Demircin, Y. Altunbasak, "Finite horizon FEC rate adaptation for real time wireless multimedia", *IEEE Wireless Communications and Networking Conference*, pp.1770-1775, 2004.

[27] P. Bucciol, E. Masala, "Perceptual ARQ for H.264 Video Streaming over 3G Wireless Networks", *IEEE International Conference on Communications*, Vol. 3, pp.1288-1292, 2004.

[28] S. Aramvith, C.W.Lin, S. Roy, M.T. Sun, "Wireless Video Transport Using Conditional Retransmission and Low-Delay Interleaving", *IEEE Transactions on Circuit and Systems for Video Technology*, Vol. 12, No. 6, June, 2002.

[29] Y. L. Liang, J.G. Apostolopoulos, B. Girod, "Model-Based Delay-Distortion Optimization for Video Streaming Using Packet Interleaving", *IEEE Signals, Systems and Computers*, Vol.2, pp.1315-1319, November, 2002.

[30] D. Tian, X. Li, G. A. Regib, Y. Altunbasak, J. R. Jackson, "Optimal Packet Scheduling for Wireless Video Streaming with Error-Prone Feedback", *IEEE WCNC*, Vol.2, pp.1287-1292, March, 2004.

[31] S.H.Kang, A.Zakhor, "Packet Scheduling Algorithm for Wireless Video Streaming",*International Packetvideo Workshop*, 2002.

[32] Z. Jiang, L. Kleinrock, "A Packet Selection Algorithm for Adaptive Transmission of Smoothed Video Over a Wireless Channel", *Journal of Parallel and Distributed Computing*,Vol.60, No.4, April, 2004.

[33] M. Elaoud, P. Ramanathan, “Adaptive Use of Error Correcting Codes for Real-Time Communication in Wireless Networks”, *IEEE INFOCOM*, pp.548-555, 1998.

[34] D. Qiao, K. G. Shin, “A Two-Step Adaptive Error Recovery Scheme for Video Transmission over Wireless Networks”, *IEEE INFOCOM*, pp. 1698-1704, 2000.

[35] D.N.Rowitch, L.B.Milstein, “On the Performance of Hybrid FEC/ARQ Systems Using Rate Compatible Punctured Turbo (RCPT) Codes”, *IEEE Transactions on Communications*, Vol. 48, No. 6, June, 2000.

[36] http://www.itu.int/home/imt.html, 2005.

[37] Eurescom Project P921, “Radio Link Simulations and Application Trials”, European Institute for Research and Strategic Studies in Telecommunications, 2000.

[38] *Multiplexing and Channel Coding (TDD)*, 3GPPP TS 25.222 V6.2.0, 2004.

[39] S. Haykin, M. Moher, *Modern Wireless Communications*, Prentice Hall, 2005.

[40] Q.Zhang, S.A. Kassam, “Finite-State Markov Model for Rayleigh Fading Channels”, *IEEE Transactions on Communications*, Vol. 47, No.11, November, 1999.

[41] E. N. Gilbert, “Capacity of a burst-noise channel”, Bell Sys.Tech. J., vol. 39, pp. 1253-1266, 1960.

[42] B. Elspas, “Design and instrumentation of error correcting codes”, Stanford Research Institute, Menlo Park, Calif., Interim Tech. Rept. RADC TR61-259, Project 3318, October, 1961.

[43] H.S. Wang, N. Moayeri, “Finite-State Markov Channel - A Useful Model for Radio Communication Channels”, *IEEE Transactions on Vehicular Technology*, Vol.44, No.1, February, 1995.

[44] C.Bormann, Robust Header Compression (RoCH), July, 2001. Available: http://www.ietf.org/rfc/rfc3095.txt.

[45] H. Seferoglu, Y. Altunbasak, O.Gurbuz, O. Ercetin, "Rate Distortion Optimized Joint ARQ-FEC Scheme for Real-Time Wireless Multimedia", *IEEE International Conference on Communications*, 2005.

[46] http://iphome.hhi.de/suehring/tml/download/, 2005.

[47] http://ise.stanford.edu/video.html, 2005.

[48] M.Roder, J. Cardinal, R. Hamzaoui, "On the Complexity of rate-distortion optimal streaming of packetized media", *Konstanzer Schriften in Mathematik und Informatik*, Januar, 2004.