Error resilient packet switched H.264 video telephony over third generation networks

Muneeb Dawood

Faculty of Technology, De Montfort University

A thesis submitted in partial fulfillment of the
requirements of De Montfort University
for the degree of
Doctor of Philosophy

December 2010

Abstract

Real-time video communication over wireless networks is a challenging problem because wireless channels suffer from fading, additive noise and interference, which translate into packet loss and delay. Since modern video encoders deliver video packets with decoding dependencies, packet loss and delay can significantly degrade the video quality at the receiver. Many error resilience mechanisms have been proposed to combat packet loss in wireless networks, but only a few were specifically designed for packet switched video telephony over Third Generation (3G) networks.

The first part of the thesis presents an error resilience technique for packet switched video telephony that combines application layer Forward Error Correction (FEC) with rateless codes, Reference Picture Selection (RPS) and cross layer optimization. Rateless codes have lower encoding and decoding computational complexity compared to traditional error correcting codes. One can use them on complexity constrained hand-held devices. Also, their redundancy does not need to be fixed in advance and any number of encoded symbols can be generated on the fly. Reference picture selection is used to limit the effect of spatio-temporal error propagation. Limiting the effect of spatio-temporal error propagation results in better video quality. Cross layer optimization is used to minimize the data loss at the application layer when data is lost at the data link layer. Experimental results on a High Speed Packet Access (HSPA) network simulator for H.264 compressed standard video sequences show that the proposed technique achieves significant Peak Signal to Noise Ratio (PSNR) and Percentage Degraded Video Duration (PDVD) improvements over a state of the art error resilience technique known as Interactive Error Control (IEC), which is a combination of Error Tracking and feedback based Reference Picture Selection. The improvement is obtained at a cost of higher end-to-end delay.

The proposed technique is improved by making the FEC (Rateless code) redundancy channel adaptive. Automatic Repeat Request (ARQ) is used to adjust the redundancy of the Rateless codes according to the channel conditions. Experimental results show that the channel adaptive scheme achieves significant PSNR and PDVD improvements over the static scheme for a simulated Long Term Evolution (LTE) network.

In the third part of the thesis, the performance of the previous two schemes is improved by making the transmitter predict when rateless decoding will fail. In this case, reference picture selection is invoked early and transmission of encoded symbols for that source block is aborted. Simulations for an LTE network show that this results in video quality improvement and bandwidth savings.

In the last part of the thesis, the performance of the adaptive technique is improved by exploiting the history of the wireless channel. In a Rayleigh fading wireless channel, the RLC-PDU losses are correlated under certain conditions. This correlation is exploited to adjust the redundancy of the Rateless code and results in higher Rateless code decoding success rate and higher video quality. Simulations for an LTE network show that the improvement was significant when the packet loss rate in the two wireless links was 10%.

To facilitate the implementation of the proposed error resilience techniques in practical scenarios, RTP/UDP/IP level packetization schemes are also proposed for each error resilience technique.

Compared to existing work, the proposed error resilience techniques provide better video quality. Also, more emphasis is given to implementation issues in 3G networks.

Acknowledgments

I would like to thank my doctoral adviser Prof. Dr. Raouf Hamzaoui for his consistent support throughout the Ph.D studies. His constructive criticism and guidance played a key role in the completion of this thesis. I want to express my gratitude to Prof. Marwan Al-Akaidi for being my second supervisor and for his constant support. I also thank Dr. Shakeel Ahmad for the useful discussions that we had. I thank De Montfort University for providing me with financial support for the duration of this dissertation. Finally, I want to thank my parents for their support, guidance and encouragement.

Contents

List of Figures

List of Tables

List of Abbreviations

2G	Second Generation
3G	Third Generation
3GPP	Third Generation Partnership Project
ACK	Acknowledgment
ARQ	Automatic Repeat Request
ASO	Arbitrary Slice Order
AVC	Advanced Video Coding
BTT	Backward Trip Time
CABAC	Context Adaptive Binary Arithmetic Coding
CAVLC	Context Adaptive Variable Length Coding
CN	Core Network
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
DCT	Discrete Cosine Transform
EF	Expanding Factor
EnB	Evolved Node B
FEC	Forward Error Correction
FMO	Flexible Macroblock Ordering
FPS	Frames per Second
FTT	Forward Trip Time

GGSN	Gateway GPRS Support Node
GOB	Group of Blocks
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
HARQ	Hybrid Automatic Repeat Request
HSDPA	High Speed Downlink Packet Access
HSPA	High Speed Packet Access
HSUPA	High-Speed Uplink Packet Access
IEC	Interactive Error Control
IP	Internet Protocol
LT	Luby Transform
LTE	Long Term Evolution
MAC	Mediim Access Control
MC	Motion Compensation
ME	Motion Estimation
MMCO	Memory Management Control Operation
MME/GW	Mobility Management Entity/Gateway
MSE	Mean Squared Error
NACK	Negative Acknowledgement
NAL	Network Abstraction Layer
NALU	Network Abstraction Layer Unit
NTP	Network Time Protocol
PDCP	Packet Data Convergence Protocol
PDVD	Percentage of Degraded Video Duration

P-GW	Packet Gateway
PHY	Physical
PR	Periodic Reference
PSNR	Peak Signal to Noise Ratio
QCIF	Quarter Common Intermediate Format
QoS	Quality of Service
QP	Quantization Parameter
RACK	Request Acknowlegement
RESCU	Recovery from Error Spread using Continuous Updates
RLC	Radio Link Control
RLC-PDU	Radio Link Control Protocol Data Unit
RLC-SDU	Radio Link Control Service Data Unit
RLC-UM	Radio Link Control Unacknowledged Mode
RNC	Radio Network Controller
ROHC	Robust Header Compression
RPS	Reference Picture Selection
RTCP	Real-Time Transport Control Protocol
RTP	Real-Time Transport Protocol
SaW	Stop and Wait
SGSN	Serving GPRS Support Node
S-GW	Serving Gateway
SIP	Session Initiation Protocol
TCP	Transport Control Protocol
TTI	Transmission Time Interval

UDP	User Datagram Protocol
UE	User Equipment
UMTS	Universal Mobile Telecommunications System
UTRAN	UMTS Terrestrial Radio Access Network
VCL	Video Coding Layer
WCDMA	Wideband Code Division Multiple Access

Chapter 1

Introduction

1.1 Motivation

3G networks allow higher data transmission rates than 2G networks. High data rate on 3G networks makes multimedia applications such as mobile TV, video streaming and video telephony possible. Due to their revenue generation potential, multimedia applications over 3G networks are attractive for mobile network operators. Multimedia applications over 3G networks also improve user experience. Thus there is a need for multimedia applications over 3G networks from operators and users perspective.

Currently commercial solutions for mobile video telephony are based on circuit switching. In circuit switching, network resources are reserved for the duration of the call. This results in wastage of network resources if the full allocated bandwidth is not utilized all the time. Thus, circuit switching reduces network capacity, which makes it expensive for the end user. There is a need to find cheaper alternatives to circuit switching for mobile video telephony. Packet switched mobile video telephony is one such alternative.

In packet switching, the end-to-end video data stream is divided into packets. Packets are then transported over the 3G network. The advantage of using packet switching is that more users can use the same link simultaneously. Thus, packet switching results in increased network capacity as well as cheaper calls compared to circuit switching. There are number of challenges for packet switched mobile video telephony.

1.2 Problem formulation

Video packets transmitted over air are subject to signal interference, electrical noise, and low signal strength, which can cause errors in packets. The corrupted packets are discarded by the lower layers according to the 3G standard and not passed on to the application layer. Due to dependencies in the compressed video, loss of video data packets results in video quality degradation at the receiver. Video quality degradation results in poor user experience. Thus, there is a need for error control mechanisms to improve the video quality for mobile video telephony.

1.3 Challenges of packet switched video telephony over 3G networks

The challenges of packet switched video telephony over 3G networks are as follows:

- Video telephony is classified as a conversational service by the 3GPP standard. A necessary requirement for video telephony is low end-to-end delay. The end-to-end delay in video telephony is the delay between capturing a video frame at one end point and the play back of the same video frame at the other end point. According to the 3GPP standard, the maximum end-to-end delay should be less than 400 ms [1].

- The bit error in wireless channels is due to shadowing, fading, interference and channel contention [2]. The packets which are corrupted by bit errors are discarded by the lower layers at the receivers. The number of retransmissions at the physical layer is limited because retransmissions of lost packets introduce additional delay, which can be unacceptable for conversational video services.

- The bit rate specified for video telephony over 3G networks is 32-384 kbit/s [1]. Raw video data requires much higher bit rate and cannot be transmitted in real-time at this range of bit rate. To accommodate the real-time video data in this bit rate range, the video data must be compressed. Video compression is used to reduce the size of the video for transmission. H.264 is the state of the art video compression standard. 3GPP has specified H.264 as one of the video codecs for video telephony over 3G networks. However, loss of packets containing compressed video data results in severe video quality degradation. The quality degradation is due to the lost data of a video frame. Due to the spatial-temporal prediction,

a single packet loss in the video data leads to errors in the successive video data which has been received correctly. This is known as spatio-temporal error propagation. To maintain acceptable quality in conversational video services, effective error-resilience techniques are required. The goal of the error resilience techniques is to reduce the effect of packet loss on the video quality and if an error or packet loss does occur then the error resilience techniques should stop the effect of error on the video data which has been received correctly [3].

- In conversational video applications, video encoding, transmission and video decoding should be done in real-time [2]. Video encoding and decoding is done on user terminals, which are typically hand-held devices. These devices operate on limited battery power. The error resilience techniques for mobile video telephony should have low computational complexity.

1.4 Error control in 3GPP HSPA and LTE standards

In video telephony (see Figure 1.1), there is a bidirectional video communication. The end users are human beings. Each end user uses a mobile phone (user equipment) to communicate over the 3G network. Each user equipment sends and receives video simultaneously. Video data travels through wireless and wired links.

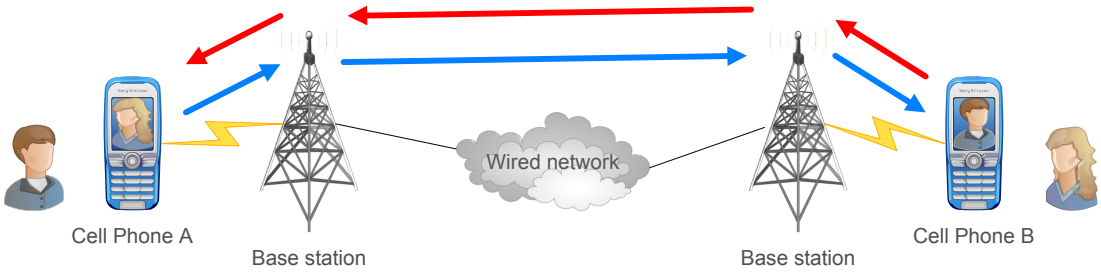


Figure 1.1: A video telephony scenario

In 3G networks, the link between the user equipment and the node B (or eNB) is a wireless link. Node B and eNB are fixed. The user equipment may be mobile. The non-line-of-sight (NLOS) radio communication between a node B (eNB) and a mobile user equipment results in Rayleigh fading [4]. Rayleigh fading is time varying [4]. Rayleigh fading results in bit errors in the packets at the physical layer [4].

To overcome the effect of bit errors in the packets, 3GPP networks implement error

control mechanisms at the physical (PHY) and Radio Link Layer (RLC) [4].

At the physical layer, an N channel stop and wait Hybrid ARQ (HARQ) is used [4] to protect packets against bit errors [4]. The physical layer HARQ does not guarantee 100% error recovery because the size of ACKs and NACKs used for physical layer HARQ is only 1 bit [5]. The bit corruption of NACK may lead to the transmitter not sending any more retransmissions, hence causing packet errors at the RLC level [5]. Also, the error correction capability of physical layer HARQ is limited for delay constrained conversational applications [6].

In conversational video applications such as video telephony, RLC unacknowledged mode is used. In this mode there are no retransmissions of corrupted RLC-PDUs.

At the transport layer, the video data is carried in User Datagram Protocol (UDP) [7] packets. UDP is suitable for real-time Communication but is an unreliable protocol. TCP cannot be used due to strict delay constraints.

Corrupted data link and IP packets are discarded and not passed on to the application layer. Thus physical layer errors result in packet loss at the application layer. However, there is no 3GPP standardized error control technique for mobile video telephony at the application layer.

1.5 Goal of error resilience techniques

Application layer packet losses result in video quality degradation. There is a need to provide error control mechanism at the application layer to improve video quality.

The goal of error resilience (or error control) techniques is to overcome the effect of errors due to packet loss or corruption [8]. The goal of error control techniques for video transmission is to improve the quality of video and lessen the effect of video quality degradation due to video data packet loss or corruption [8]. Error control techniques for video transmission include the following types [8]:

- **Forward Error Correction (FEC):** In FEC, the transmitter adds redundancy to the source data.

- **Automatic Repeat Request (ARQ):** In ARQ, lost or corrupted data is retransmitted by the transmitter.
- **Error concealment:** Error concealment is done at the video decoder to conceal the effect of lost or corrupted data [8]. These techniques do not protect the packets against packet errors and also do not limit the effect of spatio-temporal error propagation.
- **Error resilient source coding:** In error resilient source coding, the error control is done by the source encoder.
- **Hybrid techniques:**

Any of the above methods can be combined to achieve better results.

Most of the error control techniques proposed in the literature cannot be used for video telephony due to delay and complexity constraints. Moreover, only a few were specifically designed for 3G networks. A review and limitations of the existing error resilience techniques for real-time video communication and packet switched video telephony is given in Chapter 3.

1.6 Aims and objectives

The proposed error resilience technique should meet the following aims and objectives:

- It can be used in low end-to-end delay scenario such as mobile video telephony.
- It should be of low encoding and decoding complexity, so that it can be implemented on hand-held devices running on limited battery power.
- It should minimize the effect of packet loss on the video quality.
- It should limit the spatio-temporal error propagation in case of packet losses. Limiting spatio-temporal error propagation is necessary to achieve better video quality and improve user experience.
- The losses between different layers of the 3G protocol stack should be minimized.
- It should be bandwidth efficient.
- It should adapt to varying and dynamic channel conditions.
- It can be implemented in practical scenarios.

1.7 Scope of the thesis

This dissertation proposes practical application-layer error resilience techniques for packet switched video telephony over 3G networks. In 3G networks, error control is applied at the physical and link layers. However, due to stringent time constraints for real-time communication not all packets can be correctly recovered and therefore application layer error control is needed. The standards of 3G networks that are considered in this work are High Speed Packet Access (HSPA) and Long Term Evolution (LTE). The reason for considering HSPA is its wide availability. LTE is an all IP version of 3G network.

The main idea is to combine Forward Error Correction using rateless (LT, Raptor) [9, 10] codes, Reference picture selection [3] and Automatic Repeat Request (ARQ).

The motivation for using Rateless codes is that they have very low encoding and decoding complexity and hence they can be used on complexity constrained hand-held devices. In this work, a real software implementation of LT codes is used. In Automatic Repeat Request, lost packets are retransmitted. In ARQ the number of retransmissions is proportional to the number of packets lost. Hence, ARQ can adapt to varying channel conditions and dynamic packet loss rate. Reference picture selection is a technique that can limit the effect of spatio-temporal error propagation, improving video quality. Cross layer optimization is done to minimize the effect of packet loss in one layer depending on the packet loss rate in the other layer of the 3G protocol stack.

1.8 Contributions of the thesis

The main contributions of this thesis are:

- An error resilience scheme for packet switched mobile video telephony, which uses a combination of application layer fixed redundancy LT coding, Reference Picture Selection and cross layer optimization. This scheme is called **Fixed** scheme in this work. The proposed error resilience technique has been published in [11].
- A combination of application layer adaptive LT coding and reference picture selection. This scheme is called **Adaptive** scheme in this work. The proposed channel adaptive error resilience technique has been published in [12].

- An error resilience scheme for packet switched mobile video telephony, which uses a combination of **Fixed** scheme and early reference picture selection. This scheme is called **Fixed early RPS**.
- An error resilience scheme for packet switched mobile video telephony, which uses a combination of **Adaptive** scheme and early reference picture selection. This scheme is called **Adaptive early RPS**.
- An error resilience scheme for packet switched mobile video telephony, which uses a combination of **Adaptive early RPS** scheme and history of packet loss rate. This scheme is called **Adaptive past channel early RPS** in this dissertation.
- RTP/UDP/IP level packetization schemes that facilitate the implementation of the proposed error resilience techniques in practical scenarios are also presented.

1.9 Outline of the thesis

This dissertation is organized as follows:

Chapter 2 provides necessary background information about 3G mobile cellular networks, H.264 video coding standard and rateless (LT, Raptor) codes. A brief description of video quality metrics such as PSNR and PDVD is also given.

Chapter 3 surveys previous error resilience techniques for real-time video communication.

Chapter 4 proposes an error resilience technique for packet switched mobile video telephony, which combines LT coding, Reference Picture Selection and cross layer optimization. The proposed error resilience technique achieves better PSNR and PDVD results than a state of the art error resilience technique proposed in [13, 14] for 3GPP HSPA networks.

In Chapter 5, the technique is improved by making the FEC (Rateless code) redundancy channel adaptive. Experimental results show that the channel adaptive scheme achieves significant PSNR and PDVD improvements over the static scheme for 3GPP LTE networks.

In Chapter 6, the performance of the previous two schemes is improved by making the transmitter predict when rateless decoding will fail. In this case, reference picture selection is invoked early and transmission of encoded symbols for that source block is aborted. This results in video quality improvement and bandwidth savings for LTE networks.

In Chapter 7, the performance of the adaptive technique is improved by exploiting the history of the wireless channel. The improvement was significant when the packet loss rate in the two wireless links modeled as Rayleigh fading channel, was 10%.

Chapter 8 concludes this dissertation.

Chapter 2

Background

2.1 3GPP Third Generation (3G) cellular system

Third Generation Partnership Project's (3GPP) Third Generation (3G) cellular systems enable digital voice and data communication. 3G systems allow increased bit rate than second generation (2G) cellular systems. The increased bit rate makes multimedia communication such as video telephony psossible. The Universal Mobile Telecommunication System (UMTS) is the earliest of the 3G systems [15]. The UMTS is also referred to as 3GPP Release 99 [15].

2.1.1 3GPP HSPA Cellular System

The 3GPP release 99 standard was improved by the introduction of enhancements [15]. Some of these enhancements were higher peak data rates, lower latency and physical layer Hybrid Automatic Repeat Request [15]. The downlink enhancement is called High Speed Downlink Packet Access (HSDPA) and the uplink enhancement is called High Speed Uplink Packet Access (HSUPA)[15]. HSDPA and HSUPA are collectively known as High Speed Packet Access (HSPA) [15].

HSPA network architecture

The network architecture of HSPA with packet switched core is shown in Figure 2.1 [16]. The network entities are User Equipment (UE), one or many radio access networks and a core network [16]. The description of various network elements is given below:

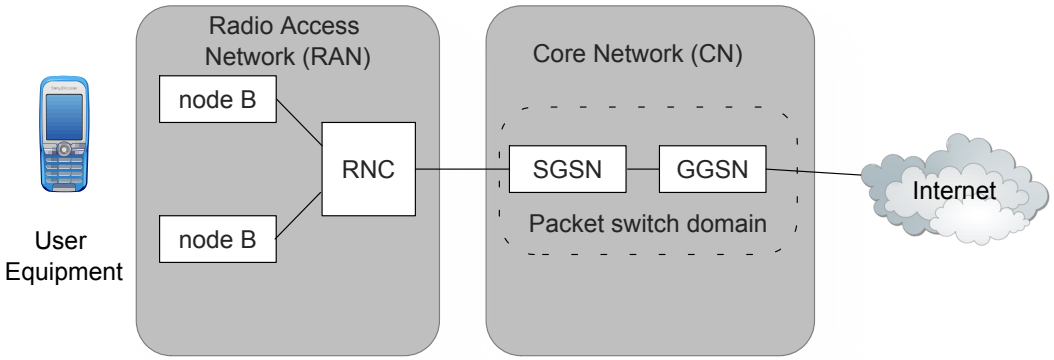


Figure 2.1: HSPA network architecture for packet switched core [15].

- **Mobile Station or User Equipment (UE):** The mobile station or user equipment (UE) can either be a hand-held mobile device or a PC (desktop or laptop) [16]. The function of user equipment (UE) is to provide interface between the user and the radio network [16].

- **Radio Access Network:** The radio access network shown in Figure 2.1 is called the UMTS Terrestrial Radio Access Network (UTRAN) [16]. The function of UTRAN is to provide radio access to UE [16]. The radio network access can be circuit switched or packet switched [16]. The architecture of radio access networks consists of base station and Radio Network Controller (RNC) [16]. Many base stations may be connected to a single RNC [16].

  The main function of RNC is to act as a control node [16]. The base station is called Node B in UTRAN [16]. Most of the network functionality is contained in the RNC [16].

  All user and control plane protocols in the UTRAN are terminated in the Radio network controller [16]. The functions of Radio network controller include radio resource management, segmentation/concatenation and retransmissions etc [16].

  The user equipment and Node B communicate through a wireless channel [16]. The Node B and Radio Network Controller (RNC) are connected through a wired channel [16].

- **Packet switched Core Network (CN):** The packet switched domain consists of Serving GPRS Support Node (SGSN) and Gateway GPRS Support Node (GGSN) [16]. The packet switched domain is used to provide connectivity toward the packet switched networks [16]. The functions of the core network include switching and routing of packets [17].

HSPA user plane protocol architecture

All information sent and received by the user, such as the encoded video in a video call, are transported via the user plane [15]. The protocol architecture of the user plane in HSPA is shown in Figure 2.2. A brief description of various protocol layers in the user plane is given below:

- **Packet Data Convergence Protocol (PDCP):** The PDCP is only used for the packet switched domain services [15]. The functions of Packet Data Convergence Protocol (PDCP) include header compression and decompression of RTP/UDP/IP packets [15]. In 3G networks, RObust Header Compression (ROHC) [18] is used. Header compression is done in the Radio Network Controller (RNC) and the user equipment, for HSPA system [16]. In LTE system, header compression is located in the eNB and the user equipment.

  Another function of PDCP is transfer of user data [19]. Thus PDCP is responsible for transfer of data between users of PDCP services [19].

- **Radio Link Control (RLC):** The Radio Link Control (RLC) protocol is responsible for the segmentation and reassembly, concatenation and padding of the upper layer Internet Protocol (IP) packets into Radio Link Control Protocol Data Unit (RLC-PDU) [20]. RLC-PDU is the data unit of WCDMA transmission [20]. The RLC layer provides in-sequence delivery of upper layer IP packets [17].

  RLC layer can operate in three different modes [17]. These are transparent mode, acknowledged mode and unacknowledged mode [17]. In unacknowledged mode, RLC layer retransmissions do not take place [17].

  Unacknowledged mode is used with delay constrained applications such as video telephony [17]. RLC level retransmissions cause delay variation. Delay constrained applications cannot tolerate delay variations [17].

- **Medium Access Control (MAC):** The Medium Access Protocol (MAC) works at the user equipment and the radio network controller [17]. MAC layer maps the logical channels to transport channels [17].

- **WCDMA L1 or physical (PHY):** The Physical Layer (PHY) maps the transport channels into physical channels [17]. The actual transmission takes place at the physical layer [17]. The functions of physical layer include modulation, demodulation, interleaving, hybrid ARQ and soft handover [17].

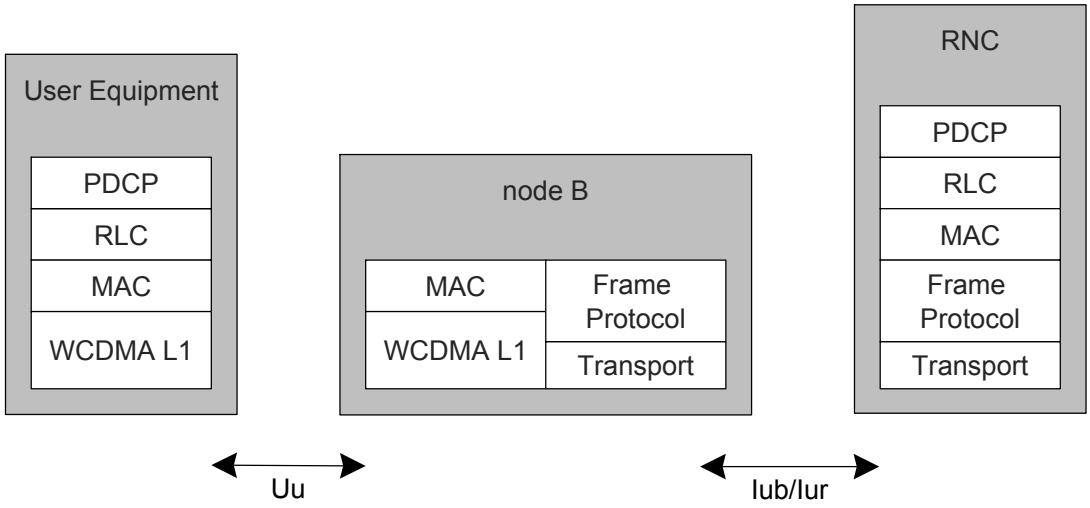


Figure 2.2: HSPA packet switched user plane protocol architecture [17].

2.1.2 3GPP Long Term Evolution (LTE)

3GPP's LTE standard is designed to provide higher data rate and lower latency than the HSPA standard [4]. LTE consists of only packet switched domain [4]. In LTE, the latency of a packet from the network to the user equipment may be less than 10 ms [4].

LTE network architecture

The LTE network architecture consists of evolved Node B (eNB) and mobility management entity/gateway (MME/GW) [4].

The LTE network architecture is given in Figure 2.3. All the network interfaces in LTE architecture are based on IP protocols [4]. The eNBs are connected to each other through an X2 interface [4]. eNBs are connected to the MME/GW through an S1 interface [4]. One MME/GW may be connected to many eNBs [4]. The description of various network elements is given below:

- **Evolved Node B (eNB):** The functions of Node B and Radio network controller, of HSPA, are implemented in the eNB [4]. The functions of eNB include header compression and reliable delivery of packets [4].

- **Mobility Management Entity/Gateway (MME/GW):** There are two logical gateway elements [4]. These are serving gateway (S-GW) and the packet data network gateway (P-GW) [4]. The packets from the user equipment are sent and received to and from the S-GW via eNB [4].

  The P-GW interfaces with external packet switched networks, for example, the Internet [4]. The functions of the P-GW include address allocation, routing and packet filtering [4].

  The functionality of the MME is limited to signalling only [4]. User data packets do not pass through the MME [4].

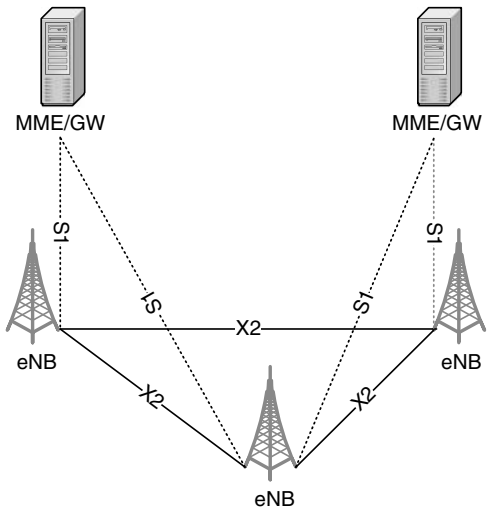


Figure 2.3: LTE network architecture [4].

LTE user plane protocol architecture

The user plane protocol architecture is given in Figure 2.4. The packet data convergence protocol (PDCP) and radio link control (RLC) layers operate between UE and eNB [4].

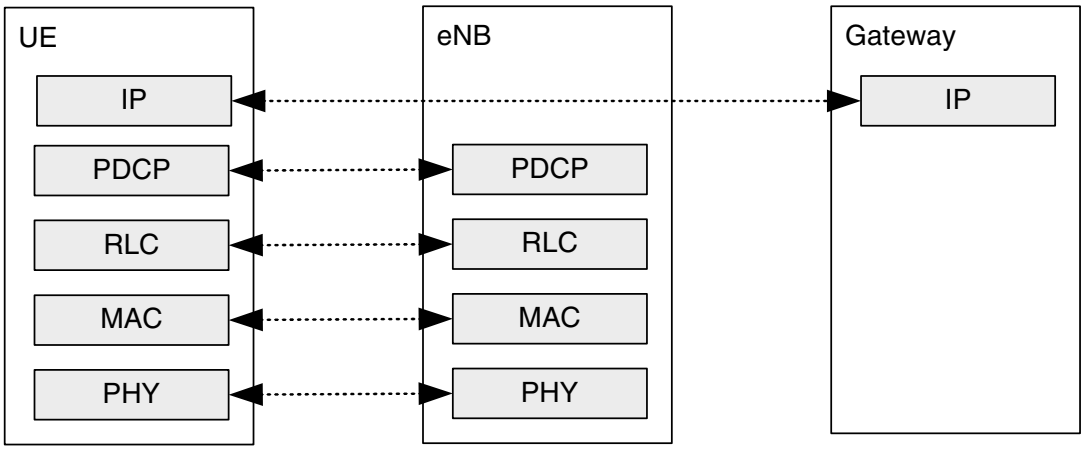


Figure 2.4: LTE user plane protocol architecture [4].

2.1.3 Video telephony user plane protocol stack over 3GPP networks

3GPP standard has specified Session Initiation Protocol (SIP) for signaling and control [21]. The media transport protocol is Real-time Transport Protocol (RTP) [21]. The user plane protocol stack for packet switched video telephony, as specified in the 3GPP standard [21], is shown in Figure 2.5. Video telephony is classified as a conversational application [20]. The video codecs specified by 3GPP for video telephony are H.263 and H.264 baseline profile [22]. In this work, H.264 video codec is used. Compressed video data is carried in RTP/UDP/IP packets. RTP/UDP/IP packets are then transported over the 3G network. For the application layer feedback, RTCP protocol is used [21]. RTCP packets are carried over UDP/IP protocol. A brief description of RTP and UDP is given below:

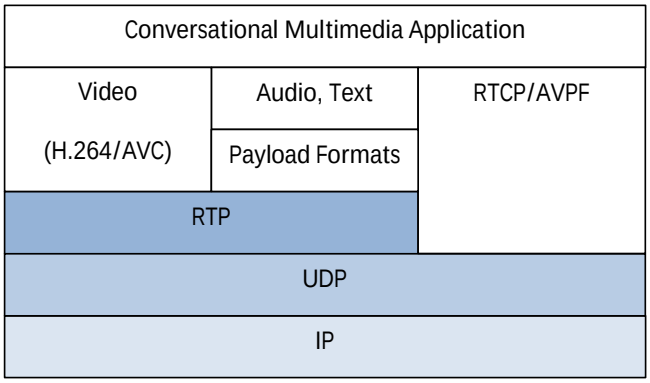


Figure 2.5: User plane protocol stack for 3G packet switched conversational multimedia terminal [21].

- **Real-Time Transport Protocol (RTP):** Real-Time Transport Protocol (RTP) [23] provides packet ordering using sequence number [16]. It also provides playback control of real-time multimedia [16]. It is not responsible for timely or error free delivery [16]. RTP protocol contains information, that helps the receiver to present the data in real-time [16].

- **User Datagram Protocol (UDP):** User Datagram Protocol (UDP) [7] is a simple and connectionless protocol. For UDP packets, no acknowledgment is needed compared to Transmission Control Protocol (TCP). UDP is an unreliable protocol and offers best effort delivery. However, it is suitable for low delay real-time applications such as video telephony.

The layer 1-2 user plane protocol stack for video telephony application over 3G networks is shown in Figure 2.6. The video telephony application is assigned a Quality of Service (QoS) class by the 3G network [20]. Video telephony is classified as conversational QoS or traffic class by the 3G network [20]. Conversational class is most stringent in terms of the delay requirements [20]. The IP packets of the video application are mapped on to 3G network for transport. The mapping of IP packets onto 3G network is done in layer 2 of the 3G network [20]. Layer 2 of 3G network contains 3 protocols [20]. These are Packet Data Convergence Protocol (PDCP) protocol, Radio Link Control (RLC) protocol and Medium Access Control (MAC) protocol. A brief description and functionality of each protocol has been given in the previous sections.

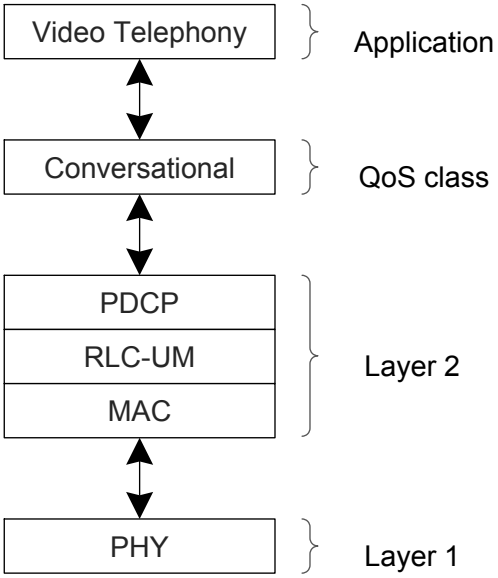


Figure 2.6: 3G layer 1-2 protocol stack for video telephony [20].

3G Quality of Service (QoS) concept and conversational traffic class

The 3G standard provides QoS mechanism [4]. The QoS mechanism differentiates between packet flows (traffic) [4]. QoS control is present between the user equipment and the 3G gateway node [4]. The QoS traffic class is indicated in the PDCP context [6]. Video telephony is classified as conversational traffic class. Conversational traffic class has most stringent delay requirements. Conversational traffic class has the following characteristics [6]:

- Preservation of time relation (variation) between information entities of the stream [6]
- Conversational pattern (stringent and low delay) [6]
- Guaranteed bit rate [6]

The bit rate of the radio bearer is characterized by the size of the RLC-PDU and the Transmission Time Interval (TTI). In this work, one RLC-PDU is sent at a given TTI. The minimum time between sending of two consecutive RLC-PDU is called TTI. The bit rate of the radio bearer is calculated as:

$$\text{Bit rate of radio bearer} = \frac{\text{RLC-PDU size in bits}}{\text{Transmission Time Interval}}$$

2.1.4 Packetization through the 3GPP protocol stack

The packetization through the 3GPP protocol stack is shown in Figure 2.7. At the PDCP layer, 40 byte RTP/UDP/IP header is compressed to 5 bytes using Robust Header Compression (RoHC) [19]. Also, a 2 byte PDCP header is added to each packet to make it a PDCP packet [19]. The PDCP packet is handed over to the Radio Link Control (RLC) layer where it becomes the Radio Link Control Service Data Unit (RLC-SDU) [2]. RLC-SDU is mapped onto Radio Link Control Protocol data unit (RLC-PDU). RLC-PDU is the data transmission unit over the wireless channel [2].

The mapping of RLC-SDU to RLC-PDU is done such that if a RLC-PDU contains the last byte of a RLC-SDU and not all the bytes in the fixed sized RLC-PDU have been used then the first byte of the next RLC-SDU is concatenated with the last byte of the previous RLC-SDU in the same RLC-PDU. Thus, a RLC-PDU may contain data of more than one RLC-SDU.

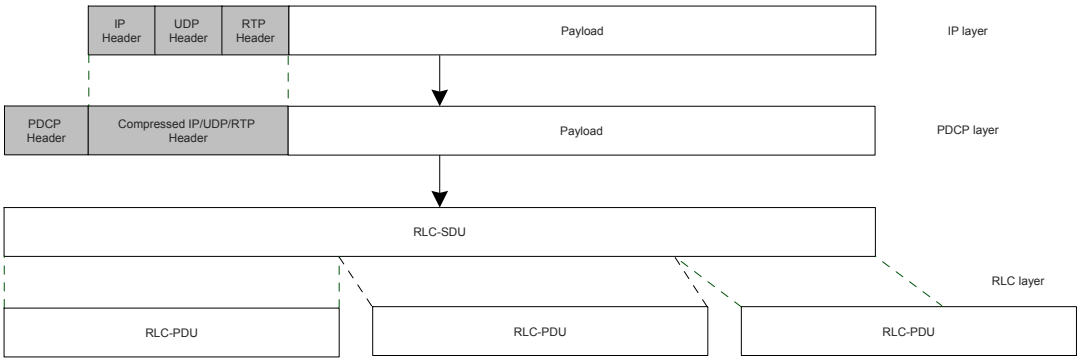


Figure 2.7: Packetization through 3GPP protocol stack [2].

2.2 H.264/AVC video coding standard

Video coding is used to reduce the size of the video data. The reduction in video data size is achieved by reducing redundant video data. This results in compression and reduces bandwidth usage. H.264/AVC is the state of the art video codec [24]. Compared to previous video codecs such as MPEG-2, MPEG-4 Part 2 and H.263, H.264/AVC has improved compression efficiency, error resilience features and network friendliness [24]. However, the improved compression efficiency comes at a cost of higher encoding and decoding complexity compared to the previous video codecs [24].

3GPP has specified H.264 baseline profile as one of the video codec for video telephony [22]. H.264 baseline profile include low encoding and decoding complexity, error resilience and network friendliness. These features make H.264 baseline profile suitable for mobile video telephony.

In H.264, nearly all the computations and calculations can be done with additions and shifts [24]. Thus, H.264 codec is suitable for hardware implementation [24]. Video coding is a trade-off between compression efficiency, video quality and computational complexity [24].

2.2.1 H.264 Design

H.264 is a block-based video codec [24]. Its compression algorithm is based on motion compensation [24]. H.264 is designed to be flexible, scalable and customizable for different applications [24]. It is efficient for low bit rate video telephony as well as high bit rate high definition broadcast applications [24].

H.264 design is divided into Video Coding Layer (VCL) and Network Abstraction Layer (NAL) [24].

The H.264 architectural design is shown in Figure 2.8.

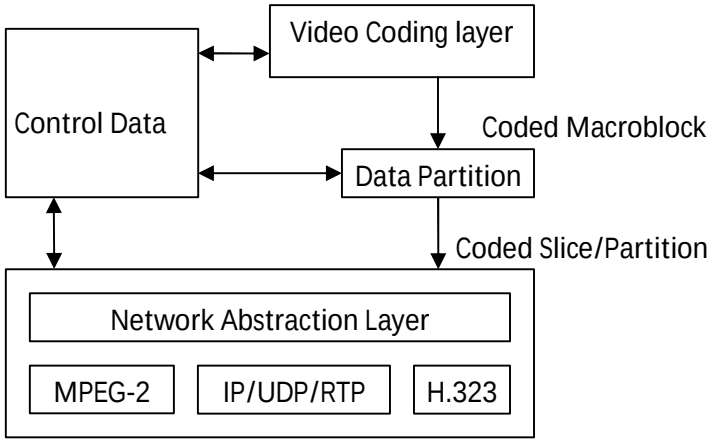


Figure 2.8: H.264 architecture [24].

- **Video Coding Layer (VCL):** The Video Coding Layer (VCL) uses signal pro-

cessing to achieve video compression [24]. The VCL design is based on block-based hybrid video coding [24].

- **Network Abstraction Layer (NAL):** The Network Abstraction Layer (NAL) encapsulates the encoded video bit-stream such that it is convenient for storage in byte format or it can be transported over a packet based network [24].

2.2.2 Terminology

The understanding of the following terminology is helpful to get an insight into the working of H.264 encoding and decoding process.

Sampling and YUV Format

H.264 uses YUV format. In YUV format, a pixel has a luma (Y) and two chroma (U,V) components. The luma component represents the brightness of the pixel while the two chroma components represents the colour of the pixel. In YUV 4:2:0 format, every pixel has its distinct luma component while every 2×2 block of pixels share the same chroma (U, V) components [25]. The reason for higher sampling of luma component is that the human visual system is more sensitive to brightness (luma), than colour (chroma) [25].

Macroblocks

H.264 divides each video frame into *macroblocks*. Macroblocks are blocks of 16×16 pixels [25]. The processing unit of video compression is a Macroblock [25]. In YUV 4:2:0 format, each macroblock has 16×16 block of luma samples and two 8×8 blocks of chroma samples [25]. During video encoding and decoding, macroblocks are processed in raster scan order [25]. The macroblocks can be sub divided into blocks of up to 4×4 pixels [25]. A 16×16 macroblock is shown in Figure 2.9.

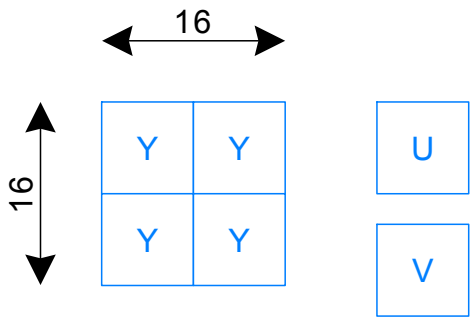


Figure 2.9: A 16 × 16 macroblock showing YUV 4:2:0 sampling scheme [25].

Slices

During compression, a video frame may be split into slices. A slice is a group of macroblocks. A slice can be encoded and decoded independently of other slices in a frame. Typically, one encoded slice is packed in one NAL unit and transmitted in one packet. The advantage of using slices is that if a packet is lost then only a part of a video frame is lost instead of the whole frame. Slices decrease compression efficiency because inter prediction can only be done between macroblocks belonging to the same slice [25]. In H.264 standard, following slice types are defined [26]:

- **I slices** I or Intra slices contain macroblocks that are encoded using macroblocks in the same slice of the same frame. All the slices in the first frame of a video sequence are encoded as I slice [26].
- **P slices** P or Predicted slices contain macroblocks that are encoded using macroblocks from a previous frame which has already been encoded and decoded. Some macroblocks in a P slice may be encoded in intra mode.
- **B slices** B or Bi-Directional Predicted slices contain macroblocks that are encoded using macroblocks in a previous or future playback frame. A B slice is decoded after the previous or future reference frames are decoded.
- **SI and SP slices** SI and SP or Switching slices may be used for switching between two different H.264 video streams.

2.2.3 The H.264 video codec

The H.264 encoder is shown in Figure 2.10. The input to the H.264 encoder is video frames in YUV format.

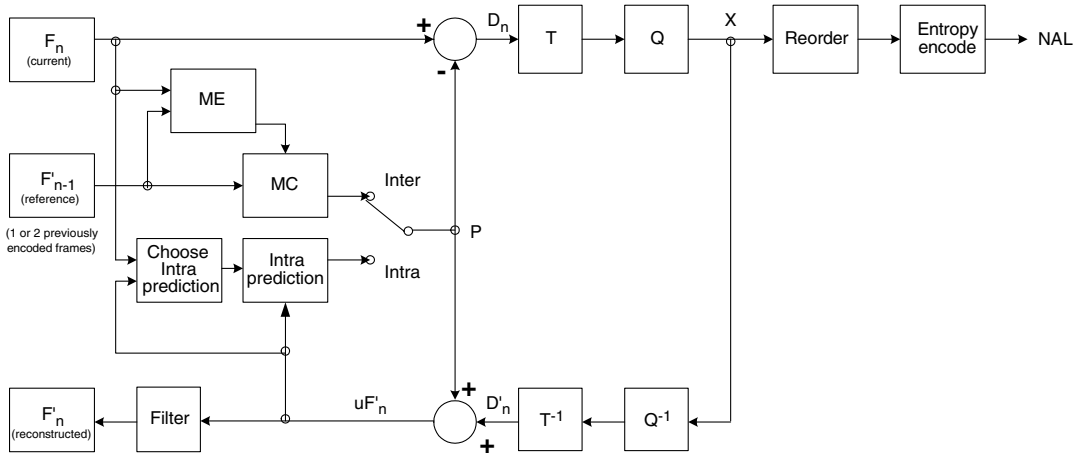


Figure 2.10: H.264 encoder [25].

- **Encoder:** A macroblock is the processing unit in the video encoder [25]. Each macroblock can be encoded in either inter or intra mode [25]. A prediction P is formed for each block in the macroblock. The prediction P is formed from the reconstructed (decoded and filtered) pixels [25]. In Intra mode, the prediction P is formed from unfiltered reconstructed pixels uF'_n of the same slice [25]. In Inter mode, P is formed by motion compensation from reconstructed (decoded and filtered) pixels in past or future frames(s) (F'_{n-1}) [25]. A residual block D_n is formed by subtracting the Prediction P from the current block [25]. The residual block is transformed and quantized. The quantized transform coefficients are reordered and entropy coded [25]. The entropy coded coefficients, prediction modes, and motion vector information form the compressed bit stream [25]. The compressed bit stream is encapsulated by the Network Abstraction Layer (NAL) to produce NAL units. The NAL units can be stored or transmitted over a packet based network. In encoder, the macroblocks are also decoded to form reconstructed pixels. These reconstructed pixels are used for inter prediction [25].

- **Decoder:** The H.264 Decoder is shown in Figure 2.11. The decoder receives the compressed bit stream in the form of NAL units [25]. The NAL units are entropy decoded to recover the coefficients [25]. The coefficients are scaled and inverse

transformed [25]. This is now a residual block D'_n [25]. The decoder adds the residual block to the prediction to produce uF'_n [25]. This is filtered to produce decoded block F'_n [25].

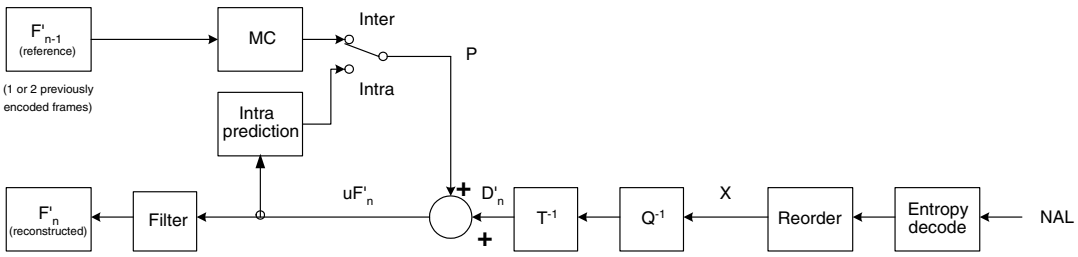


Figure 2.11: H.264 decoder [25].

2.2.4 Prediction overview

H.264 uses intra and inter prediction. Prediction is formed from the samples that are already reconstructed (decoded). A brief description of intra and inter prediction is given in the following sections.

Intra prediction

In Intra prediction, the prediction is formed from the reconstructed samples belonging to the same slice [25]. The neighboring samples which are left and/or above the current block are used for intra prediction [25]. For the luma component, intra prediction can be done on 4×4 block or 16×16 block (macroblock) [25].

There are nine 4×4 prediction modes, shown in Figure 2.12, and four 16×16 modes given in Figure 2.13 [25]. The cost of each mode is calculated and the mode that results in least cost and most closely resembles the original block is selected as the coding mode [25].

For the chroma component, 8×8 block is used. The same intra prediction mode is used for both chroma components. Intra coding generates more bits than inter coding.

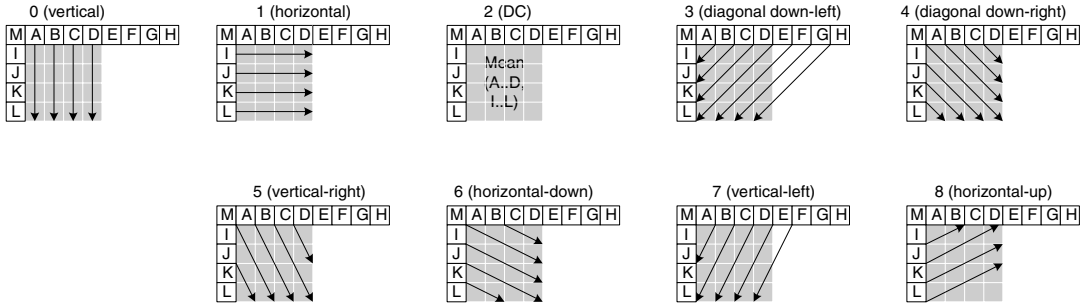


Figure 2.12: Intra 4×4 luma prediction modes [25].

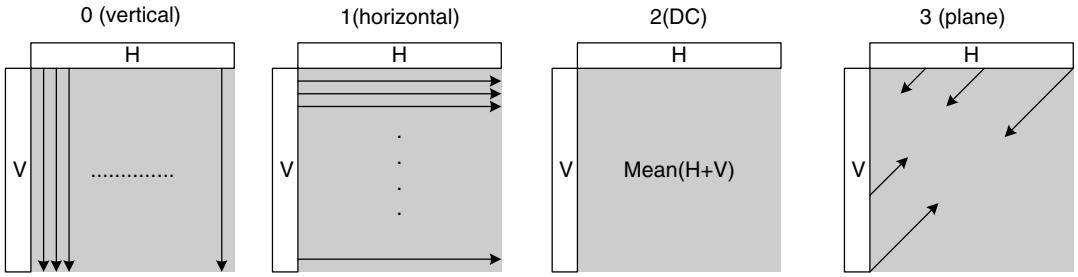


Figure 2.13: Intra 16×16 luma modes [25].

Inter prediction

The data in a video frame may have high correlation to the data in a previous frame. Inter prediction exploits this fact to reduce the redundancy. In intra prediction, the encoder looks for a macroblock (or submacroblock) in a previous frame, which is very similar to the current macroblock (or submacroblock). The displacement between macroblock or submacroblock in a current frame and its closest match in a previous frame is denoted by a motion vector. This process is known as motion estimation. The decoder use motion vector information to create the original macroblock (or submacroblock). This process is known as motion compensation.

H.264 uses half and quarter pixel motion vectors to achieve better accuracy. For motion estimation and compensation, 16×16 block can be sub-partitioned into as small as 4×4 submacroblock The sub-macroblock partitions are shown in Figure 2.14 [25].

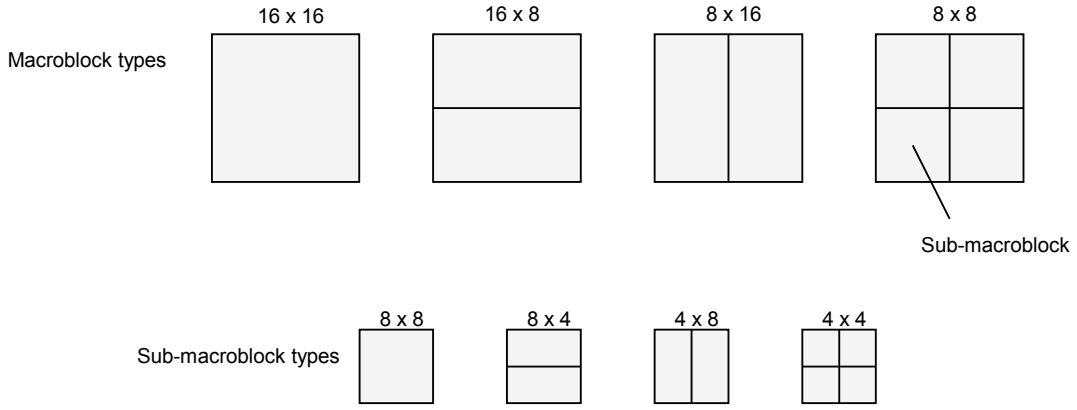


Figure 2.14: Macroblock and sub-macroblock partitions [25].

2.2.5 Deblocking filter

Due to the use of blocks in video coding, visual artefacts are produced along the block boundaries. These visual artefacts are due to the sudden discontinuities at block boundaries. H.264 uses deblocking filter to smooth out the discontinuities along the block boundaries. This results in improved visual quality. In H.264, the deblocking filter is included in the encoding loop. Deblocking filter is applied after decoding in the encoder and decoder. Deblocking filter is applied on a 4×4 block [25].

2.2.6 Transform

The transform in H.264 is DCT like, 4×4 transform. The transform is 16 bit integer transform. Integer transform avoids encoder decoder mismatches. The transform can be implemented with 16 bit addition and shifts only. After transform, the data is converted into frequency domain. In frequency domain, the data is scaled and quantized to achieve compression [25].

2.2.7 Quantization

After applying transform, the coefficients are quantized. H.264 uses a scalar quantizer. The quantization process is controlled by a parameter called Quantization Parameter (QP). There are 52 values for Quantization Parameter. Higher values of QP result in lower quantization (and lower bit rate) and vice-versa. Quantization Parameter is used by the encoder to achieve flexible trade-off between compression efficiency and visual quality [25].

2.2.8 Entropy Coding

The entropy coder assigns code words to the data symbols. More frequent data symbols are assigned shorter codewords and less occurring data symbols are assigned longer codewords. In H.264, there are two types of entropy coding. These are Context Adaptive Variable Length Coding (CAVLC) and Context Adaptive Binary Arithmetic Coding (CABAC). Any one type of entropy coding may be used at a time. CABAC achieves better compression efficiency than CAVLC at a cost of higher computational complexity [25].

2.2.9 Network Abstraction Layer (NAL)

The Network Abstraction Layer (NAL) of H.264 video codec encapsulates the encoded video bit stream into NAL units. NAL units are convenient for byte storage or transport over a packet oriented network. Each NAL unit contains an integer number of bytes. An NAL unit may contain compressed video data or header information related to the video data or sequence. The header information contains useful information for the decoder.

The size of NAL unit header is 1 byte. Each NAL unit can be independently decoded by the H.264 decoder. The start of the NAL unit is identified by the start code prefix. In packet oriented networks, one packet typically contains one NAL unit. For packet oriented networks, start code prefix is not used [24].

2.2.10 H.264 error resilience features

H.264 standard contains many error resilience features to combat against data error and/or losses [24]. Use of error resilience features may result in better video quality. A brief description of H.264 error resilience features is given below.

- **Parameter set structure:** Important header information can be transported reliably using parameter set structure [24].
- **NAL unit syntax structure:** NAL units allows the video data to be stored or carried over a network in packets. NAL units are independently decodeable. Thus, H.264 decoding can continue if some NAL units are lost [24].
- **Flexible slice size:** The flexible slice size allows the size of the network packet to be flexible. Since each packet typically contains one slice, this may improve

robustness during network transmission [24].

- **Flexible macroblock ordering (FMO):** Flexible macroblock ordering partitions the macroblocks in a frame into spatial regions called slice groups. The shape of the slice group can be chosen to improve visual quality in a case when some slices are lost [24].
- **Arbitrary slice ordering (ASO):** Each slice in H.264 standard can be independently decoded. The order of transmission and decoding of slices can be changed to achieve better error resilience [24].
- **Redundant pictures:** Redundant pictures are redundant representations of region of pictures. In case the primary representation of the picture has been lost during transmission, the redundant pictures can be used to achieve better video quality [24].
- **Data partitioning:** Data partition separates encoded video data into three types of partitions. Different levels of protection can be used for each type of partition. Key information may be carried in partition with higher level of protection [24].
- **SP/SI synchronization/switching pictures:** The SP/SI pictures allows switching and synchronization between video streams meant for different decoders. This eliminates the need for I frame stream for switching and synchronization between different decoders [24].
- **Multiple reference pictures:** H.264 standard allows prediction from many reference frames. This feature can be used for compression efficiency as well as error resilience [27].

2.3 Effect of network architecture and codec parameters on video quality

In 3G network architecture, there can be one or two wireless links. The major source of errors and packet losses is the wireless channel. Packet losses may be due to shadowing, interference, multipath fading, refraction, reflection or attenuation. A 3G network architecture with 2 wireless links is more susceptible to packet losses than a 3G architecture with one wireless link. So stronger error resilience techniques are needed to provide better video quality on a 3G network with two wireless links.

In 3G networks, the each user is assigned a bit rate. Assigning higher bit rate to each user results in better video quality and improve user experience but it also decreases network capacity. For a given bit rate, increasing frame rate makes the video smoother but also decreases the quality of each video frame. Increasing the video frame size results in video quality degradation of each frame. Thus, there is a trade-off between bit rate, frame rate, frame size and video quality.

2.4 Rateless codes

Rateless codes are a class of sparse-graph codes [9]. They are erasure correction codes. The examples of rateless codes are LT and Raptor codes. Raptor codes are an extension and improvement to LT codes. Rateless codes are also known as Digital Fountain codes [28]. They have many desirable properties such as [9]:

- They have lower encoding and decoding complexity compared to traditional error correcting codes such as Reed-Solomon codes [9]. The encoding complexity of LT code is $O(\ln k)$ for each encoding symbol. The decoding complexity of LT codes is $O(k \ln k)$. k is the number of source or information symbols.
- Their redundancy does not need to be fixed in advance. Unlimited number of encoded symbols can be generated on the fly when needed [9].
- The source block can be of any size [9].

For successful decoding, LT decoder needs to receive more than k symbols. The number of symbols that are more than k and needed for decoding success is called the LT code overhead.

Digital fountain codes are based on graphs, that are bipartite [29]. In bipartite graphs, there are two disjoint sets of vertices [29]. One set of vertices contains the source symbols [29]. Other set of vertices contains the encoded symbols [29]. Vertices in one set can only be connected to vertices in the other set [29]. A connection between element of two sets is called an edge [29]. The operation on symbols during encoding and decoding is the exclusive-OR $\oplus$ operation [29].

The encoding and decoding of rateless codes are governed by degree distributions [29]. There are two degree distributions [29]. These are Ideal Soliton distribution and Robust Soliton distribution [29]. Robust Soliton distribution is more practical

distribution [29]. Details of Ideal Soliton distribution and Robust Soliton distribution are given in [9].

2.4.1 LT code

The input to the LT encoder is a set of k source symbols [29]. These symbols can be bits or bytes [29]. The LT encoder generates a sequence of encoded symbols as output [29]. The number of encoded symbols can be infinite [29]. The alphabet of the source and encoded symbols is the same [29]. The computation of each encoded symbol is independent of the other encoded symbols [29].

For successful LT decoding, a little more than k encoded symbols are needed as input [29]. The probability of successful LT decoding (recovery of all source symbols) is 1 - ϵ [29]. ϵ is a positive number [29].

LT Encoding

The input to the LT encoder are k source symbols [29]. A degree distribution $\Omega(x)$ is computed for each of the k source symbols [29]. An encoded symbol is generated as follows [29]:

1. Select randomly a degree from the degree distribution $\Omega(x)$.
2. Select uniformly at random distinct source symbols. The number of source symbols selected is equal to the selected degree in step 1.
3. Compute the bitwise exclusive-OR of the selected source symbols. The result of the bitwise exclusive-OR is the value of the encoded symbol.

This process is repeated for all the encoded symbols [29].

LT Decoding

For LT decoding, the decoder needs to know the degree of each encoded symbol and the graph (connectivity information) between the source and the encoded symbols [29]. One way of achieving this is to use a pseudo random number generator with same seed at the LT encoder and decoder [29]. The LT decoding is done as follows [29]:

1. Find an encoded symbol with degree 1. An encoded symbol with degree 1 means that it is connected to only one source symbol. If an encoded symbol with degree 1 cannot be found then decoding is stopped and cannot continue.

2. Find the source symbol which is connected to the encoded symbol of degree 1.

3. Set the value of the source symbol in step 2 to the value of the encoded symbol.

4. Find all the encoded symbols that are connected to the source symbol in step 2. This excludes the encoded symbol in step 1.

5. For each connected encoded symbol:

   (a) Find the bitwise exclusive-OR of the encoded symbol and the source symbol.

   (b) Set the value of the encoded symbol to the result of the bitwise exclusive-OR operation in step (a).

   (c) Remove the edge connecting the encoded symbol and the source symbol.

6. Repeat Step 1 until all k source symbols are recovered.

The decoding efficiency of traditional LT code is improved by using a method presented in [30]. This method is called Block Duplication method. The decoding efficiency of LT code is improved by virtually increasing the number of source symbols. This results in increased number of edges in the LT code graph, hence improving the decoding efficiency. The increased number of edges results in higher encoding and decoding complexity. In this research work, a real software based implementation of LT codes with block duplication method is used. The details of how LT encoder and decoder are used in this work are given in Section 4.2.1.

2.4.2 Raptor code

Raptor codes [10] are an extension of LT codes. Raptor code has lower encoding and decoding complexity compared to LT code [10]. The complexity improvement is obtained by reducing the number of edges in the code graph [10]. Since the number of edges in an LT code depends on the underlying distribution $\Omega(x)$, the idea is to choose a distribution that generates a small number of edges [10]. However, by using such a distribution, the decoding probability of the code is decreased [10]. To overcome this problem, a traditional block code called precode is concatenated before the LT code [10]. Thus, Raptor code is a concatenation of a precode and an Lt code [10]. The encoding complexity of Raptor code is $O(1)$ for each encoding symbol. The decoding complexity of Raptor code is $O(k)$.

The pros and cons for using Rateless codes for mobile video telephony is given below:

- The redundancy of Rateless codes does not need to be fixed in advance. Hence, any number of encoded symbols can be produced on the fly. This makes Rateless codes suitable for real-time low delay video telephony.
- The encoding and decoding complexity of Rateless codes is lower than traditional error correction codes. This makes Rateless codes suitable for implementation on complexity constrained hand-held devices.
- Rateless codes require more than k (information or source) symbols to be received for successful decoding. Thus, there is a overhead in terms of bandwidth.
- The decoding performance of Rateless codes depends on the source block size. A Larger source block size results in better decoding performance of Rateless codes. However, larger source block size also increases the encoding and decoding complexity.

2.5 Network simulators

Many network simulators are tested to simulate the 3G wireless channel and to select the network simulator that is suitable for this research. A brief description of the tested network simulators is given below.

2.5.1 Ns-2

Ns-2 [31] is an event driven network simulator. There are third party Ns-2 modules for 3G UMTS. These include EURANE [32] and NS-MIRACLE [33]. However, these modules are old and not updated to the latest version of HSPA and LTE standards. In order to use Ns-2 for this research, several modules in Ns-2 needed to be updated. However, this was found to be time consuming and not enough documentation was available to make these changes. For these reasons, it was not used for this research.

2.5.2 Seawind

Seawind [34] is a real-time wireless network emulator. Seawind emulator allows examination of data transfers over wireless networks such as GSM and GPRS. At the time of this research, Seawind does not include 3G HSPA and LTE cellular network functionality. To add 3G HSPA and LTE functionality, a lot of modifications were needed. The documentation of Seawind was not available to make these changes. Hence, Seawind was not used for this research.

2.5.3 NCTUNS

NCTUNS is a real-time network emulator and simulator [35]. At the time of this research NCTUNS does not include 3G HSPA and LTE network module. For this reason, it was not used for this research.

2.5.4 OPNET Modeler

OPNET is a commercial simulator [36]. OPNET supports many wireless channels and protocols. However, it was found to be complex and time consuming to integrate OPNET with H.264 video encoder and decoder.

2.5.5 3GPP SA4 channel simulator

This is a 3G wireless channel simulator. This simulator is developed by the 3GPP [37]. This channel simulator supports packetization of RTP/UDP/IP according to the 3G standard. This channel simulator also supports conversion of RTP/UDP/IP packets into RLC-SDUs and mapping RLC-SDUs onto RLC-PDUs. This mapping is done according to the 3G standard. The input to the simulator is RTP packets in RTPdump format. This channel simulator supports simulation of RLC-PDU transmission according to the given bearer parameters. It also supports simulation of RLC-PDU losses, based on real 3G network traces. Loss of RLC-PDUs results in RTP/UDP/IP packet loss, that are partially or fully mapped to these RLC-PDUs. Thus, realistic results are obtained using this channel simulator. The source code is available in C++. The methods IEC1 and IEC2 proposed in [13, 14] are tested on this simulator. In order to compare the results of the proposed method with IEC1 and IEC2 under similar conditions, this channel simulator is used in the first part of this research work. The channel simulator, its manual, traces and video sequences are available from 3GPP [37].

2.5.6 Rayleigh fading wireless channel model

In [38], a Rayleigh fading wireless channel model is proposed. The transport blocks (RLC-PDU) successes/failures can be modeled by a 2-state Markov process. 2-state Markov process has two states: good and bad. In good state, the transport block is assumed to be received correctly. In bad state, the transport block is assumed to be lost. This model for Rayleigh fading is validated in [39] and is widely used, for example, in [40]. The transition matrix M(x) for this model is given as [41]:

$$\mathrm{M}(x) = \begin{pmatrix} p(x) & r(x) \\ s(x) & q(x) \end{pmatrix} = \begin{pmatrix} p & 1-p \\ 1-q & q \end{pmatrix}^{x} \tag{2.1}$$

where p and 1-q are the probabilities that the jth transport block transmission is successful, given that the (j-1)th transport block transmission was successful or unsuccessful, respectively [38]. The steady-state transport block error rate, P_{BL} for this model is given by [38]:

$$P_{\mathrm{BL}} = \frac{1-p}{2-p-q} \tag{2.2}$$

As discussed in [38], for a Rayleigh fading channel with fading margin F, the average transport block error rate P_{BL} is given by:

$$P_{\mathrm{BL}} = 1 - e^{\frac{-1}{F}} \tag{2.3}$$

The Markov parameter (q) is given as [38]:

$$q = 1 - \frac{Q(\theta, \rho\theta) - Q(\rho\theta, \theta)}{e^{\frac{1}{F}} - 1} \tag{2.4}$$

where θ is given by [38]:

$$\theta = \sqrt{\frac{\frac{2}{F}}{1-\rho^2}} \tag{2.5}$$

$Q(..,.)$ is the Marcum-Q function. Marcum-Q function is given by [42]:

$$Q(x,y) = \int_{y}^{\infty} e^{-\frac{x^2+w^2}{2}} I_0(xw)wdw \tag{2.6}$$

The correlation coefficient of two successive samples is ρ [38]. It is given by [38]:

$$\rho = J_0(2\pi f_d T) \tag{2.7}$$

where T is the time between successive samples [38]. T is in seconds. For 3G networks, this is equal to the Transmission Time Interval (TTI). f_d is the Doppler frequency [38]. f_d is given by [38]:

$$f_d = \frac{\text{mobile velocity}}{\text{carrier wavelength}} \tag{2.8}$$

$J_0(.)$ is the Bessel function of first kind and zeroth order [38].

The relationship between steady-state block error rate P_{BL} and Markov parameter q is given by [38]:

$$q = 1 - \frac{(1 - P_{\text{BL}}) \times (Q(\theta, \rho\theta) - Q(\rho\theta, \theta))}{P_{\text{BL}}} \tag{2.9}$$

where

$$\theta = \sqrt{\frac{-2\log(1 - P_{\text{BL}})}{1 - J_0^2(2\pi f_d T)}} \tag{2.10}$$

In [43], it was shown that the relationship between Markov parameters are independent of the communication system parameters. These parameters include block length and modulation scheme. This model can be used for a wide variety of communication systems [43]. Using this model, the analysis of upper layer protocols is simplified [43]. This model is used in the later part of this research work.

2.6 Video quality metrics

The video quality can be measured using objective or subjective metrics. Objective video quality is measured using a mathematical formula. Objective video quality metrics include Peak Signal to Noise Ratio (PSNR), Percentage of Degraded Video Duration (PDVD), Structural Similarity (SSIM) index [44, 45] etc. In [46], the video quality is measured using Noise Quality Measure (NQM) and Distortion Measure (DM) [46]. In [47] Moving Pictures Quality Metric (MPQM) has been proposed. In [48], Video Quality Metric (VQM) has been proposed. Most objective video quality metrics require original video sequence. An overview and validation of different objective video quality metrics has been give in [49].

Subjective video quality is based on human perception. Example of Subjective video quality metric is Mean Opinion Score (MOS).

The objective video quality metrics used in this work are PSNR and PDVD. The average PSNR over whole video sequence is calculated. PDVD metric is useful to calculate the total percentage of degraded frames in a video sequence. The methods IEC1 and IEC2 proposed in [13, 14] are evaluated using PSNR and PDVD. In order to compare the proposed method with IEC1 and IEC2, PSNR and PDVD metrics are used in this work.

2.6.1 Peak Signal to Noise Ratio

Peak signal to noise ratio is widely used video quality metric. In PSNR, the peak signal to noise ratio is computed between two images (frames) [50]. PSNR is computed in decibels. Higher value of PSNR means better video quality. For PSNR calculation, the original video sequence is needed. PSNR is computed as [50]:

$$PSNR = 10 \log_{10} \left(\frac{R^2}{MSE} \right) \tag{2.11}$$

In equation 2.11, R is the maximum value of the pixel data type [50]. For 8 bit integer pixel data type, the value of R is 255 [50]. In this work, the PSNR is calculated from the luma component only.

Mean Squared Error (MSE) is the sum of the squared error between the original and the compressed image [50]. MSE is given by [50]:

$$MSE = \frac{\sum_{M,N} [I_1(m,n) - I_2(m,n)]^2}{M \times N} \tag{2.12}$$

In equation 2.12, M and N are the number of rows and columns in the input images, respectively [50]. Lower value of MSE means lower error and better image quality [50].

2.6.2 Percentage of Degraded Video Duration

PDVD is an objective video quality evaluation metric. PDVD is defined as the percentage of frames out of the total frames whose PSNR is 2 dB worse than PSNR of the corresponding reconstructed frames [51]. This metric computation requires three sequences, the original video sequence, the reconstructed video sequence and the received video sequence. Only the luma component of the video signal is used for PDVD calculation [51].

PDVD is calculated as follows [51]:

$$PDVD = \frac{\sum_{i=1}^{N} f(\hat{d}_i, \tilde{d}_i)}{N}\% \tag{2.13}$$

$$f(\hat{d}_i, \tilde{d}_i) = 1 \text{ if } (\hat{d}_i - \tilde{d}_i) > 2 \text{ dB , 0 otherwise}$$

where $\hat{d}_i$ is the reconstructed PSNR of ith frame, and $\tilde{d}_i$ is the decoded PSNR of the corresponding frame for a video sequence with N frames. Lower PDVD means fewer number of degraded frames.

Chapter 3

Related work

A comprehensive review of error resilience techniques for packet switched video transmission is given in [8, 52]. A review of recent error resilience techniques for packet switched mobile video telephony is given in [53].

A brief description and limitations of various error resilient techniques that have been proposed for packet switched video telephony or real-time video communication are given in the following sections.

3.1 Forward Error Correction (FEC)

Recent forward error correction based error resilient techniques for real-time video communication have been proposed in [54, 55, 56, 57]. A limitation of these techniques is that they are not tested to work in very low delay video telephony applications. They are mostly suitable for video streaming applications. Also, these techniques do not address the problem of spatio-temporal error propagation.

3.2 Automatic Repeat Request (ARQ)

Automatic repeat request techniques have been proposed in [58, 59, 60, 61, 62, 63].

In [53], NACK based retransmissions are evaluated. The packets are retransmitted only if they satisfy the end-to-end delay criteria.

A limitation of ARQ based techniques is that they do not stop spatio-temporal error

propagation.

3.3 Error concealment

There are many error concealment techniques proposed in the literature. Recently error concealment techniques have been proposed in [64, 65, 66, 67, 68]. Error concealment is a passive error control technique which is invoked after an error or packet loss has been detected at the video decoder. Error concealment is used to conceal the effect of lost data in a video frame. Error concealment does not stop spatio-temporal error propagation.

3.4 Error resilient source coding

3.4.1 Intra refresh schemes

In intra refresh schemes, a frame or macroblocks in a frame are encoded in intra mode. Encoding in intra mode may stop error propagation. Intra encoding requires many times more bits than inter mode. Encoding a whole frame in intra mode may not be feasible for conversational applications due to delay and bandwidth constraints. So, some macroblocks in a frame may be intra coded. Several intra macroblock refresh schemes have been proposed in the literature.

In [69], intra refresh method based on the end-to-end rate distortion model is proposed. The end-to-end distortion model is an improvement to the original Lagrangian rate distortion model [70]. The end-to-end rate distortion model takes into account packet loss ratio and error concealment in the decoder.

In [71], another end-to-end rate distortion optimization strategy has been proposed. This method takes into account the potential error propagation, packet loss ratio and decoder error concealment.

The method in [72] recursively computes the total decoder distortion at pixel level precision. This results in an accurate spatio-temporal error propagation estimate. This estimate is used with rate distortion optimization framework to optimally refresh macroblocks in intra mode. However, due to sub-pixel motion estimation in H.264, the pixel based error propagation estimate may not be accurate for the H.264 video codec.

Isolated region based intra refresh has been proposed in [73]. This method introduces a gradually growing region named isolated region, and the prediction area for those blocks, that are in the isolated region, is restricted.

In [74], attention based adaptive intra refresh technique has been proposed. In this technique, an attention area is determined. The determination of the attention area is based on human perception for better subjective video quality. An attention based end-to-end rate distortion model is derived. The attention area is assigned higher priority for intra coding than non-attention area.

In [75], Group of Blocks (GOB) intra refresh technique has been proposed. The macroblocks are grouped in blocks. The group of blocks are intra refreshed based on the motion content in the macroblocks. High motion macroblocks are grouped together and intra refreshed more often.

In [76], the decision to encode a macroblock in either intra or inter mode is based on content textual information, the frame to frame pixel value variation and the estimated channel loss probability. This decision is taken in advance.

In [77], periodic random intra refresh and motion information based conditional intra refresh methods are proposed. In motion based intra refresh, the blocks with most rapid motion changes are selectively intra refreshed.

In [78], a spiral intra refresh technique has been proposed. In this technique, macroblocks are refreshed in spiral pattern. The intra refresh starts with the macroblocks in the center of the picture and for the next frames, the macroblocks are refreshed in the spiral fashion. Spiral macroblock intra refresh pattern is suitable for human perception and subjective visual quality improvement.

In [79], several intra refresh based error resilience techniques have been proposed. Some intra refresh techniques are combined with Flexible Macroblock Ordering (FMO).

A limitation of intra macroblock refresh techniques is the number and spatial placements of intra macroblocks. Another limitation is how to optimize the trade-off between compression efficiency and suppression of potential errors [71]. Also, in order to achieve rapid error recovery the intra macroblock refresh rate has to be high, which leads to

low video compression efficiency. This is not desirable for bandwidth restricted mobile video communication [27]. Intra refresh schemes may not be effective in environments where the intra coded data itself is lost.

3.4.2 Reference picture selection based techniques

Reference picture selection technique has been proposed in [3]. The reference picture selection is a feedback based error resilience technique. The video decoder informs the encoder of the reference frames which are received correctly or not received correctly [3]. The video encoder, on receiving this feedback information, adapts the encoding operation. This technique can work in Positive Acknowledgment (ACK) or Negative Acknowledgment (NACK) mode [3]. In ACK mode, the encoder uses only correctly received frames as reference [3]. In NACK mode, the encoder does not use corrupted frames as reference [3]. The use of reference picture selection stops error propagation. Since the feedback from the decoder takes time to reach the encoder, the performance of this technique depends on the forward and backward trip time. This technique works better for shorter forward and backward trip time.

In [80], a reference picture selection based error resilience technique is proposed. This scheme utilizes ACK and NACK to adjust the video coding of the encoder in case of packet losses [80]. When using only ACK, the encoder uses and older reference frame, which means that coding efficiency is decreased. In this scheme, a downsampled version of the frame is sent in case of errors, which differentiate it from the standard reference picture selection scheme.

A reference picture selection scheme was proposed in [81] for real-time video communication. In this method, the encoder selects the best reference frame by evaluating the expected reconstruction video quality. The reconstruction video quality is calculated, based on the error feedback and an error propagation model.

In [82], fixed distance reference picture selection is used with proxy based retransmissions. A limitation of this technique is that implementation of proxy server in the base station requires modification in the network hardware, which may be costly and not feasible to implement in practice.

In [83], a retransmission and reference picture selection based approach for end-to-

end video error recovery called RESCU has been proposed. In RESCU, every pth frame is a so called periodic frame that references another periodic frame p frame intervals away [83]. The frames that are in-between two consecutive periodic frames are encoded using previous frame prediction [83]. The error propagation is stopped at the next periodic frame. In case of periodic frame losses, a retransmission request is sent to the encoder and re decoding is used to create an error free periodic frame.

In [84], reference frames are periodically marked as long term reference frames using the encoder memory management control operation (MMCO) [84]. An method is proposed to mark the long term reference frame as reliable reference frame. Reliable reference frame can stop error propagation. In this technique the interval of updating long term reference is an important decision.

In [85], some macroblocks in nth frame uses a reference frame that is n frame interval away [85]. These macroblocks are named as periodic macroblocks [85]. In this technique, it is shown that encoding selected periodic macroblocks will reduce the loss probability of a pixel. The selection of periodic macroblock is based on the distortion expectation of each macroblock in every nth frame [85]. This technique does not adapt to varying packet loss rate in the channel.

In [53], reference picture selection technique is evaluated for video transmission over a 3G network simulator.

A limitation of reference picture selection based techniques is that using older reference frames decreases the video coding efficiency. The performance of reference picture selection based techniques also depends on the forward and the backward trip time.

3.4.3 Error Tracking

Error Tracking [27] is a feedback based error resilience technique. The feedback channel is used to report corrupted image areas to the video encoder. The video encoder reacts to this feedback by tracking the spatio-temporal error propagation [27]. Those future frame areas, that have been identified to be in the corrupted area, are encoded in intra mode [27]. Since the encoder calculates the area to be encoded in intra mode for future frames, error tracking does not introduce additional delay [27]. The error tracking method does not stop the error propagation completely because only the most damaged

image regions are coded in the intra mode, and some error propagation may still be present. Also, encoding of macroblocks in intra mode decreases the coding efficiency significantly.

In [86], two error tracking schemes for transmission of H.263 video are presented. First scheme tracks the propagated error based on backward motion dependency of each pixel. The second scheme also utilizes four corner tracking approximation and linear motion model.

In [87], proxy based error tracking technique has been proposed. The proxy server at the base station performs error tracking. Since the end-to-end delay is low in the wired part, this technique performs better than the end-to-end error tracking technique [87]. In this technique, an additional update stream (I stream) is sent through the wired network [87]. This additional update stream is used to perform error tracking of the original video stream [87]. This technique may not be feasible to implement in practice because sending an extra intra stream through the wired part requires lot of bandwidth, especially if there are many users. The base station would have to decode and re-encode (transcode) the video for feedback-triggered intra updates [87]. Also, a lot of modification is required in the network hardware (base station) to implement this technique, which may be costly and not feasible in practice.

In [88], error compensation method has been proposed. In this method, encoder can remove visual quality degradation due to spatio-temporal error propagation by utilizing a feedback channel [88]. In this method, a corrupted group of blocks (GOB) is concealed, and the GOB and its corresponding frame number are reported to the encoder via the feedback channel [88]. Then, the encoder reconstructs the spatial and temporal error propagation by using a redecoding algorithm [88].

In selective recovery [89] method, the receiver detects the damaged picture area caused by packet loss and does error error concealment. It notifies the transmitter of the damaged area. The decoding continues during this process. The transmitter does not use the damaged areas as reference. This technique is unsuitable for H.264 video coding standard because of sub-pixel motion estimation and compensation and so the damaged area in a frame is expanded in the successive frames which results in spatio-temporal error propagation. Also, restricting the area used for inter prediction reduces the coding efficiency. Another limitation is that it is not clear whether real-time region

of interest based processing is feasible on hand-held devices.

3.4.4 Interactive Error Control (IEC)

In interactive error control [13, 14] technique, the video encoder uses the negative feedback to determine the area(s) that has been damaged, due to packet loss, in the frame. It reconstructs the spatio-temporal error propagation by increasing the original damaged area(s) by two pixels in every successive frame. The increase of two pixels in successive frames is done to compensate for the sub-pixel motion estimation and compensation in the H.264. The encoder and decoder keep multiple past frames in the buffer. The encoder uses rate distortion optimization. For inter coding modes the distortion from the damaged area(s) is infinite and hence, these modes are invalid and not used. If no valid inter coding modes are found in the reference frame, then the encoder uses older reference frames and the same process is repeated. This process does not affect the intra coding modes. Using older reference frames decreases compression efficiency. Also, in this technique a frame is divided into many slices, which results in decreased compression efficiency and increased header overhead. There are two variants of this technique. The first variant called IEC1 uses the nearest previous frame as reference. The second variant called IEC2 uses a reference frame which is further back in time. The reason for using an older reference frame is that the error propagation is limited to fewer frames. However, use of older reference frame decreases compression efficiency severely.

3.4.5 Redundant slices based techniques

In [90], redundant slices and flexible macroblock ordering has been used to provide error resilience. The redundant slices are generated based on motion content [90]. For static background sequences, a region of interest composed mainly of foreground is defined and only this region is encoded as redundant slices [90]. A method is proposed that determines the bit rate of the redundant slices description such that the expected decoder distortion is minimized [90].

In [95], redundant picture coding is combined with reference picture selection and reference picture list reordering [95]. A hierarchical redundant picture allocation method and content adaptive redundant picture allocation techniques are proposed [95]. The limitation of redundant picture coding techniques is how to achieve an optimal trade-off between compression efficiency and error resilience in dynamic packet loss rate environ-

ment.

In [91], flexible macroblock ordering and redundant slices is combined with an algorithm for dynamic allocation of redundant slices [91]. The dynamic allocation of redundant slices is based on the fading wireless channel characteristics [91].

A drawback of these techniques are that use of redundant slices requires extra bandwidth. Redundant slices can also be lost. Slices decrease compression efficiency. Moreover, these techniques do not address the problem of spatio-temporal error propagation. Redundant slices require defining a region of interest. For high motion video sequences, defining a region of interest is an open question.

3.4.6 Slice size adaptation

In [53], the size of the slice is adapted dynamically according to the RLC-PDU loss rate as reported by the RTCP reports. Dynamically adapting slice size can achieve better error resilience and video quality.

3.5 Hybrid techniques

In [92], Reed-Solomon forward error correction is used with periodic slice intra update. This scheme does not use any feedback. In this technique, many frames may pass before error propagation is stopped. Since the location of error is not known to the encoder, therefore the error propagation is not terminated completely. Reed-Solomon codes are not the fastest error correction codes.

In [93], Reed-Solomon forward error correction is used along with periodic reference frames. In periodic reference frame, every mth frame is encoded with the nth previous frame as reference [93]. These frames are known as periodic reference (PR) frames, and n is the frame period, which is the number of frames between periodic reference frames [93]. All the other frames are coded as using the previous frame as reference [93]. A limitation of this technique is that using fixed distance reference frame decreases the compression efficiency in the error free case. Also, it may not stop error propagation early enough compared to instantaneous feedback based schemes.

In [94], a joint source channel coding framework for robust video transmission is presented. This technique uses forward error correction using Reed-Solomon codes,

application layer retransmissions and inter/intra mode switching based on expected decoder distortion.

An error resilience technique has been proposed in [96]. This technique uses forward error correction, automatic repeat request, data partition and error tracking [96]. A proxy server is implemented in the base station [96]. The proxy server in the base station reduces the latency with retransmissions and also provides error tracking functionality [96]. 3GPP has specified H.264 baseline profile for video telephony and data partition is not included in the H.264 baseline profile. Implementation of proxy server in the base station is expensive process and may require modification of network hardware. So, this technique may be not feasible in practical 3G video telephony applications.

In [98], an efficient packetization and rate control based error resilience technique has been proposed for real-time video transmission. The source coding parameters are adjusted based on channel parameters such as packet loss and delay.

HARQ based techniques use a combination of forward error correction and automatic repeat request [99]. In [100], a conditional retransmission strategy has been proposed. This method reduces the number of retransmissions. For delay constrained real-time multimedia applications, the retransmission is aborted if the presentation deadline left is less than the round trip delay time.

In [101], a channel adaptive HARQ technique is proposed. An algorithm is proposed that adjust the parity data length and the maximum number of retransmissions based on the channel error model.

A limitation of the above techniques is that they do not address the problem of spatio-temporal error propagation.

In [97], MAC/physical layer forward error correction is combined with automatic repeat request and reference picture selection. A limitation of this technique is that it is valid only when the transmitter and receiver are connected through a direct link. This technique is not valid in scenarios where there are multiple wired or wireless links between the transmitter and the receiver.

Unequal error protection based error resilience technique has been evaluated in [53].

The reference frames are protected using forward error correction. To compensate for forward error correction redundancy, the source bit rate is reduced.

3.6 Cross layer optimization

In [102, 103], cross layer optimization of video transmission over wireless channels has been discussed. In [104], the effect of packet size and packetization for video transmission over wireless channels has been researched. The effect of slice size and hence the packet size on the decoded PSNR for video transmission over 3G networks has been given in [13, 14]. These papers discuss the cross layer optimization between application/transport layer and data-link layer and the parameter for optimization is the packet size.

Chapter 4

Fixed redundancy LT coding, reference picture selection and cross layer optimization

4.1 Introduction

In this chapter, an error resilience technique for packet switched video telephony over 3G networks, is proposed. The proposed error resilience technique is a combination of application layer fixed redundancy forward error correction, reference picture selection and cross layer optimization. For forward error correction, rateless (Luby Transform (LT)) codes are used. Also, a RTP/UDP/IP level packetization scheme is proposed. The end-to-end delay analysis is given in Section 4.4.

4.2 Proposed method

The proposed method is a combination of application layer forward error correction using rateless (LT) codes, reference picture selection and cross layer optimization. This method is called **Fixed** method in this dissertation. The details of the method are given in the following sections.

4.2.1 System description

The block diagram of the proposed system is shown in Figure 4.1. The description of various blocks in the system is given below:

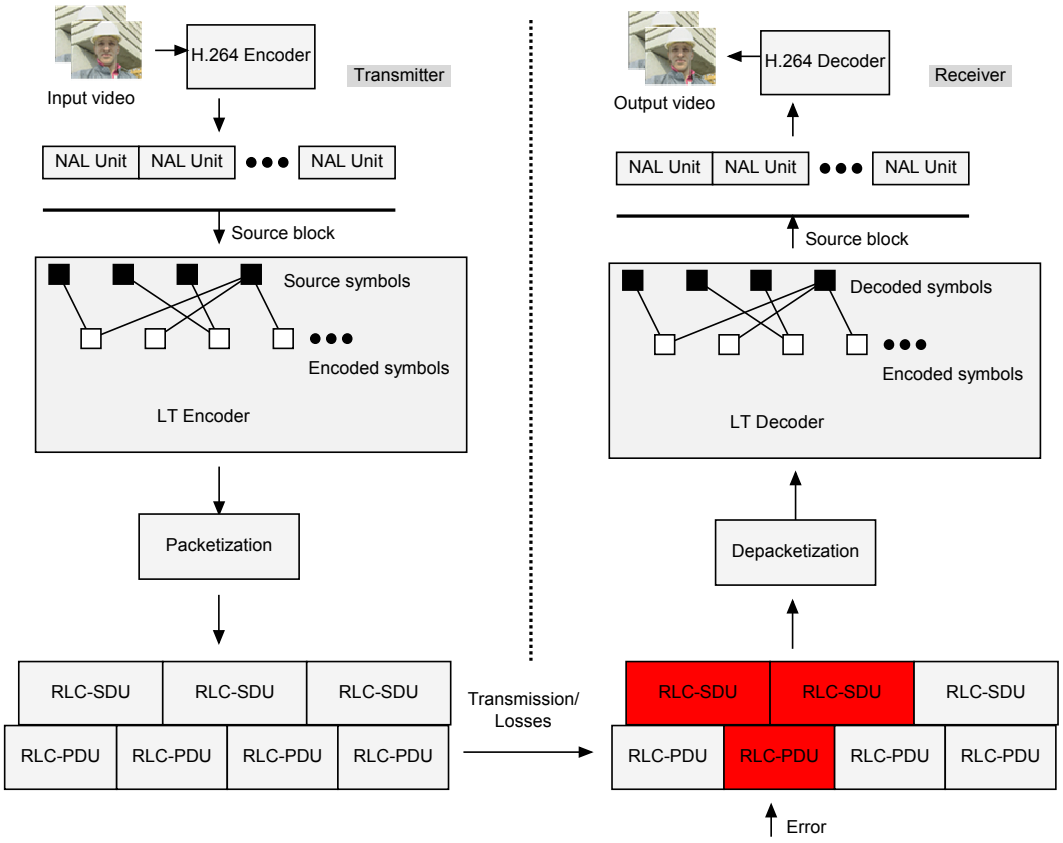


Figure 4.1: System block diagram.

H.264 video encoder

Input to the H.264 video encoder is video in the form of YUV frames. Video frames are fed to the H.264 video encoder in real-time. The video frame rate is constant. H.264 video encoder compresses the video and produce output bitstream in the form of NAL units. The NAL units of each frame are stored in a buffer. This is done so that LT encoding can be applied to the NAL units of specific number of frames.

LT encoder

When a specific number of video frames have been encoded, then the NAL units of these video frames become the source block of the LT encoder. LT source block consists of NAL units corresponding to two video frames. The reason for choosing two video frames as LT source block is that higher number of video frames result in larger LT source block. Larger LT source block is more efficient for LT decoding. However,

larger number of frames in LT source block also results in more frame buffering and transmission delay. A compromise must be done between the decoding performance of LT code and the end- to-end-delay. A LT source block of 2 frames is thus a compromise between LT decoding performance and the end-to-end-delay. When all the NAL units for two frames are available in the buffer then LT encoder is called. LT encoder generates encoded symbols according to the LT code redundancy. The LT code redundancy is fixed in advance.

Packetization

The LT encoded symbols produced by the LT encoder are packed in RTP/UDP/IP packets. RTP/UDP/IP packets are converted to RLC-SDUs according to the 3GPP protocol stack given in Section 2.1.4. The contents of RTP/UDP/IP packets are given in Section 4.2.6.

Transmission over a 3G wireless channel and packet losses

A 3GPP software simulator [37] is used to simulate the video transmission over a 3G wireless channel. The 3GPP simulator maps the RLC-SDU to RLC-PDU for transmission. The size of the RLC-PDU is fixed.

The mapping of the RLC-SDU to RLC-PDU is done such that if a RLC-PDU contains the last byte of a RLC-SDU and not all the bytes in the fixed sized RLC-PDU have been used then the first byte of the next RLC-SDU is concatenated with the last byte of the previous RLC-SDU in the same RLC-PDU. Thus, a RLC-PDU may contain data of more than one RLC-SDU. Losses are applied to RLC-PDUs, based on traces obtained from real transmission over a 3G network. If a RLC-PDU is lost according to the traces, then the RLC-SDUs that are partially or fully mapped to that RLC-PDU are also considered to be lost. This is shown in Figure 4.1.

Depacketization

The RLC-SDUs which are not lost according to the traces, are depacketized to form PDCP packets. PDCP packets are depacketized to form RTP/UDP/IP packets.

LT decoder

RTP/UDP/IP packets contain LT encoded symbols and information for LT decoder to decode all the symbols contained in these packets. The LT decoder decodes the

encoded symbols in these packets. If enough encoded symbols have been received, then LT decoder successfully decodes the encoded symbols to recover LT source block. The recovered LT source block consists of the NAL units corresponding to two video frames. Successfully recovered NAL units are given to the H.264 video decoder. If due to packet losses, enough symbols are not received, then LT decoding fails and not all the source symbols are recovered. In that case, all the recovered source symbols are discarded. Also, the H.264 decoder is notified of the LT decoding failure.

H.264 video decoder

When LT decoding is successful, then the H.264 decoder receives NAL units for two video frames. These NAL units are decoded by the H.264 video decoder to get two video frames. The video frames are displayed at the same frame rate as the input frame rate at the transmitter. In case of LT decoding failure, the H.264 decoder conceals the two missing video frames by the last successfully decoded frame. The concealed video frames are then displayed.

4.2.2 Timing diagram

The timing diagram is given in Figure 4.2. It is assumed that the video coding sequence is in the form IPPPP. The first video frame is encoded as I frame and the rest of the frames in the video sequence are encoded as P frames. Video telephony is bidirectional video transmission. It is assumed that the transmitter and receiver clocks are synchronized, which can be achieved by using, for example, Network time Protocol (NTP) [105]. Also, the LT source block transmission start time for forward and backward transmission is the same. The description of various end-to-end delay components is given below:

- **Video encoding and frame buffering:** The first P frame is encoded at time t=0. The second P frame is encoded at time t=T1. Thus at time t=T1, two encoded video frames are available. The duration t=T1 is equal to 1/FPS. FPS is the video frame rate in frames per second. It is assumed that video encoding and decoding delay is negligible compared to the delay 1/FPS. The LT source block contains NAL units corresponding to two encoded video frames. Thus the time required to form LT source block 1 is 1/FPS. This time duration is known as frame buffering delay. The frame buffering delay for LT source block 1 is 1/FPS. The frame buffering delay for subsequent LT source blocks is 2/FPS.

- **LT encoding and transmission of LT source block:** The LT source block 1 is LT encoded on the fly, and the encoded symbols are transmitted in packets. LT source block 1 is transmitted from time t=T1 until time t=T2. At time t=T2, LT source block 2 is available for transmission. The duration between t=T1 and t=T2 is 2/FPS. The time duration between sending of each packet is fixed and is known as Transmission Time Interval (TTI).

- **Propagation delay:** The receiving deadline of packets for LT source block 1 at the receiver is t=T2+FTT. FTT is the maximum forward trip time. It should be noted that FTT may not be a multiple of transmission time interval (TTI).

- **LT decoding and video decoding:** At time t=T2+FTT, the LT decoding of received encoded symbols of the LT source block is started. After successful LT decoding, the NAL units associated with two video frames are obtained. The NAL units are decoded by the H.264 video decoder to obtain two video frames. The decoded video frames are played back at the same frame rate as the input frame rate at the transmitter. Thus, the time delay between the capture of every video frame and its playback is the same.

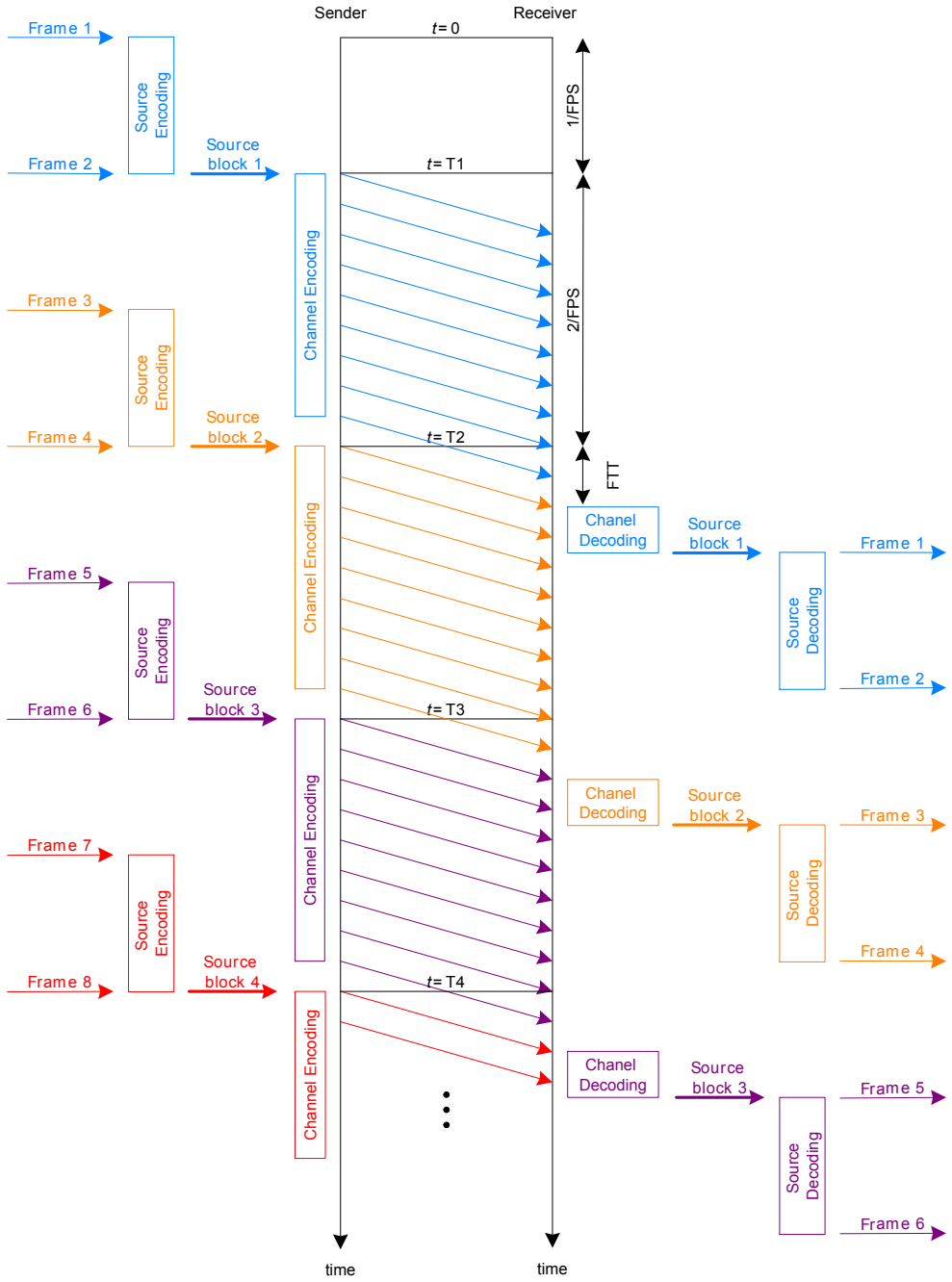


Figure 4.2: Timing diagram of the proposed error resilience scheme.

4.2.3 Transmission strategy

The transmission strategy of the proposed method is shown in Figure 4.3. Each user equipment sends and receives video simultaneously. The transmission start time of LT source block for both user equipments is the same. The transmission sending deadline of a LT source block is the deadline for sending the last packet of a LT source block. The transmission sending deadline of a LT source block for forward and backward transmission is given by:

$$\frac{n_f}{f_r} \tag{4.1}$$

where n_f is the number of video frames corresponding to a LT source block and f_r is the video frame rate.

The transmission receiving deadline of a LT source block is the deadline for receiving the packets for the current LT source block. After the transmission receiving deadline has passed, the LT decoder at the receiver decodes the received LT encoded symbols of the current LT source block. The transmission receiving deadline of a LT source block for the forward transmission is determined by:

$$\left(\frac{n_f}{f_r}\right) + \mathrm{FTT} \tag{4.2}$$

FTT is the maximum forward trip time. Similarly, the transmission receiving deadline of a LT source block for the backward transmission is determined by:

$$\left(\frac{n_f}{f_r}\right) + \mathrm{BTT} \tag{4.3}$$

BTT is the maximum backward trip time.

The number of LT encoded symbols k_s to send are determined by the formula:

$$k_s = k \times (1 + r) \tag{4.4}$$

Where k is the number of source or information symbols and r is the LT code redundancy. The redundancy of the LT code is fixed in advance. k_s symbols are packed in p number of RTP/UDP/IP packets. p is determined as:

$$p = \frac{k_s}{m} \tag{4.5}$$

m is the size of the RTP/UDP/IP packet. The LT encoded symbols are carried in fixed size RTP/UDP/IP packets. The size of the RTP/UDP/IP packet is chosen such that one RTP/UDP/IP packet is mapped to exactly one RLC-PDU. The size of the RTP/UDP/IP can be adjusted by changing the number of encoded symbols in a RTP/UDP/IP packet. After packetization, the redundancy of the LT code is as close to r as possible. The final redundancy of the LT code may be smaller or greater than r. The size of the RLC-PDU is fixed in advance. The Transmission Time Interval (TTI) is also fixed in advance. One RLC-PDU (and hence one RTP/UDP/IP packet) is sent at a given TTI.

If some packets and LT encoded symbols are lost, and not enough LT symbols are received then LT decoding fails. The deadline for declaring LT decoding failure is the LT decoding failure deadline. The LT decoding failure deadline is defined as the time needed before LT decoding stops, if k encoded symbols are received. After LT decoding failure deadline has passed, the feedback to invoke reference picture selection is generated and sent in the backward packets.

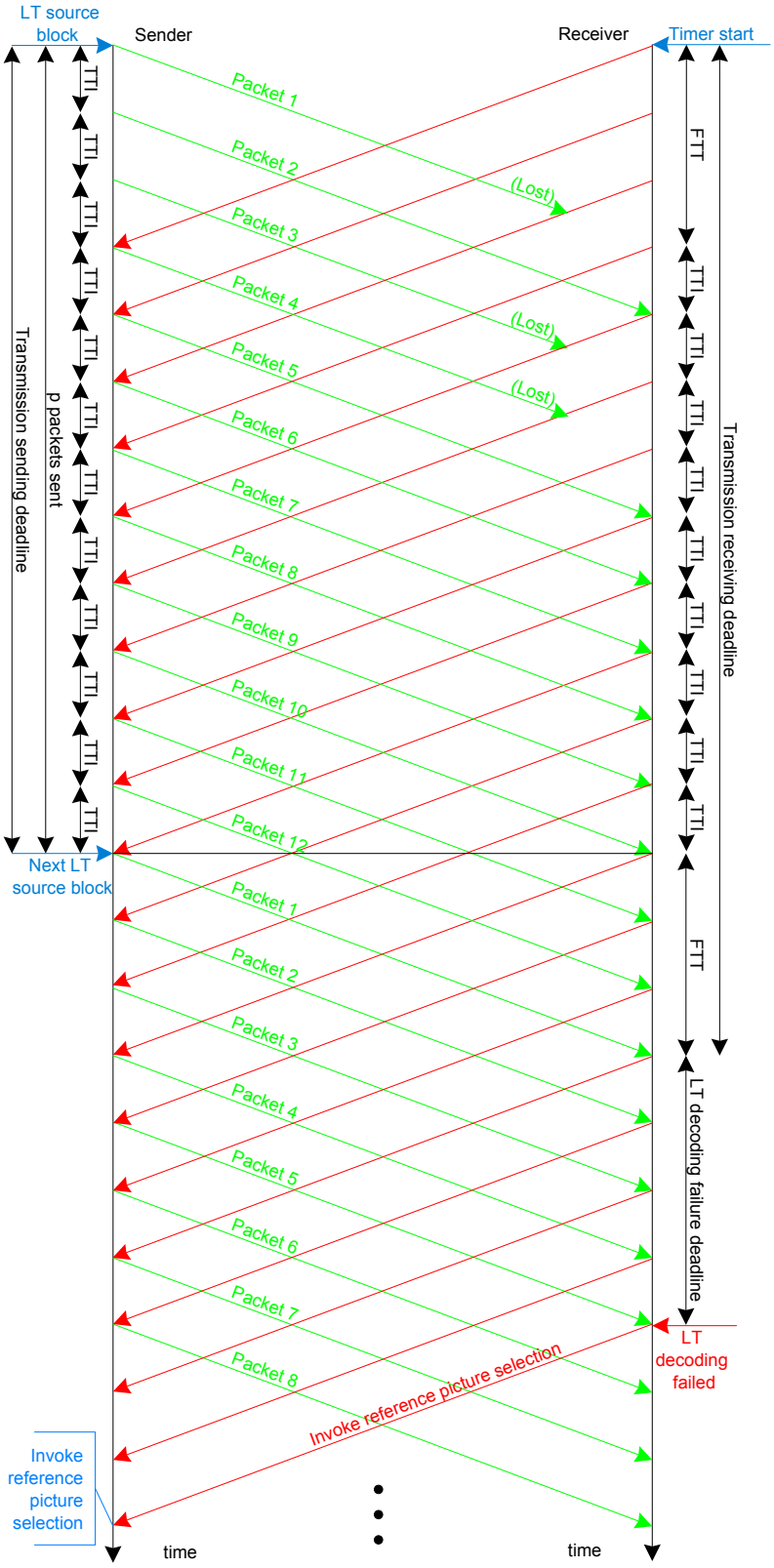


Figure 4.3: Transmissions strategy of the proposed error resilience technique.

4.2.4 Application layer Forward Error Correction

An application layer Forward Error Correction using LT code [9] is applied to the source block to generate the encoded symbols. The advantage of using forward error correction at the application layer is the flexibility to adapt to different network conditions and transmission parameters without needing to change the network hardware or the lower layers. The performance of the standard LT coding is enhanced with block duplication [30] method. The block duplication method increases the probability of LT decoding success by virtually increasing the number of source symbols.

4.2.5 Reference Picture Selection

If enough encoded symbols are not received due to packet losses, then LT decoding fails. The NAL units contained in the LT source block cannot be recovered. The video frames contained in the LT block are considered to be lost. The H.264 video decoder replaces the lost video frames by the last video frame in the last successfully decoded LT source block. This is known as freeze frame concealment. This results in different reference video frames at the transmitter and the receiver.

The difference of reference frames at the transmitter and receiver results in visual error in the decoded video frame. This visual error is propagated to successive video frames even if their LT source blocks are successfully decoded. The propagation of error in successive video frames is due to predictive nature of video coding. The propagation of errors in successive video frames is known as spatio-temporal error propagation.

To solve the problem of spatio-temporal error propagation, a variation of Reference Picture Selection [27] is used. The receiver sends feedback to the transmitter to inform the transmitter of the LT decoding failure at the receiver. The feedback also informs the transmitter of the last successfully decoded LT source block at the receiver.

Since the feedback takes some time to reach the transmitter, many video frames may have been encoded and sent by the H.264 encoder at the transmitter side. This results in mismatch of some reference frames at the transmitter and receiver side.

When the transmitter receives the feedback information about the LT decoding failure at the receiver side, it uses last successfully received and decoded video frame at the receiver as the reference frame for the next video frame. This results in exactly

the same reference frame at the transmitter and the receiver, hence stopping error propagation. The transmitter calculates the last successfully received and decoded video frame at the receiver by the last successfully decoded LT source block information. This method requires storing few reference video frames at the transmitter and the receiver side.

When not using reference picture selection, the other video frames are encoded normally using the previous video frame as reference. The reference picture selection method is shown in Figure 4.4. Here, LT source block 2 fails to decode and the feedback about its decoding failure is sent to the transmitter. The feedback reaches the transmitter before it encodes frame 7. The encoder calculates the lost video frames in the failed LT block and uses frame 2 as reference frame for encoding frame 7. The reference frames at the transmitter and receiver are now the same, hence error propagation is stopped.

The reference picture selection feedback is contained in the payload of the upstream RTP/UDP/IP packets (since this is a bidirectional video communication). The RPS feedback field is contained in every RTP/UDP/IP packet to provide robustness to RPS feedback against packet losses.

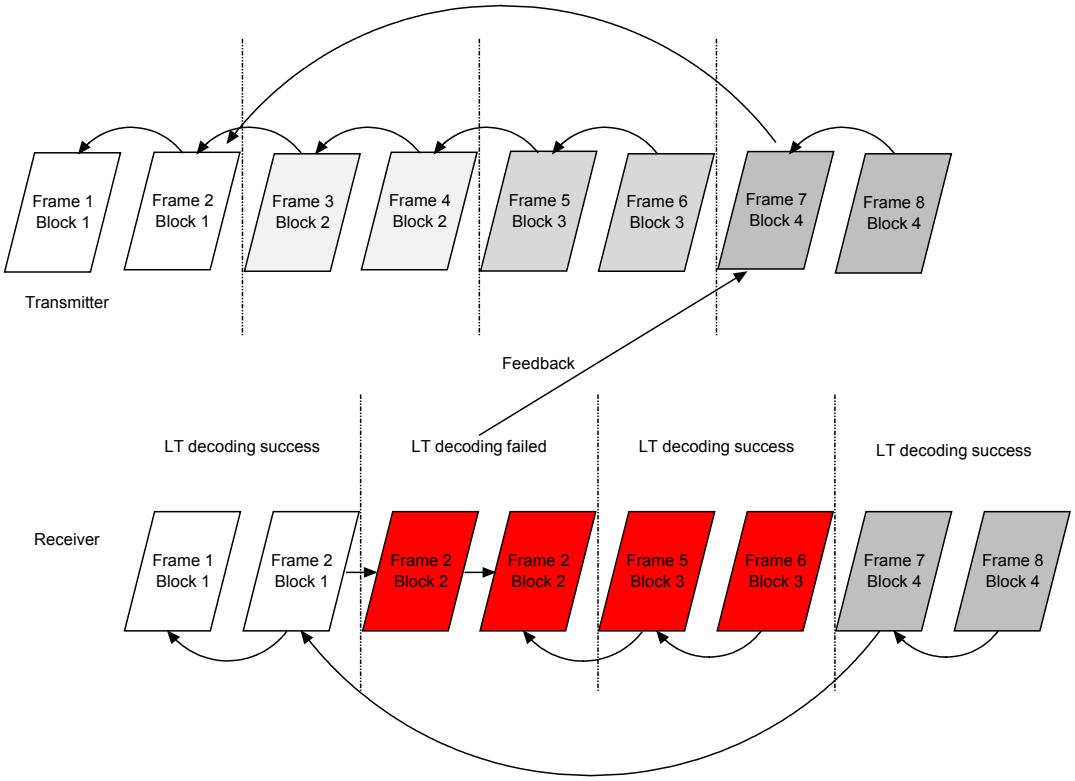


Figure 4.4: Illustration of Reference Picture Selection. The LT decoding of source block 2 has failed. The feedback is received by the encoder before encoding frame 7. The error propagation is stopped at frame 7.

4.2.6 Proposed packetization

The proposed packetization is shown in Figure 4.5. To facilitate LT decoding, LT information is added as fixed size fields in the payload of the RTP/UDP/IP packet. The description and function of the LT information fields is given below:

- **Sequence number:** Sequence number field contains the sequence number of the first LT encoded symbol in the RTP/UDP/IP packet. This is done so that the encoded symbols in the RTP/UDP/IP packet can be put in the same position in the LT coding graph at the LT decoder as it is in the LT encoder. This is necessary to achieve successful decoding of the LT encoded symbols. The size of this field is 2 bytes. This field is represented in base 2 (binary) format.
- **LT source block size:** This field contains the size of the LT source block, to

which the encoded symbols in the RTP/UDP/IP packet belong. This field informs the LT decoder of the source block size k. For LT decoding to be successful, the LT decoder must decode k source symbols for each source block. The size of the LT source block is in number of symbols. The size of this field is 2 bytes. This field is represented in base 2 (binary) format.

- **Block ID:** Block ID field contains the ID of the LT source block to which the encoded symbols in the RTP/UDP/IP packet belong. The block ID field informs the decoder of the LT source block to which the encoded symbols in the RTP/UDP/IP packet belong to. The size of this field is 4 bits. The LT source block ID range is from 0 to 15. This field is represented in base 2 (binary) format. The Block ID field is incremented by 1 for successive LT source blocks. After 15, the next LT source block ID starts from 0 and so on.

Also, fixed size fields are added in the RTP/UDP/IP packet payload for invoking reference picture selection at the transmitter. The description and function of the RPS fields is given below:

- **Invoke RPS:** When the receiver wants to invoke reference picture selection at the transmitter, then this field is set to 1, otherwise this field is set to 0. The size of this field is 4 bits.

- **Failed block ID:** This field contains the LT source block ID of the failed block at the receiver. This field informs the video encoder to discard the video frames which belongs to the failed LT source block. This field contains the ID of the failed LT block for which reference picture selection is being invoked. The size of this field is 4 bits. This field is represented in base 2 (binary) format.

- **Last successful block ID:** This field contains the block ID of the last successfully decoded LT source block at the receiver. The last successfully decoded LT block ID is used to determine the last successfully decoded video frame at the receiver. During reference picture selection, the transmitter uses that video frame as reference for encoding the next video frame. The size of this field is 4 bits. This field is represented in base 2 (binary) format.

The total size of the LT and reference picture selection fields is 6 bytes.

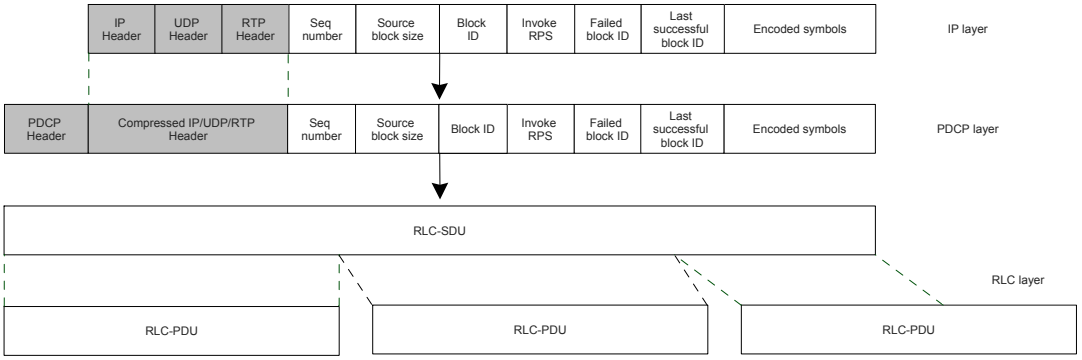


Figure 4.5: Proposed packetization of the RTP/UDP/IP packet. LT and reference picture selection fields are added to the RTP/UDP/IP payload. Also, the packetization of the RTP/UDP/IP packet through different layers of 3G protocol stack is shown.

4.2.7 Cross layer optimization

The RTP/UDP/IP packets are converted to RLC-SDU at data link layer. RLC-SDUs are mapped onto RLC-PDU for transmission over the 3G network. One RLC-SDU can be mapped to more than one RLC-PDU or vice versa, depending on the size of the RLC-SDU and RLC-PDU. The loss of an RLC-PDU results in loss of all RLC-SDUs that are mapped to the RLC-PDU. Since one RLC-SDU corresponds to one RTP/UDP/IP packet, so RLC-PDU loss results in loss of RTP/UDP/IP packets. The goal of cross layer optimization is to minimize the data loss at the transport layer, when the data is lost at the RLC layer. This goal is achieved by adjusting the size of the RTP/UDP/IP packet. Clearly, the best strategy is to map one RTP/UDP/IP packet to exactly one RLC-PDU. Thus, if a RLC-PDU is lost, only one RTP/UDP/IP packet is lost. The size of the RLC-PDU is fixed. The size of the RTP/UDP/IP packet is set so that one RTP/UDP/IP is mapped to exactly one RLC-PDU and all the bytes in the RLC-PDU are used. The size of the RTP/UDP/IP packet is calculated according to the following formula:

$$\text{RTP/UDP/IP payload size} = \text{RLC-PDU size} - \text{PDCP header size} - \text{Compressed RTP/UDP/IP header size}$$

The RTP/UDP/IP packet size can be configured by changing the number of encoded symbols in the RTP/UDP/IP packet. Since LT code is rateless, so a RTP/UDP/IP packet can contain any number of LT encoded symbols. Thus, RTP/UDP/IP packet size can be set to any value and it can be matched to RLC-PDU of any size. The cross layer optimization is shown in Figure 4.6.

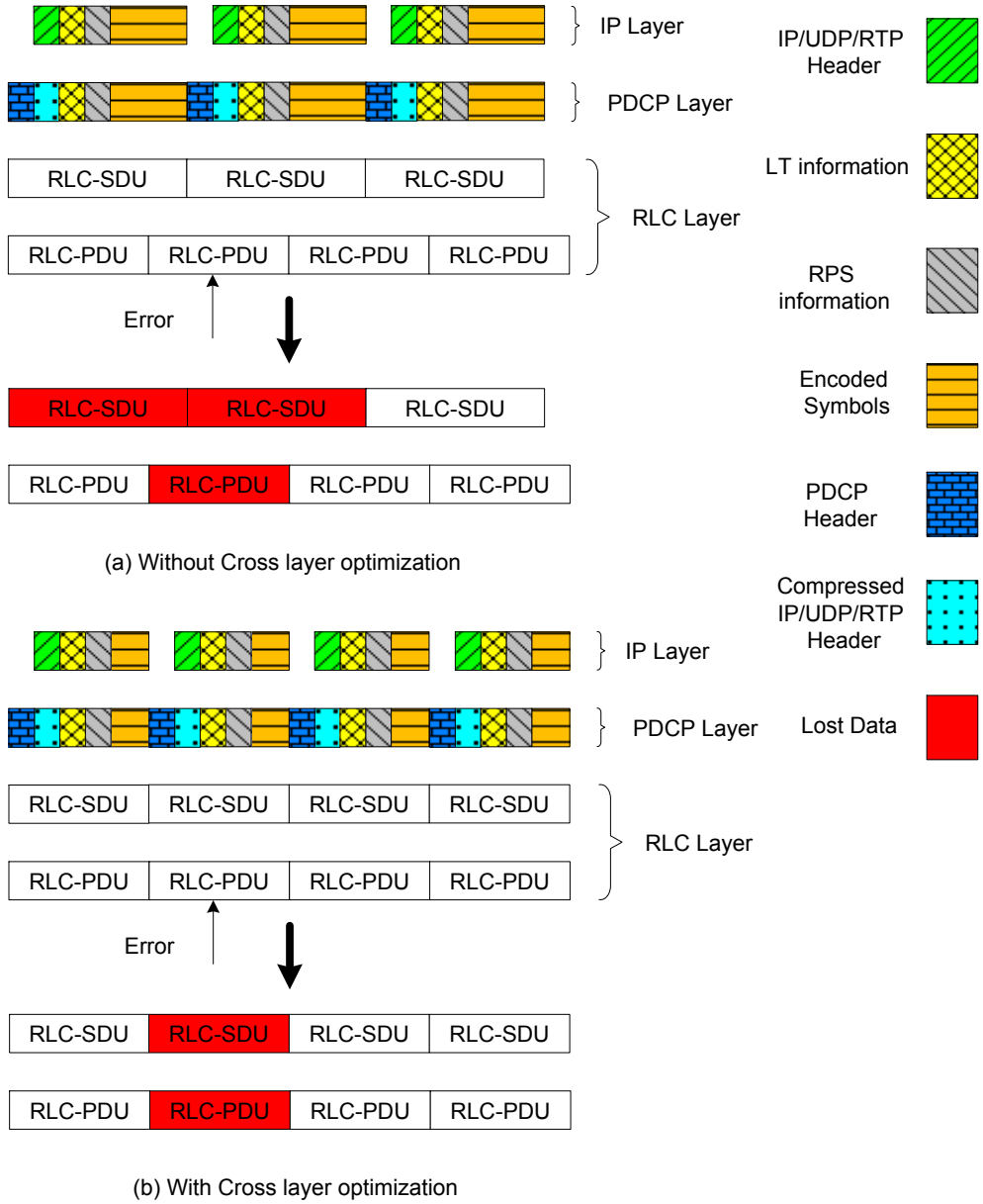


Figure 4.6: (a) Without cross layer optimization, if a RLC-PDU is lost then two RLC-SDUs are lost. (b) With cross layer optimization, if a RLC-PDU is lost then only one RLC-SDU is lost.

4.3 Experimental results

The proposed error resilience technique is compared to the reference techniques IEC1 and IEC2 [13, 14]. The goal is to show that the proposed technique can achieve better

results than IEC1 and IEC2 at the same experimental settings. The experiments are done under simulated and controlled settings. IEC is the most sophisticated and state of the art error resilience technique for packet switched video telephony over 3G networks. It combines several techniques for video telephony such as error tracking, reference picture selection, intra refresh, exploitation of feedback and RTP/UDP/IP packet size optimization.

The target system for simulations is the 3GPP HSPA system. The network topology is shown in Figure 4.7. The user equipment is connected to the base station through a wireless link. The base station is connected to the IP network and the other terminal through a wired link. This network topology is the same as used in the reference techniques IEC1 and IEC2.

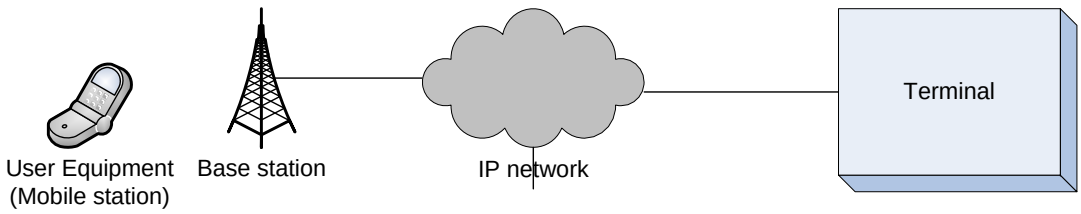


Figure 4.7: The network topology. One terminal is connected to the base station (node B) through a wireless link. The second terminal is connected to the wired IP based network.

- **Video coding parameters:** For video coding, Nokia H.264 video coder is used. QCIF (176×144) video sequences *Stunt* and *Party* are used. These video sequences are available from 3GPP [51]. *Stunt* sequence contains 240 frames and *Party* sequence contains 360 frames. The compressed video format is IPPP.... First frame is encoded as I frame and the rest of the frames are encoded as P frames. One reference frame is used. Rate distortion optimization is not used. Sub 8×8 coding modes are not used. Motion vector range is set to 8 pixels. The frame rate of the *Stunt* video sequence is 15 frames per second and the frame rate of *Party* sequence is 12 frames per second. CAVLC entropy coding is used.

  The compressed bit rate for *Stunt* video sequence is 89 kbit/s and for *Party* sequence is 90 kbit/s. These values of bit rate are chosen such that after applying LT coding, the total bit rate is 128 kbit/s. One slice per frame is used. Each video sequence is transmitted 20 times. The total number of frames transmitted for the *Stunt* sequence are 4800 and for the *Party* sequence are 7200. These video

coding parameters are chosen so that the proposed method can be compared to IEC1 and IEC2 [13, 14] under similar conditions.

- **LT coding parameters:** The LT source block contains NAL units of 2 video frames. Higher number of video frames in a source block results in better decoding efficiency for LT code. However, it also increase the frame buffering and transmission delay and increase the overall end-to-end delay. LT source block of 2 frames is a trade off between LT decoding efficiency and the end-to-end-delay. The number of LT encoded symbols to be generated are set as multiple of 307 bytes and as close to the redundancy of 35% as possible. This is done to facilitate cross layer optimization. This value of redundancy is chosen because it gives the best performance. The total send bit rate in the reference techniques IEC1 and IEC2 is 128 kbit/s. LT symbol size is 1 bit. Smaller LT symbol size results in more symbols per source block and improves LT code efficiency in terms of the decoding success rate. Robust Soliton distribution is used. The value of LT code constant c is 0.1 and δ is 0.5. The decoding efficiency of traditional LT code is improved by using the block duplication method [30]. Expanding Factor of 1 and 8 are used. Expanding Factor (EF) value of 1 means that block duplication is not used. Expanding Factor value of 8 means that block is duplicated 7 times. A real software implementation of LT codes is used.

- **Packetization parameters:** Each RTP/UDP/IP packet contains 2456 (307×8) encoded symbols. This number of encoded symbols in RTP/UDP/IP packet ensures that the size of the RLC-SDU is equal to 320 bytes. At PDCP layer, 5 bytes PDCP header is added and RTP/UDP/IP header is compressed to 5 bytes. Also every RTP/UDP/IP packet contains 6 bytes of LT and reference picture selection fields. Hence, the total size of each RLC-SDU is $2 + 5 + 6 + 307 = 320$ bytes. Since the size of RLC-PDU is 320 bytes, so one RLC-SDU (and one RTP/UDP/IP packet) is mapped to one RLC-PDU for transmission.

- **Transmission parameters:** The RLC-PDU size is fixed to 320 bytes. The TTI is also fixed to 20 ms. The bit rate of the radio bearer is given by:

  Bit rate of radio bearer (kbit/s) = RLC-PDU size (bits) / TTI (ms)

  The above formula gives the radio bearer bit rate of 128 kbit/s. The maximum forward trip time (FTT) and the maximum backward trip time (BTT) is 65 ms as in [13]. According to [13], this value of FTT and BTT was measured in a real

HSPA network. Average RLC-PDU loss rate traces of 0%, 0.5%, 1%, 1.5% and 5% are used. The above transmission parameters are exactly the same as in the reference techniques IEC1 and IEC2. This is done so that the proposed method can be compared to the reference techniques at exactly the same settings. The backward traffic is subject to the same RLC-PDU loss rate as the forward traffic.

- **Reference picture selection parameters:** The LT decoding failure deadline for the *Stunt* sequence is 30 ms and for the *Party* sequence is 40 ms. These deadlines are obtained through simulations using a real implementation of the LT code. After this deadline has passed, the feedback to invoke reference picture selection is generated.

  For the *Stunt* video sequence at 15 frames per second and *Party* video sequence at 12 frames per second, the reference picture selection feedback typically arrives at the transmitter after it has encoded 6 frames. Thus, a LT decoding failure typically affects 6 frames before the encoder uses the feedback information to invoke reference picture selection and stop error propagation. Since the feedback is contained in the data packets, so loss of data packets may delay the invocation of reference picture selection.

The PSNR vs. RLC-PDU loss rate results for the *Stunt* sequence are shown in Figure 4.8. At 0% RLC-PDU loss rate, the PSNR results of IEC1 [13, 14] are better than the proposed method. This is because the source bit rate of the proposed method is less than IEC1. The source bit rate of the proposed method is reduced because of application layer forward error correction (channel coding).

However, at RLC-PDU loss rates of 0.5% to 5%, the proposed method achieves better PSNR results than IEC1 and IEC2. This is due to the fact that application layer forward error correction protects the video data at the application layer. Also, the use of cross layer optimization decrease the number of RTP/UDP/IP packets lost compared to IEC1 and IEC2.

The PSNR results of the proposed method are better when Expanding Factor of 8 is used. This is due to the fact that Expanding Factor of 8 results in higher decoding efficiency than Expanding Factor of 1. Higher decoding efficiency of LT codes results in fewer LT decoding failures. Thus, PSNR results are improved. At RLC-PDU loss rate of 5%, the proposed method using Expanding Factor 8 achieves PSNR improvement of 2.1 dB compared to IEC1.

The PDVD vs. RLC-PDU loss rate results for the *Stunt* sequence are shown in Figure 4.9. The PDVD results indicate which technique is more effective in limiting spatio-temporal error propagation. Lower PDVD means better performance.

The proposed method using Expanding Factor 8 performs best at RLC-PDU loss rates of 0.5%, 1% and 1.5%. IEC2 performs best at RLC-PDU loss rate of 5%. This is because application layer forward error correction (channel coding) results in fewer LT decoding failures at RLC-PDU loss rate of up to 1.5% than at RLC-PDU loss rate of 5%. Fewer LT decoding failures results in fewer corrupted frames. IEC2 technique always uses an older reference frame. Thus, IEC2 technique is optimized to limit the spatio-temporal error propagation. That is why it achieves better PDVD results than IEC1 but lower PSNR results than IEC1. Also PDVD metric regards a frame as corrupted if its decoded PSNR is 2 dB less than its corresponding reconstructed PSNR.

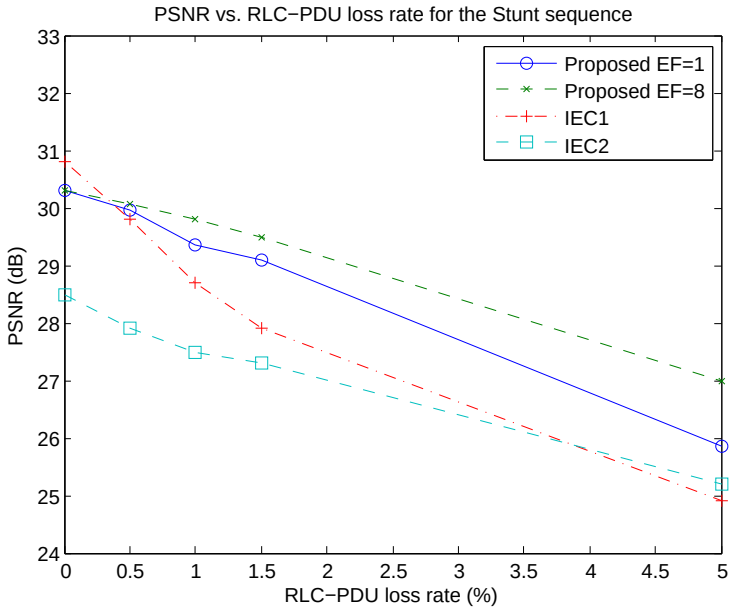


Figure 4.8: PSNR vs. RLC-PDU loss rate for the *Stunt* sequence.

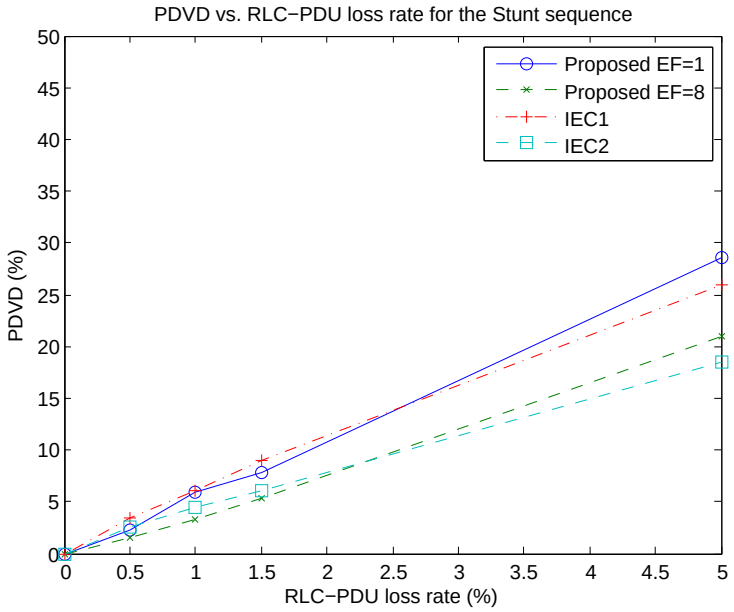


Figure 4.9: PDVD vs. RLC-PDU loss rate for the *Stunt* sequence.

The PSNR vs. RLC-PDU loss rate results for the *Party* sequence are shown in Figure 4.10. At 0% RLC-PDU loss rate, the PSNR results of IEC1 are better than the proposed method. This is because the source bit rate of the proposed method is less than IEC1.

However, at RLC-PDU loss rates of 1% to 5%, the proposed method achieves better PSNR results than IEC1 and IEC2. The PSNR results of the proposed method are better when Expanding Factor of 8 is used. At RLC-PDU loss rate of 5%, the proposed method using Expanding Factor 8 achieves PSNR improvement of 2.71 dB compared to IEC1.

The PDVD vs. RLC-PDU loss rate results for the *Party* sequence are shown in Figure 4.11. The proposed method using Expanding Factor 1 and 8 perform better than IEC1 and IEC2. The results of the proposed method for Expanding Factor 8 and 1 are almost similar at RLC-PDU loss rates of 0.5%, 1% and 1.5%. This is because when the RLC-PDU loss rate is low then the decoding efficiency improvement by the block duplication method is not significant. However, at RLC-PDU loss rate of 5%, the block duplication method results in significant improvement in LT decoding efficiency. At RLC-PDU loss rate of 5%, the proposed method using Expanding Factor 8 achieves

PDVD improvement of 7.49% compared to IEC2.

The results of the *Party* sequence are better than the *Stunt* sequence, in general. This is because the average LT source block size of the *Party* sequence is larger than the *Stunt* sequence. For *Stunt*, the average LT source block size is 12,064 symbols while for *Party* the average LT source block size is 14,960 symbols. The *Party* sequence has higher average LT source block size because of lower frame rate (12 fps) compared to the *Stunt* sequence (15 fps). Higher LT source block size results in more efficient LT codes. Moreover, the block duplication method achieves much more improvement for *Stunt* sequence than *Party* sequence due to the smaller source block size of the *Stunt* sequence.

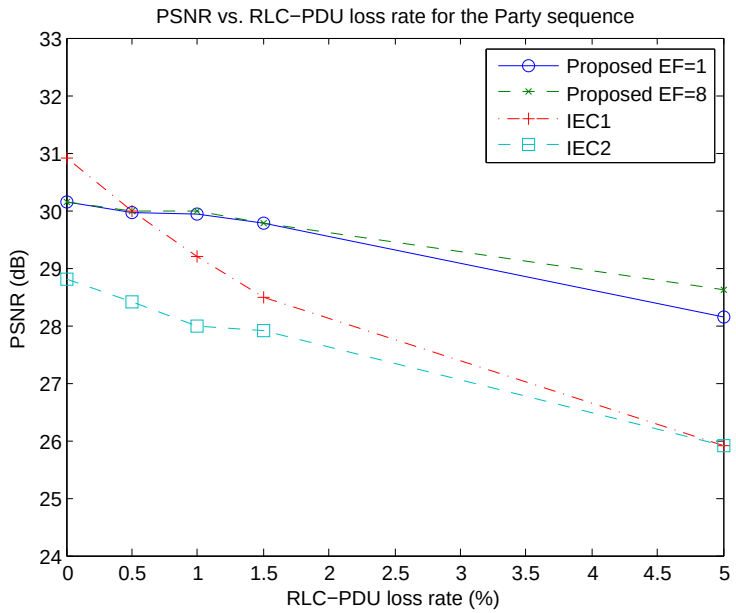


Figure 4.10: PSNR vs. RLC-PDU loss rate for the *Party* sequence.

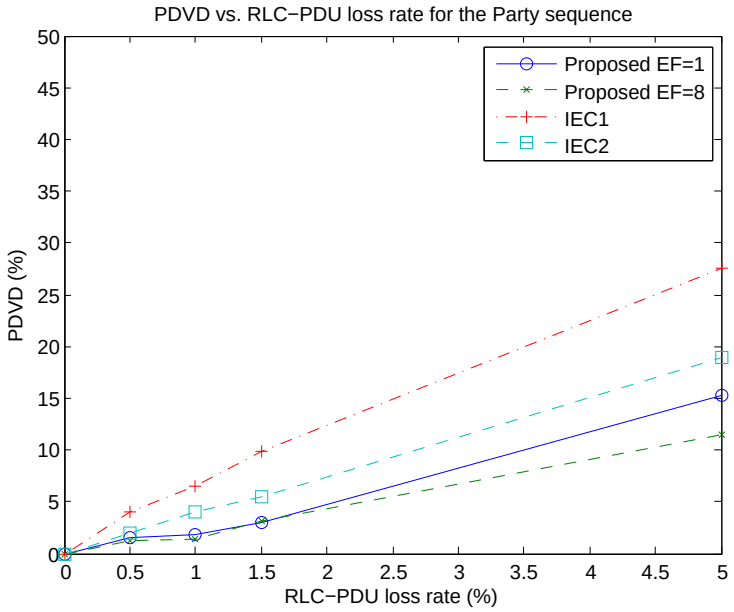


Figure 4.11: PDVD vs. RLC-PDU loss rate for the *Party* sequence.

4.4 End-to-end delay analysis

The end-to-end delay components for the *Stunt* video sequence are given in Table 4.1. The LT source block contains NAL units of 2 video frames. The video frame buffering (LT source block construction) time for the *Stunt* video sequence at 15 frames per second is equal to 1/15 = 66.6 ms. The H.264 video encoding delay is 4.6 ms, which is the time to encode 1 video frame. The LT encoding time delay is less than 1 ms because LT encoded symbols are generated with very low delay and on the fly.

The transmission sending deadline (transmission delay) is equal to 2/FPS = 2/15 = 133.3 ms. The propagation delay or maximum forward/backward trip time is 65 ms. The LT decoding delay for Expanding Factor 1 is 120 ms and for Expanding Factor 8 is 139.2 ms. The H.264 video decoding delay is 1.6 ms, which is the time required to decode 2 video frames. Thus, the total end-to-end delay for the *Stunt* video sequence for Expanding Factor 1 is 392.1 ms and for Expanding Factor 8 is 411.3 ms. The increase in the LT decoding delay for Expanding Factor 8 is due to increased complexity of LT codes when higher Expanding Factor is used.

The end-to-end delay components for the *Party* video sequence is given in Table

4.1. The LT source block contains NAL units of 2 video frames. The video frame buffering (LT source block construction) time for the *Party* video sequence at 12 frames per second is equal to 1/12 = 83.3 ms. The H.264 video encoding delay is 4.6 ms, which is the time to encode 1 video frame. The LT encoding time delay is less than 1 ms.

The transmission sending deadline (transmission delay) is equal to 2/FPS = 2/12 = 166.6 ms. The propagation delay or maximum forward/backward trip time is 65 ms. The LT decoding delay for Expanding Factor 1 is 186 ms and for Expanding Factor 8 is 223 ms. The H.264 video decoding delay is 1.6 ms, which is the time required to decode 2 video frames. Thus, the total end-to-end delay for the *Party* video sequence for Expanding Factor 1 is 508.1 ms and for Expanding Factor 8 is 545.1 ms.

The total end-to-end delay for the *Stunt* video sequence at Expanding Factor 8 and for *Party* video sequence at Expanding Factor 1 and 8 is higher than the maximum end-to-end delay of 400 ms [1], as specified by the 3GPP for video telephony. The major component of the end-to-end delay in the proposed method is the LT decoding delay. This is due to the fact that the software implementation of the LT code which is used in the proposed method is not optimized for speed.

A faster implementation of LT code is given in [106]. For a source block size of 1000000 symbols, the implementation of LT code used in the proposed method took 15 seconds for encoding and 327 seconds for decoding compared to 2 seconds for LT encoding and 37 seconds for LT decoding for the LT code implementation given in [106]. It should be noted that the PC configuration used in [106] is Intel P4 1.7 GHz and 512 MB RAM and the PC configuration used to test the proposed method is Intel Core 2 Duo 1.83 Ghz and 1 GB RAM. Thus, the LT code implementation given in [106] achieves lower encoding and decoding times compared to the LT code implementation, which is used in the proposed method. Also, the hardware configuration used in the proposed method is more powerful than the hardware configuration used in [106]. Thus, if a faster implementation of LT codes is used, then the total end-to-end delay for all configurations can be below the maximum end-to-end delay limit of 400 ms. The implementation of LT code that was used in the proposed method was not optimized for speed due to limited time frame. Since the implementation of [106] was not available, so it was not possible to comment on the differences between our implementation of LT code and the implementation of [106].

Delay component	Stunt EF=1 delay (ms)	Stunt EF=8 delay (ms)	Party EF=1 delay (ms)	Party EF=8 delay (ms)
Frame buffering	66.6	66.6	83.3	83.3
H.264 encoding	4.6	4.6	4.6	4.6
LT encoding	< 1	< 1	< 1	< 1
Transmission delay	133.3	133.3	166.66	166.66
Propagation delay	65	65	65	65
LT decoding	120	139.2	186	223
H.264 decoding	1.6	1.6	1.6	1.6
Total delay	392.1	411.3	508.1	545.1

Table 4.1: End-to-end delay components for *Stunt* and *Party* video sequences for Expanding Factor (EF) 1 and 8. The total end-to-end delay is the sum of the end-to-end delay components.

4.5 Summary

An error resilient technique for packet switched mobile video telephony was proposed. The proposed error resilient technique is a combination of application layer fixed redundancy forward error correction using rateless codes, reference picture selection and cross layer optimization.

The proposed error resilience technique achieves better results than a state of the art error resilience technique known as IEC1 and its variant (IEC2) [13].

The CPU time of the proposed error resilient technique using a real implementation of LT codes was measured to be 391.1 ms, 410.3 ms, 507.1 ms and 544.1 ms. The CPU time can be reduced by using an efficient implementation of LT codes or by using Raptor codes.

The improvement in results comes at an increased end-to-end delay compared to the method in [13]. The method of [13] combines the advantages of intra refresh, error tracking, reference picture selection, H.264 error resilience feature and instantaneous feedback. This is the reason for comparing the proposed fixed method with the method of [13].

A RTP/UDP/IP packetization scheme was also presented for the proposed error resilient scheme.

Chapter 5

Channel adaptive LT coding and reference picture selection

5.1 Introduction

In the fixed redundancy method proposed in Chapter 4 [11], the redundancy of the LT codes for a LT source block is fixed in advance. The performance of this method can be improved by adapting the LT code redundancy to the channel conditions. The channel adaptivity is achieved by combining the method presented in Chapter 4 with Automatic Repeat Request (ARQ).

ARQ is a retransmission based transmission technique, where the transmitter retransmits lost or corrupted packets. ARQ can be useful in low end-to-end delay conversational video application, such as video telephony. Low forward and backward trip time on mobile cellular system such as Long Term Evolution make ARQ an attractive choice.

5.2 Proposed method

The proposed method is a combination of channel adaptive LT coding and reference picture selection. This method is called **Adaptive** method in this dissertation. In case of a RLC-PDU loss, request for retransmission is sent to the transmitter. The transmitter retransmits RTP/UDP/IP packets containing LT encoded symbols. The total number of LT encoded symbols sent depends on the RLC-PDU loss rate in the wireless channel. Hence, the total redundancy of the LT code is adaptive to the wireless

channel conditions.

5.2.1 System description

The system used in this method is given in Figure 4.1.

5.2.2 Transmission strategy

The transmission strategy of the proposed method is shown in Figure 5.1. It is assumed that the transmitter and receiver clocks are synchronized. Each user equipment sends and receives video simultaneously. The transmission start time of LT source block for both user equipments is the same. The transmission sending deadline and the transmission receiving deadline of a LT source block is determined as in Section 4.2.3. The maximum forward trip time is denoted by FTT and the maximum the backward trip time is denoted by BTT.

The number of initial LT encoded symbols k_s to send in the first round are determined by the formula:

$$k_s = k \times (1 + r)$$

Where k is the number of source or information symbols and r is the initial LT code redundancy. The initial redundancy is chosen such that LT decoding has very high probability of success, if all the symbols are received and no encoded symbol is lost. These symbols are packed in RTP/UDP/IP packets. Flexible RLC-PDU size feature of LTE standard is used. The flexible RLC-PDU size allows RTP/UDP/IP packet of any size to be mapped onto exactly one RLC-PDU. In LTE, the segmentation is done at the Medium Access Control (MAC) layer [107]. This is known as flexible RLC and MAC segmentation solution [107]. In flexible RLC-PDU size, the size of the RLC-PDU can be from 10 bytes to 1500 bytes [107]. The maximum size m of the RTP/UDP/IP packet is chosen such that it is mapped to RLC-PDU size of exactly 320 bytes. The Transmission Time Interval (TTI) is fixed. k_s symbols are sent in p number of RTP/UDP/IP packets, with $p = \frac{k_s}{m}$. First p-1 packets have the same size which is equal to the maximum RTP/UDP/IP packet size. The last packet size may be smaller than the first p-1 packets. RTP/UDP/IP packets are mapped to RLC-PDU for transmission.

At t=0, the LT source block is available for transmission. At t=0, the transmitter

sends first packet. At t = TTI, the transmitter sends next packet. For each $t \geq 1$, the transmitter sends nth packet at $t = (n-1) \times$ TTI. The total number of initial packets sent are p. p packets are shown in green in Figure 5.1.

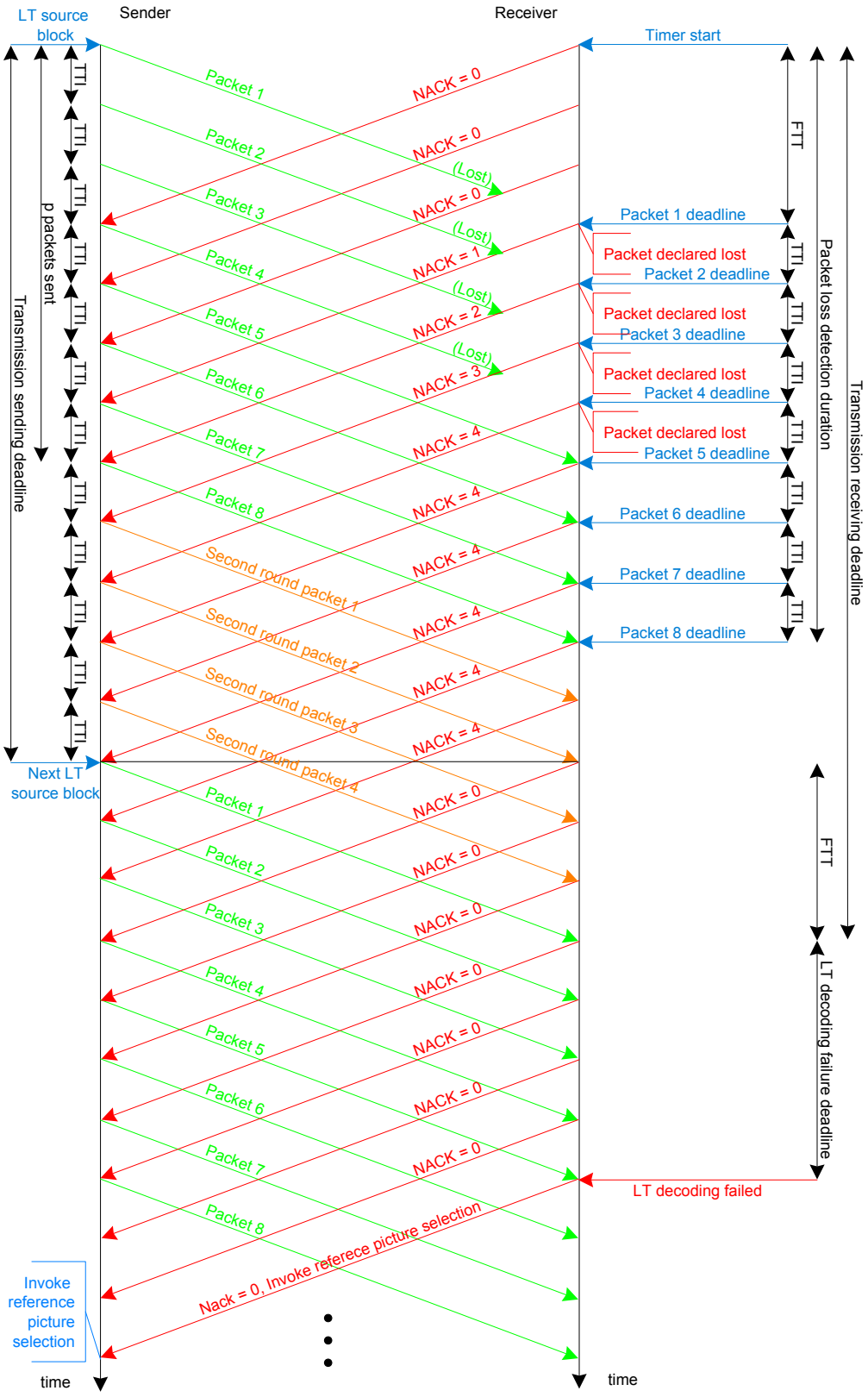


Figure 5.1: Transmission strategy of the proposed channel adaptive method.

The receiver starts a timer at $t = 0$. It sets a deadline t_n for receiving the nth packet by:

$$t_n = \text{FTT} + (n-1) \times \text{TTI}, n = 1..., \hat{p}$$

Here $\hat{p}$ is an estimation of p. It is computed as:

$$\hat{p} = \frac{\hat{k} \times (1+r)}{m}$$

Where

$$\hat{k} = \frac{s_r \times n_f}{f_r}$$

s_r is the source rate, n_f is the number of frames in LT source block and f_r is the video frame rate. $\hat{p}$ is then updated to p as soon as a packet is received (this is possible since each packet contains the value of k). The receiver keeps a record of the number of packets received for the current source block. For each $n = 1, ..., p$, if by the nth deadline it does not receive at least n packets, it concludes that some packets have been lost.

The receiver keeps a record of the number of packets received for the current LT source block. The cumulative number of lost packets is determined by the following formula:

Cumulative number of packets lost = nth deadline - Number of packets received

The cumulative number of lost packets is sent as Negative Acknowledgment (NACK) in the backward packets. In Figure 5.1, first 4 packets are not received by the receiver by 4th deadline. So the number of lost packets is 4-0=4. The receiver sets NACK to 4 and sends it in the subsequent packets. The packets are sent in both ways because this is bidirectional communication. The value of NACK is updated as soon as another packet loss is detected.

The packet deadline scheme is used to detect packet losses because there are two wireless links, which are source of packet losses. If RLC-PDU (one RTP/UDP/IP packet

is mapped onto one RLC-PDU) is corrupted in one wireless link, then the RLC layer discards the RLC-PDU (packet) in that wireless link. Hence, the packet does not pass through the second wireless link, so the receiver at the second wireless link has no exact information about packet being sent or lost. The reason for using packet deadline is to detect packet loss early at the application layer by the user equipment.

When the transmitter receives a backward packet, it checks the NACK for the current LT source block. The first p packets are sent at first p transmission opportunities. These p packets are called first round or initial round packets. The packets containing LT encoded symbols, that are sent after all p packets have been sent are called second round packets. The number of second round packets to send at every TTI is determined using the formula below. The sender keeps a record of the number of second round packets it has already sent for the current LT source block. It determines the number of second round packets to send by:

Number of second round packets to send = NACK - Cumulative number of second round packets already sent

The number of second round packets to be sent is updated at every TTI. The second round packets are sent at the next available TTI. If a NACK is received but not all the initial p packets have been sent then the transmitter will send one of the p packet at the next available TTI and second round packet will be sent once all p packets have been sent. In Figure 5.1, when the sender receives the NACK, it sends a second round packet which is shown in orange colour. The size of the second round packet is set to the maximum size of the RTP/UDP/IP packet. The second round packets are not sent if they would miss their delay deadline.

There is no deadline for the second round packets at the receiver. This is because the backward packets are susceptible to losses and NACK information can get lost and the receiver has no prior knowledge of when or if a second round packet is sent.

If the transmitter has sent all p packets and there is no second round packet to send, then it sends packets containing only NACK and reference picture selection information at the next available TTI. These non-LT packets contain NACK and reference picture selection information for the receiver side. Since these packets do not contain LT symbols, this results in bandwidth savings. The NACK and reference picture selection

information in these packets help to convey feedback information for the transmitter side reliably and early. These non-LT packets are sent until the transmission sending deadline for the current LT source block is reached. The non-LT packets are not shown in Figure 5.1 for simplicity. The contents of the initial round packets, second round packets and non-LT packets are given in Table 5.1.

Initial round packet	Second round packet	Non-LT packet
Contains LT encoded symbols, LT information, NACK and reference picture selection feedback	Contains LT encoded symbols, LT information, NACK and reference picture selection feedback	Contains NACK and reference picture selection feedback

Table 5.1: Table showing packet types and their contents.

A case when packets containing NACK are lost is shown in Figure 5.2. The transmitter sends the second round packet after it has received the packet containing NACK and if the number of second round packets already sent is less than NACK.

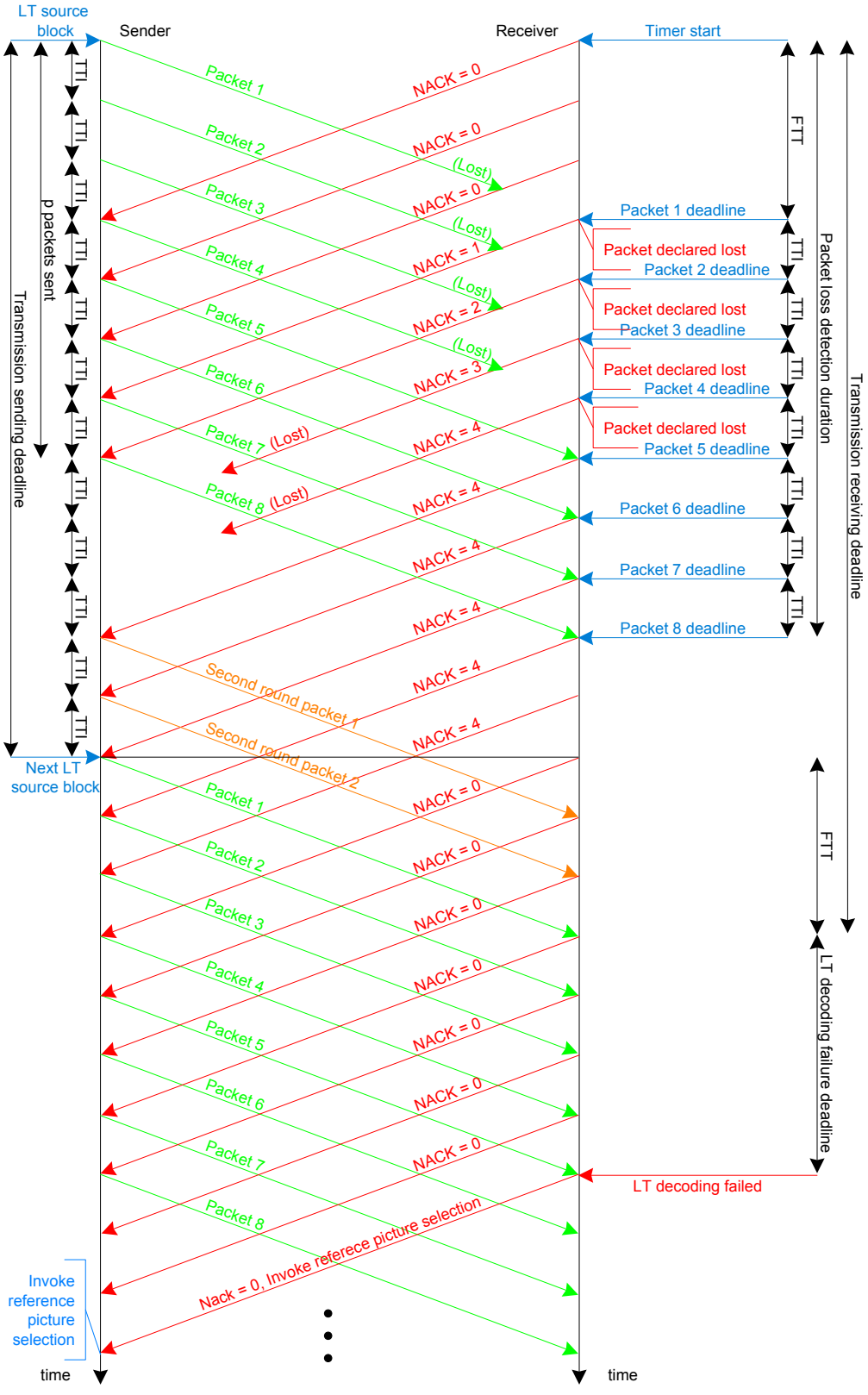


Figure 5.2: Transmission strategy in a case when two packets containing NACK are lost.

5.3 Proposed packetization

The proposed packetization for the initial round packets and the second round packets of the channel adaptive method is given in Figure 5.3. The packetization is similar to Figure 4.5 except for the addition of one new field, which is the Negative Acknowledgment (NACK) field. The NACK field contains the cumulative number of lost packets for the current LT source block. The size of the NACK field is 1 byte. The total size of the LT information, reference picture selection and NACK fields is 7 bytes. Non-LT packets do not contain LT information fields and LT encoded symbols.

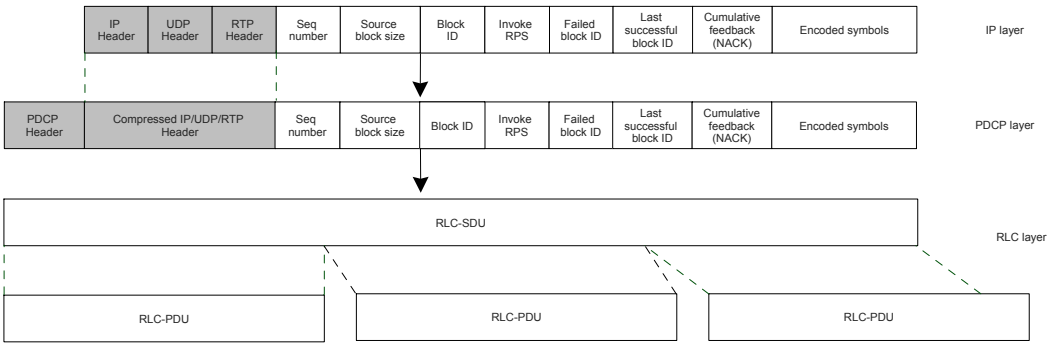


Figure 5.3: Packetization of the RTP/UDP/IP packet. LT information, reference picture selection information and NACK fields are added to the payload of the RTP/UDP/IP packet as fixed size fields.

5.4 Timing diagram

The timing diagram of the proposed channel adaptive method is the same as in Figure 4.2.

5.5 Experimental results

The target 3GPP mobile cellular system is the Long Term Evolution (LTE) system. The scenario is that of 2 mobile users communicating with each other. The network topology of the scenario is given in Figure 5.4. Each user equipment is connected to the base station (eNB) through a wireless channel. Each eNB is connected to the cellular network gateway through a wired link. The cellular gateways are connected to each other through a wired backbone link. In this work, the wired links are assumed error

free. The sources of packet losses are the two wireless links between each user equipment and the eNB. Since, there are two wireless links, so this scenario is more challenging than the scenario given in Figure 4.7. Also, this scenario is most commonly used for mobile communication. The goal of the experiments is to show that the proposed **Adaptive** method performs better than the **Fixed** method when higher transmission rate is available. Higher transmission rate allows more retransmissions to be sent for a given delay.

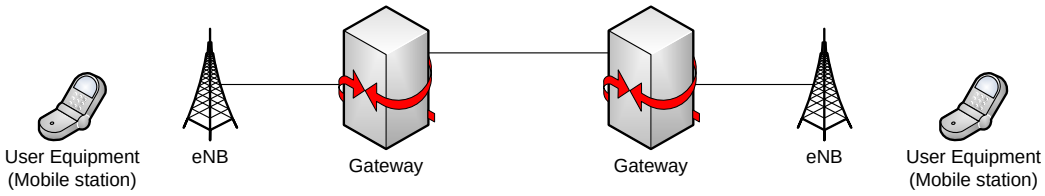


Figure 5.4: Network topology.

- **Video coding parameters:** The video coding parameters are the same as given in Section 4.3.

- **LT coding parameters:** The LT source block consists of NAL units of 2 video frames. The redundancy for the **Fixed** LT coding method is set to 35%. The initial redundancy for the **Adaptive** method is set to 17%. Expanding Factor of 8 is used. LT code symbol size is 1 bit. Robust soliton distribution is used. The value of LT codes constant c is 0.1 and δ is 0.5. A real implementation of LT codes is used. LT decoding failure deadline is 30 ms for the *Stunt* sequence and 40 ms for the *Party* sequence.

- **Packetization parameters:** The maximum size of the RTP/UDP/IP packet is $40 + 7 + 306 = 353$ bytes and each RTP/UDP/IP packet has a maximum of 306 bytes for LT encoded symbols. RTP/UDP/IP packet can contain a maximum of $306 \times 8 = 2448$ LT symbols. The number of RTP/UDP/IP packets required for k_s symbols are p. First p-1 RTP/UDP/IP packets contain 2448 LT encoded symbols each. The last of the p packet may contain less than 2448 symbols. The size of each second round RTP/UDP/IP packet is 353 bytes and contains 2448 symbols. RTP/UDP/IP packet size of 353 bytes results in RLC-SDU size of 320 bytes, after compression of RTP/UDP/IP header and adding PDCP header.

- **Transmission parameters:** The maximum size of RLC-PDU is 320 bytes. Flexible RLC-PDU size is used. One RLC-SDU and hence one RTP/UDP/IP packet

is mapped to one RLC-PDU. Transmission time interval (TTI) is set to 10 ms. The maximum bit rate of the radio bearer is 256 kbit/s. One RLC-PDU is sent at a given TTI. The maximum forward trip time (FTT) and the maximum backward trip time (BTT) is set to 40 ms. This value of FTT and BTT is realistic for 3GPP Long Term Evolution (LTE) system [107].

- **Rayleigh fading wireless channel model and RLC-PDU losses:** The Rayleigh fading wireless channel model proposed in [38] is used. The Rayleigh fading results in RLC-PDU losses. In this model, RLC-PDU loss is approximated by 2-state Markov process. In the 2-state Markov process, the channel has 2 states (good, bad). In the good state, the RLC-PDU is assumed to be received correctly. In the bad state, the RLC-PDU is assumed to be lost. The parameters for the Rayleigh fading channel model are set as follows: Mobile velocity is 3 km/h (pedestrian), carrier frequency is 2 GHz (LTE) and TTI is 10 ms. The Rayleigh fading margin parameter F is selected to give steady-state RLC-PDU loss rates of 0%, 0.5%, 1%, 1.5% and 5% over 100000000 iterations. The same RLC-PDU loss rate is used in both wireless channels. Forward and backward traffic is subject to RLC-PDU losses.

The PSNR results for the *Stunt* sequence are given in Figure 5.5. At no loss, the PSNR results of the **Fixed** [11] and the proposed **Adaptive** methods are nearly the same. At 0% RLC-PDU loss rate, all LT blocks are decoded successfully. As the RLC-PDU loss rate increases, the **Adaptive** method achieves better PSNR results than the **Fixed** method. This is due to the fact that in the **Fixed** method, the LT code redundancy is fixed regardless of the RLC-PDU loss rate while in the **Adaptive** method the LT code redundancy is varied for each LT source block depending on the RLC-PDU loss rate. This results in higher LT decoding success rate in **Adaptive** method compared to the **Fixed** method, hence PSNR is improved. At RLC-PDU loss rate of 5% in each wireless link, the **Adaptive** method achieves PSNR gain of 4.13 dB compared to the **Fixed** method.

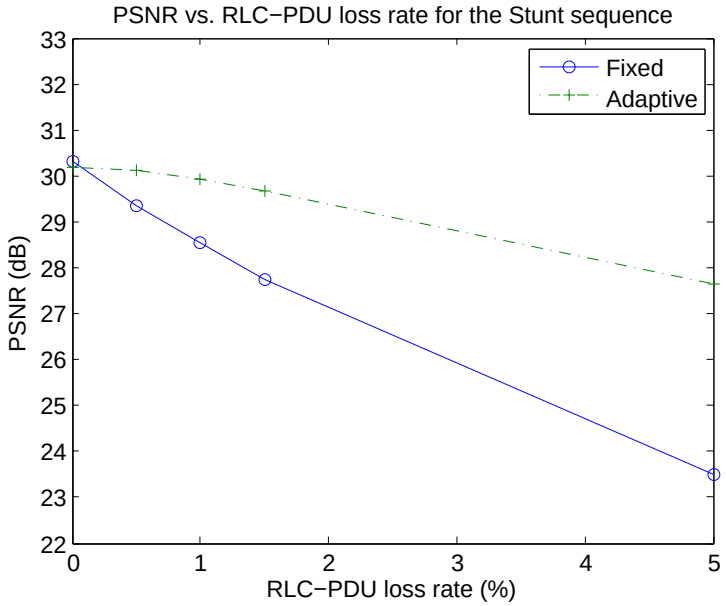


Figure 5.5: PSNR vs. RLC-PDU loss rate for the *Stunt* sequence.

The PDVD results for the *Stunt* sequence are given in Figure 5.6. The PDVD results of the **Fixed** and **Adaptive** method are the same in error free case. The **Adaptive** method achieves better PDVD results than the **Fixed** method as the RLC-PDU loss rate increases. The reason for better PDVD results for **Adaptive** method is fewer LT decoding failures compared to the **Fixed** method. Thus, the **Adaptive** method reduces the error propagation compared to the **Fixed** method. At RLC-PDU loss rate of 5% in each wireless channel, the **Adaptive** method achieves PDVD improvement of 26.11% compared to the **Fixed** method.

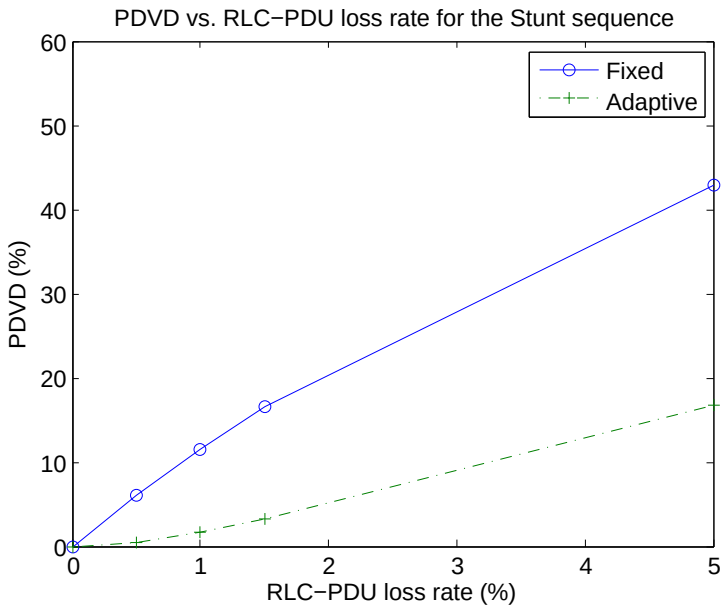


Figure 5.6: PDVD vs. RLC-PDU loss rate for the *Stunt* sequence.

The bit rate curves for the *Stunt* sequence are given in Figure 5.7. In the error free case, the bit rate of the **Fixed** method is higher than that of the **Adaptive** method. This is due to the fact that the LT code redundancy of the **Fixed** method is fixed to 35% while the initial LT code redundancy of the **Adaptive** method is 17%. As the RLC-PDU loss rate increases, the bit rate of the **Fixed** method does not change noticeably while the bit rate of the **Adaptive** method increases noticeably. The increase in bit rate of the **Adaptive** method is due to increased number of restranmissions, at higher RLC-PDU loss rates. Thus, in the error free case, the **Adaptive** method saves bandwidth compared to the **Fixed** method while giving same PSNR and PDVD results. In case of RLC-PDU losses, the **Adaptive** method gives better PSNR and PDVD results than the **Fixed** method, at lower bit rate.

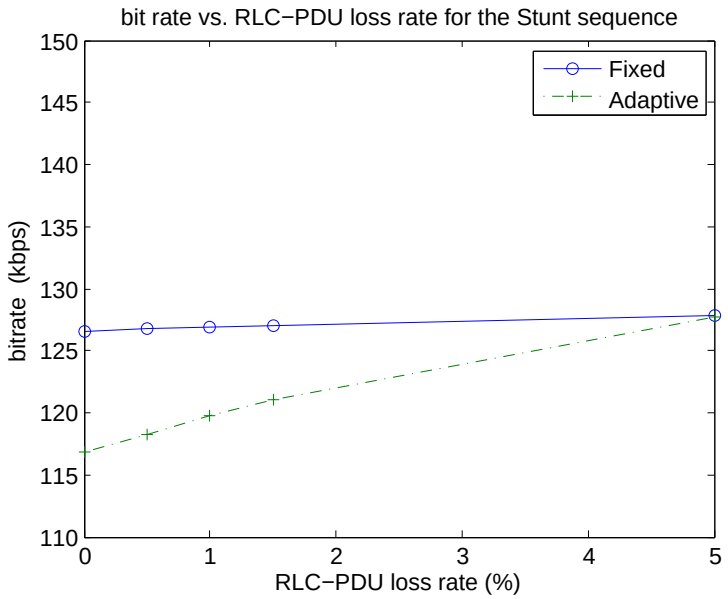


Figure 5.7: Bit rate vs. RLC-PDU loss rate for the *Stunt* sequence.

The PSNR, PDVD and bit rate curves for the *Party* sequence are given in Figure 5.8, Figure 5.9 and Figure 5.10, respectively. The PSNR, PDVD and bit rate results for the *Party* sequence are similar to that of the *Stunt* sequence. At RLC-PDU loss rate of 5% in each wireless channel, the **Adaptive** method achieves PSNR improvement of 3.98 dB and PDVD improvement of 28.62% compared to the **Fixed** method.

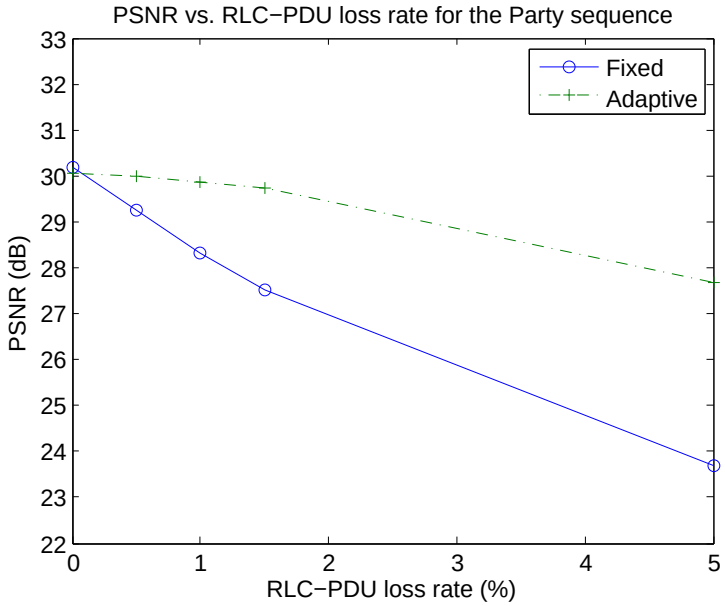


Figure 5.8: PSNR vs. RLC-PDU loss rate for the *Party* sequence.

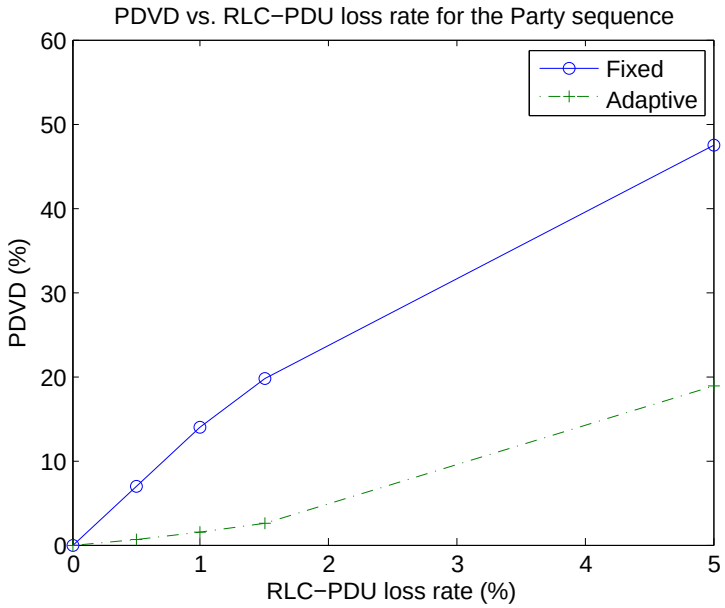


Figure 5.9: PDVD vs. RLC-PDU loss rate for the *Party* sequence.

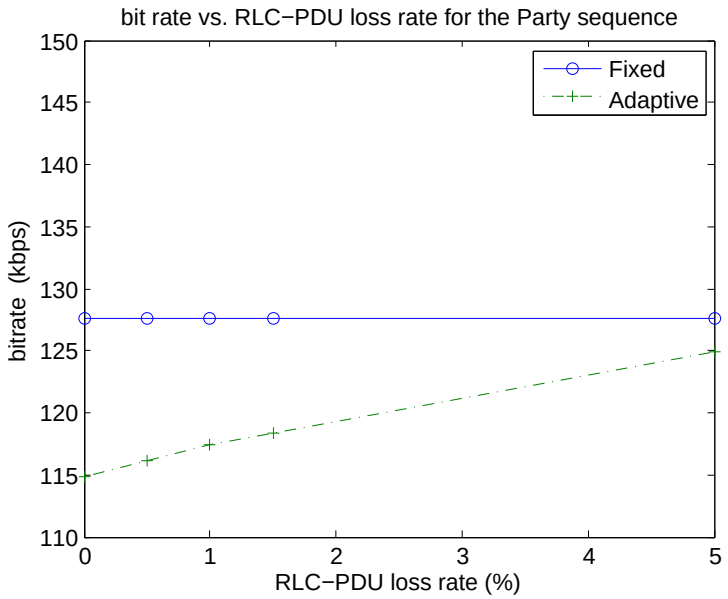
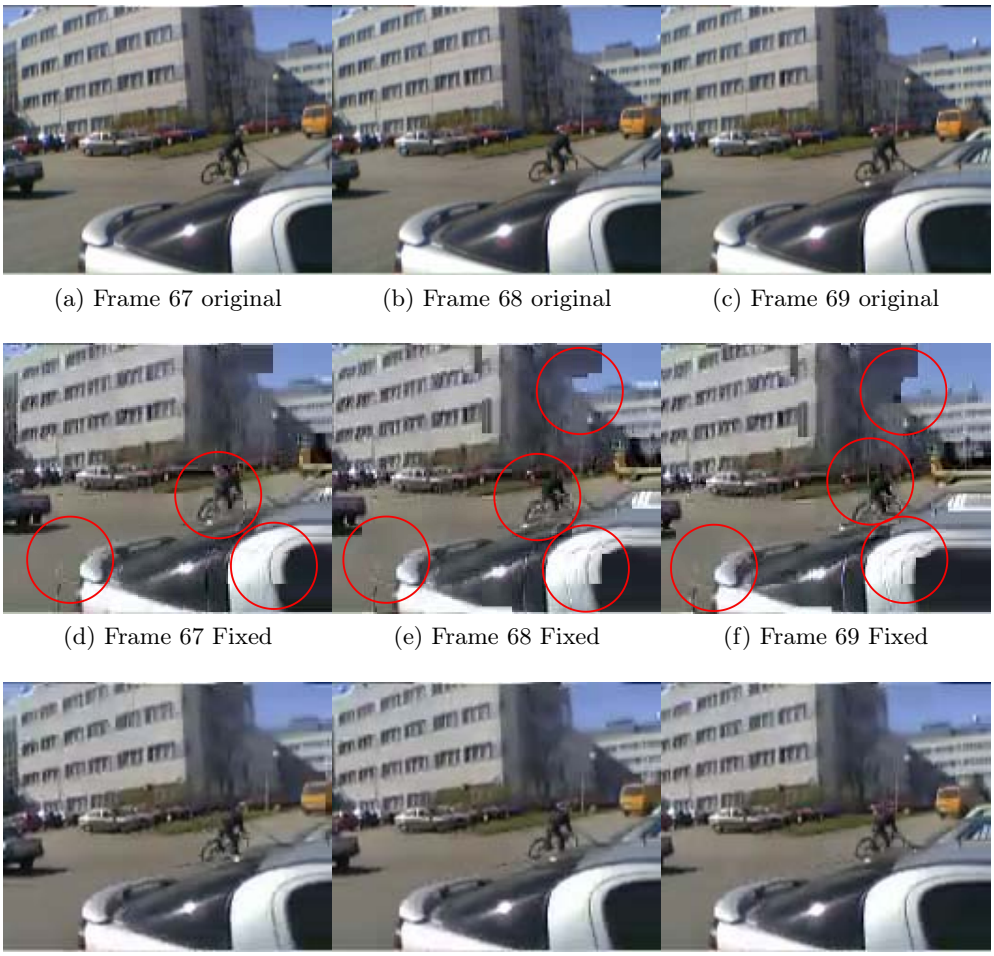


Figure 5.10: Bit rate vs. RLC-PDU loss rate for the *Party* sequence.

5.6 Visual results

Visual results are given in Figure 5.11 and Figure 5.12. The RLC-PDU loss rate is 5% in each wireless link. A real implementation of LT code is used. The visual results follow the objective results. The figures show that the **Adaptive** method can achieve better visual results than the **Fixed** method. The degradation in the video when using the **Fixed** method is shown in red circles. This degradation is due to LT decoding failure and resulting spatio-temporal error propagation. The LT decoding is successful for the **Adaptive** method for the corresponding frames. The LT decoding success is due to channel adaptivity of LT code redundancy.

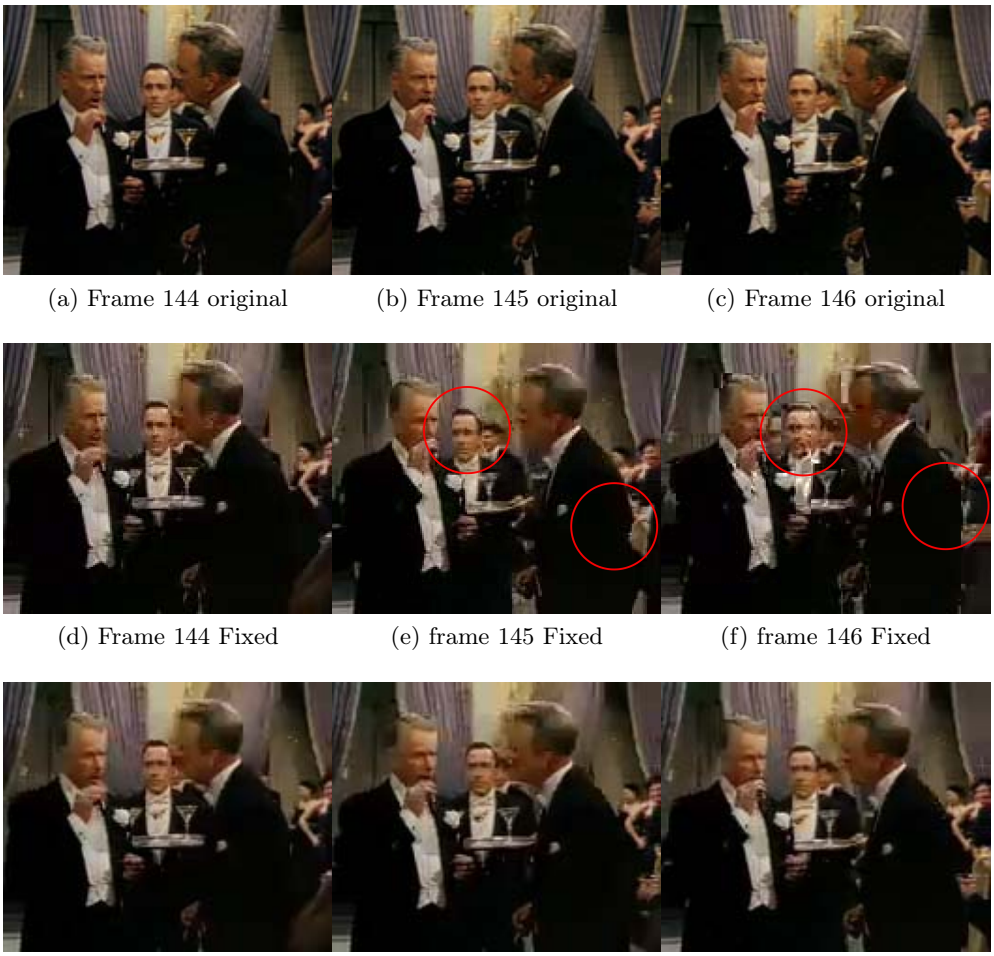


(a) Frame 67 original (b) Frame 68 original (c) Frame 69 original

(d) Frame 67 Fixed (e) Frame 68 Fixed (f) Frame 69 Fixed

(g) Frame 67 Adaptive (h) Frame 68 Adaptive (i) Frame 69 Adaptive

Figure 5.11: Visual results for the *Stunt* sequence



(a) Frame 144 original (b) Frame 145 original (c) Frame 146 original

(d) Frame 144 Fixed (e) frame 145 Fixed (f) frame 146 Fixed

(g) Frame 144 Adaptive (h) Frame 145 Adaptive (i) Frame 146 Adaptive

Figure 5.12: Visual results for the *Party* sequence

5.7 End-to-end delay analysis

The end-to-end delay components are given in Table 5.2. The end-to-end delay for the *Stunt* sequence is below 400 ms. The end-to-end delay for the *Party* sequence is 519 ms. The end-to-end delay can be reduced to below 400 ms by using an efficient implementation of LT codes or by using Raptor codes.

Delay Component	Stunt EF=8 Delay (ms)	Party EF=8 Delay (ms)
Frame buffering	66.6	83.3
H.264 encoding	4.6	4.6
LT encoding	< 1	< 1
Transmission delay	133.3	166.66
Propagation delay	40	40
LT decoding	139.2	223
H.264 decoding	1.6	1.6
Total delay	386.3	520.1

Table 5.2: End-to-end delay components for *Stunt* and *Party* video sequences for Expanding Factor (EF) 8. The total end-to-end delay is the sum of the end-to-end delay components.

5.8 Previous works

The state of the art channel adaptive transmission scheme is the N channel Stop and Wait (SaW) hybrid ARQ transmission scheme which is used at the physical layer of 3GPP HSPA and LTE standards [108]. In [97], a channel adaptive error resilience technique for real-time video communication is proposed.

5.9 Differences between the proposed method and state of the art methods

The proposed method has similarities with the method proposed in [97]. However, there are many differences between the proposed method and the method of [97]. The differences between the two methods are given below:

- Forward error correction is applied at the MAC layer in [97]. The forward error correction is applied at the application layer in the proposed method.
- Packet deadline is not used for declaring lost packets in [97]. In the proposed method, packet deadline is used to declare lost packets.
- Packet corruption/loss is detected by the MAC layer in [97]. Packet deadline is used to detect packet loss at the application layer in the proposed method.

The MAC layer packet corruption/loss detection as in [97] is valid for directly connected point to point links and MAC layer only.

- The method in [97] cannot be implemented at the application layer for video telephony over 3G networks. According to the 3G standard, corrupted RLC-PDUs (and RTP/UDP/IP packets) are discarded by the lower layers and not passed on to the application layer. The application layer has no knowledge of the corrupted packets. This is the reason for not comparing the results of [97] with the proposed **Adaptive** method.

- The transmission strategy in the proposed method is different from [97]. In [97], the transmitter sends initial round packets and a RACK frame and then stops and waits for RACK reply. It sends retransmissions after getting the results of the RACK frame. In the proposed method, the transmitter keeps on sending initial round packets and then non-LT packets.

- The encoder at the transmitter side uses only positively acknowledged frames as reference in [97]. The transmitter can use any frame as reference in the proposed method. The reference frame for reference picture selection is explicitly marked by the receiver in the proposed method.

- In the proposed method, the reference picture selection and NACK feedback information is sent in packets containing encoded symbols as well as in packets that do not contain the encoded symbols. This is done to ensure robustness and early reception of the feedback information for the transmitter. In [97], the NACK feedback information is not sent in packets containing encoded symbols. The NACK feedback is sent in a dedicated RACK frame.

- The packet loss detection is used for the initial round packets only in the proposed method. This is because the receiver knows the exact number of packets and also the time at which these packets should be received. On the other hand, in the method proposed in [97], packet loss detection is used for every transmission round.

- The packetization scheme in [97] is different from the packetization scheme of the proposed **Adaptive** method.

- Rateless codes are used in the proposed method. In [97], Reed-Solomon codes are used.

- Encoded symbols are packed in packets of fixed size in [97]. In the proposed method, the encoded symbols are packed in packets of variable size.
- Video frame skipping is used in [97]. Whereas, video frame skipping is not used in the proposed method.
- Many reference frames are used in [97]. Only one reference frame is used in the proposed method.
- In [97], if all the reference frames are corrupt, then Intra frame coding is used. In the proposed method, only inter coding is used.
- The method in [97] was proposed for one way video communication.

The transmission scheme used in the proposed method has similarities with the N channel Stop and Wait (SaW) hybrid ARQ scheme that is used in the 3GPP HSPA and LTE standards [107]. The differences between the transmission scheme used in the proposed method and the transmission scheme used in the hybrid ARQ of 3GPP HSPA and LTE standards, are given below:

- In the transmission strategy of the proposed method, rateless codes are used. In 3GPP HSPA and LTE hybrid ARQ, turbo codes are used.
- The NACK feedback is contained in the backward data packet in the transmission strategy of the proposed method. In 3GPP HSPA and LTE standard hybrid ARQ, the feedback channel is separate from the data channel.
- The feedback consists of 1 bit Positive Acknowledgment (ACK) or Negative Acknowledgment (NACK) in 3GPP HSPA and LTE hybrid ARQ. Only Negative Acknowledgment (NACK) is used in the transmission strategy of the proposed method.
- The NACK is cumulative in the transmission strategy of the proposed method. In the 3GPP HSPA and LTE hybrid ARQ transmission strategy, the NACK is not cumulative. Cumulative NACK results in robust and early reception of NACK feedback for retransmission.
- The packet deadline at the receiver is for the initial round packets only in the transmission strategy of the proposed method. The packet deadline at the transmitter is for every packet, either first round or second round in the 3GPP HSPA and LTE hybrid ARQ.

- The maximum number of hybrid ARQ channels are fixed to 8 in 3GPP HSPA and LTE hybrid ARQ. In the transmission strategy of the proposed method, the maximum number of hybrid ARQ channels are not fixed.
- The initial forward error correction redundancy in the proposed method is just enough to decode with high probability, when the channel is error free. The initial forward error correction code rate is fixed to 1/3 in the 3GPP HSPA and LTE hybrid ARQ.

5.10 Summary

In this chapter, a channel adaptive error resilience technique for video telephony over 3G networks has been proposed. The proposed **Adaptive** [12] error resilience technique is a combination of application layer forward error correction using rateless codes, reference picture selection and automatic repeat request. In the proposed technique, the redundancy of the LT code is dynamically adapted according to the packet loss rate in the wireless channel. When the packet loss rate is high, then high LT code redundancy is used and vice-versa. The proposed **Adaptive** error resilience technique achieves PSNR gain of up to 4.13 dB and PDVD gain of up to 28.62% compared to the **Fixed** error resilience scheme proposed in Chapter 4 [11]. This improvement in results is obtained at lower or equal bit rate compared to the **Fixed** method. Also, a RTP/UDP/IP packetization scheme is proposed to facilitate the implementation of the proposed **Adaptive** error resilience technique in practical scenarios. Also, a comparison of the proposed method was done with the method presented in [97]. The method of [97] cannot work at application layer. So it is not suitable for providing application layer error control for packet switched mobile video telephony.

Chapter 6

Early reference picture selection

6.1 Introduction

In the methods presented in Chapters 4 and 5, the LT decoding fails when a certain number of encoded symbols are lost during transmission. The feedback for the reference picture selection is generated after the LT decoding failure deadline has passed. The feedback informs the transmitter that LT decoding has failed at the receiver and the H.264 encoder at the transmitter should invoke the reference picture selection. The reference picture selection is invoked to stop spatio-temoporal error propagation. However, the time between encoding of a video frame and the reception of feedback for source block containing that frame could be too long. During that time, the H.264 encoder may have encoded many video frames.

In case of LT decoding failure, the errors due to spatio-temporal error propagation may have propagated to many frames, before the reference picture selection can be invoked by the H.264 encoder at the transmitter side. However, the failure of LT decoding at the receiver can be predicted much earlier by the transmitter. In that case, the reference picture selection can be invoked earlier at the transmitter.

The advantage of invoking reference picture selection early is that the number of frames that are corrupted due to LT decoding failure, is reduced. This reduces spatio-temporal error propagation and results in PSNR and PDVD improvement compared to the methods where the reference picture selection is invoked by the feedback generated after the LT decoding failure deadline.

6.2 System description

The system block diagram is the same as in Figure 4.1.

6.3 Fixed redundancy rateless coding and early reference picture selection

The proposed method is a combination of the **Fixed** method proposed in Chapter 4 and early reference picture selection. The description of the method is given in the following. The proposed method is called **Fixed early RPS** in this dissertation.

The transmission strategy of the proposed method is shown in Figure 6.1. It is assumed that the transmitter and receiver clocks are synchronized. Each user equipment sends and receives video simultaneously. The transmission start time of the corresponding source block at each user equipment is the same.

The transmission sending deadline and the transmission receiving deadline is determined as in Section 4.2.3.

The number of encoded symbols k_s for the current source block is determined by:

$$k_s = k \times (1 + r)$$

where k_s is the number of source symbols for the current source block and r is the rateless code redundancy. The rateless code redundancy is fixed.

k_s symbols are packed in RTP/UDP/IP packets. The maximum size m of the RTP/UDP/IP packet is chosen such that it is mapped to RLC-PDU size of exactly 320 bytes. k_s symbols are sent in p RTP/UDP/IP packets with $p = \frac{k_s}{m}$. The first p-1 packets have the same size, which is equal to the maximum RTP/UDP/IP packet size. The last packet size may be smaller than the first p-1 packets. RTP/UDP/IP packets are mapped to RLC-PDU for transmission. Flexible RLC-PDU size is used. The flexible RLC-PDU size allows RTP/UDP/IP packet of any size to be mapped onto exactly one RLC-PDU. In flexible RLC-PDU size, the size of the RLC-PDU can be from 10 bytes to 1500 bytes [107]. One RLC-PDU can be sent at a given TTI.

The Transmission Time Interval (TTI) is fixed. At t=0, the LT source block is available for transmission. At t=0, the transmitter sends the first packet. At t = TTI, the transmitter sends the next packet. For each $t \geq 1$, the transmitter sends the nth packet at $t = (n-1) \times$ TTI. The total number of packets sent is p. These p packets are shown in green in Figure 6.1.

The packet deadline scheme given in Section 5.2.2 is used at the receiver. Cumulative NACK is used for reporting lost packets to the transmitter.

The transmitter defines a threshold of number of lost symbols t_l as:

$$t_l = k_s - k$$

The transmitter checks for NACK in the backward packets at every TTI. It determines the cumulative number of forward packets lost. From the cumulative number of forward packets lost, the transmitter calculates the total number of encoded symbols lost. The transmitter keeps a record of the total number of forward packets it has sent and also the total number of encoded symbols sent for the current source block.

If the number of encoded symbols that are lost exceeds t_l, then the transmitter concludes that the rateless code decoding at the receiver will fail with 100% certainty. The transmitter invokes reference picture selection as soon as the above condition becomes true. In Figure 6.1, this condition becomes true when NACK=3 is received at the transmitter. This is different from Figure 4.3 where reference picture selection was invoked after the reference picture selection feedback is received by the transmitter. Since the reference picture selection may be invoked earlier in this case, the number of frames corrupted by spatio-temporal propagation is reduced. This results in improved video quality.

Early abort

If the condition for the early reference picture selection becomes true or the transmitter has sent all p packets, then the transmitter sends only non-rateless packets at each TTI.

These non-rateless packets contain only reference picture selection information and negative acknowledgment (NACK) for the other user equipment. This is called early abort. These non-rateless packets are sent at every TTI, till the next source block is

available for transmission. As these packets do not contain encoded symbols, bandwidth is saved. The non-rateless packets are shown in grey colour in Figure 6.1. The packets that contain encoded symbols are called rateless packets.

The contents of the rateless packet and non-rateless packet are given in Table 6.1.

Rateless packet	Non-rateless packet
Contains encoded symbols, rateless code information, Negative Acknowledgment (NACK) and reference picture selection information	Contains Negative Acknowledgment (NACK) and reference picture selection information

Table 6.1: Contents of rateless packets and non-rateless packets.

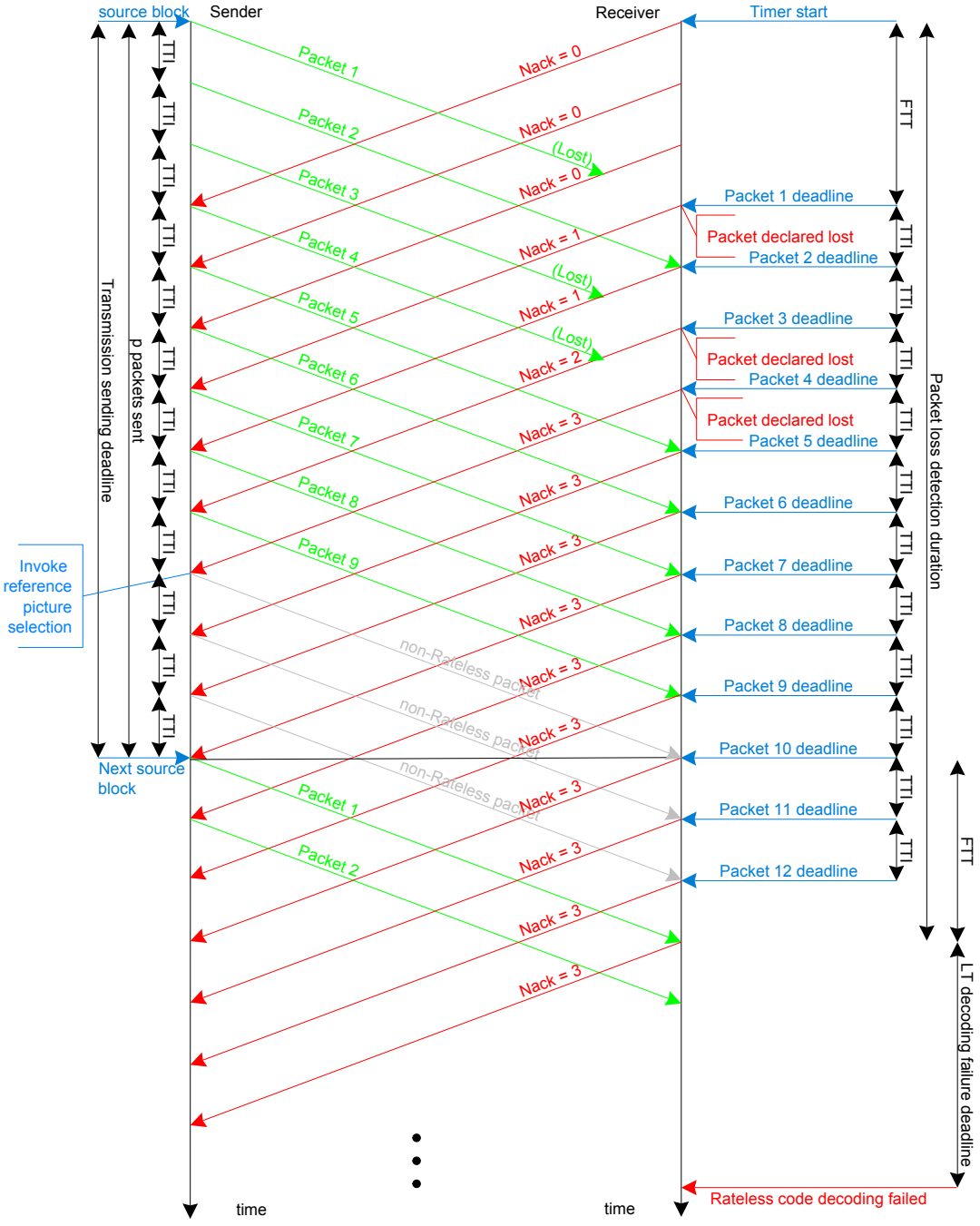


Figure 6.1: Transmission strategy of the proposed **Fixed early RPS** method.

6.3.1 Proposed packetization

The proposed packetization for the **Fixed early RPS** method is given in Figure 6.2. The following fields are present in the proposed packetization:

- **Sequence number:** Index of the first encoded symbol in the packet. The size of this field is 2 bytes.
- **Source block size:** The size of this field is 2 bytes.
- **Source block ID:** The size of this field is 4 bits.
- **Last successful block ID:** The source block ID of the last successfully decoded source block at the receiver. The size of this field is 3 bits.
- **Negative Acknowledgment (NACK):** This field contains the cumulative number of lost packets for the current source block. The size of this field is 8 bits. The first four bits of the field contains the NACK value of the current source block. The last 4 bits contains the NACK value for the previous source block. Each NACK value is retained for the duration of two source blocks. This is done to facilitate early reference picture selection and early abort.
- **Invoke RPS flag:** The size of this flag is one bit. When the receiver wants to invoke RPS explicitly, then this flag is set to 1, otherwise it is set to zero.

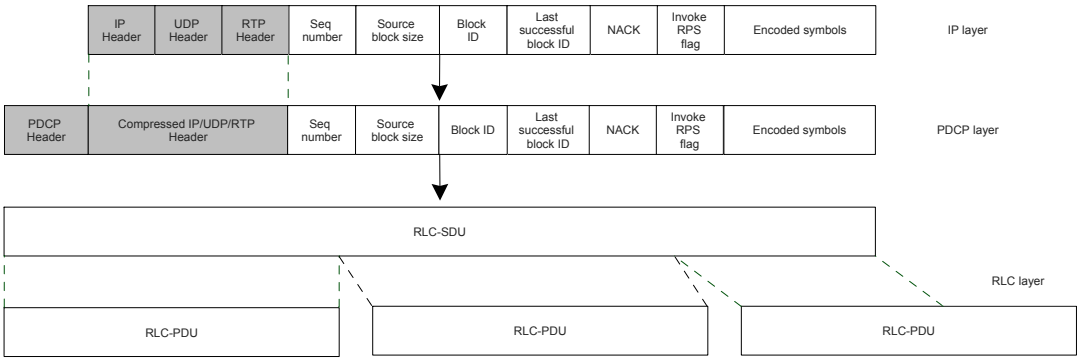


Figure 6.2: Proposed packetization of RTP/UDP/IP packet for early reference picture selection schemes.

6.3.2 Experimental results

The target 3GPP mobile cellular system is the Long Term Evolution (LTE) system. The network topology of the scenario is given in Figure 5.4. The goal of the experiments is to show that using early reference picture selection improves video quality.

- **Video coding parameters:** The frame rate of the *Stunt* video sequence is 30 frames per second and the frame rate of the *Party* sequence is 24 frames per second. The maximum frame rate specified by 3GPP for video telephony is 30 frames per second [22]. The compressed bit rate for both video sequences is 178.3 kbit/s. Early reference picture selection gives significant results at higher frame rate that is why frame rate of 24 fps and 30 fps are used. However, increasing frame rate decreases the PSNR of each frame and to compensate for that the bit rate has to be increased. This value of bit rate is chosen so that that LT encoding and decoding complexity is the same as in the **Fixed** and **Adaptive** method, when a LT symbol size of 2 bit is used. These values of source rate and LT symbol size is a trade off between LT decoding efficiency, computational complexity and end-to-end delay. The rest of the video coding parameters are the same as given in Section 4.3. The benefit of using early reference picture selection is evident when using higher frame rate. This is the reason for using higher frame rate. Since increasing frame rate decreases the video quality of each frame, so the bit rate is increased to get better video quality in error free conditions.

- **LT code parameters:** The LT source block consists of NAL units of 4 video frames. The LT code symbol size is 2 bit. These values of LT code parameters are chosen, so that the encoding and decoding complexity of the LT code is similar to that given in Section 5.7. The LT code redundancy is fixed to 25% for the *Stunt* sequence and 26% for the *Party* sequence. Expanding Factor of 8 is used. Robust Soliton distribution is used. The value of LT code constant c is 0.1 and δ is 0.5. A real implementation of LT code is used. The LT decoding failure deadline is 30 ms for the *Stunt* sequence and 40 ms for the *Party* video sequence. These values are obtained through simulations of the LT code implementation.

- **Raptor code parameters:** The Raptor code failure probability model given in [109] is used. For $k > 200$, the failure probability of the Raptor code is modeled by the following equation:

$$P_f(m,k) = \begin{cases} 1 & \text{if } m < k, \\ \\ 0.85 \times 0.567^{m-k} & \text{if } m \geq k, \end{cases}$$

where k is the number of source symbols and m is the number of received symbols. The decoding failure deadline for the Raptor code is the same as used for the LT code. The symbol size of the Raptor code is 2 bit.

- **Packetization parameters:** The maximum size of the RTP/UDP/IP packet is $40 + 6 + 307 = 353$ bytes and each RTP/UDP/IP packet has a maximum of 307 bytes for encoded symbols. RTP/UDP/IP packet contains a maximum of $307 \times 4 = 1228$ encoded symbols. RTP/UDP/IP packet size of 353 bytes results in RLC-SDU size of 320 bytes.

- **Transmission parameters:** The maximum size of RLC-PDU is 320 bytes. Flexible RLC-PDU size is used. One RLC-SDU and hence one RTP/UDP/IP packet is mapped to one RLC-PDU. Transmission Time Interval (TTI) is set to 10 ms. The maximum bit rate of the radio bearer is 256 kbit/s. One RLC-PDU is sent at a given TTI. The maximum forward trip time (FTT) and the maximum backward trip time (BTT) is 40 ms.

- **Rayleigh fading wireless channel model and RLC-PDU losses:** The Rayleigh fading wireless channel model proposed in [38] is used. The parameters for the Rayleigh fading channel model are set as follows: Mobile velocity is 3 km/h, carrier frequency is 2 GHz and TTI is 10 ms. The fade margin parameter F is selected to give steady-state RLC-PDU loss rates of 0%, 0.5%, 1%, 1.5% and 5% over 100000000 iterations. The same RLC-PDU loss rate is used in both wireless channels.

The following methods are compared:

- **Fixed** The LT and Raptor code redundancy rate is fixed.

- **Fixed early RPS** The LT and Raptor code redundancy rate is fixed. Early Reference Picture Selection is used.

The LT code PSNR results for the *Stunt* and *Party* video sequences are given in Figure 6.3 and Figure 6.4, respectively. The results show that as the RLC-PDU loss rate increases, the **Fixed early RPS** scheme achieves better PSNR results than the

Fixed scheme. At higher RLC-PDU loss rates, the early reference picture selection is invoked more often. At 5% RLC-PDU loss rate in each wireless link, the **Fixed early RPS** scheme achieves PSNR gain of 0.7 dB and 1.22 dB for *Stunt* and *Party* video sequences, respectively. This shows that using early reference picture selection improves video quality. The use of early rference picture selection reduces the number of video frames that are corrupted by spatio-temporal error propagation.

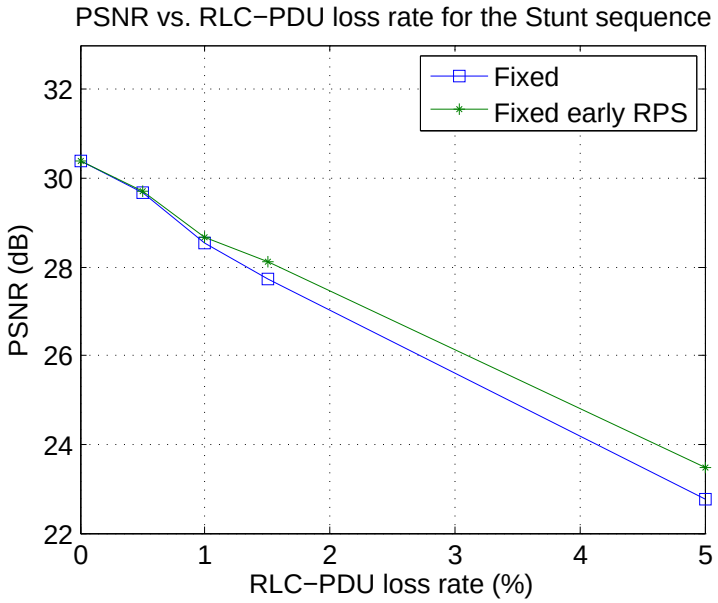


Figure 6.3: PSNR vs. RLC-PDU loss rate for the *Stunt* sequence using LT code.

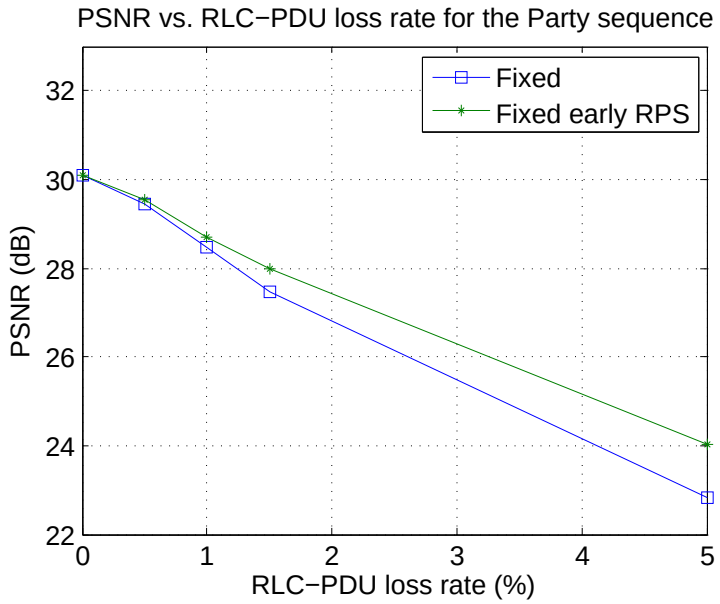


Figure 6.4: PSNR vs. RLC-PDU loss rate for the *Party* sequence using LT code.

The Raptor code PSNR results for the *Stunt* and *Party* video sequences are given in Figure 6.5 and Figure 6.6, respectively. The trend in the results is similar to the LT code results.

At 5% RLC-PDU loss rate in each wireless link, the **Fixed early RPS** scheme achieves PSNR gain of 1.31 dB and 1.27 dB for *Stunt* and *Party* video sequences, respectively.

The PSNR results of Raptor code are higher than the corresponding PSNR results of the LT code. This is due to the fact that Raptor code is more efficient than LT code. Raptor code has higher decoding success probability than LT code for the same number of received encoded symbols.

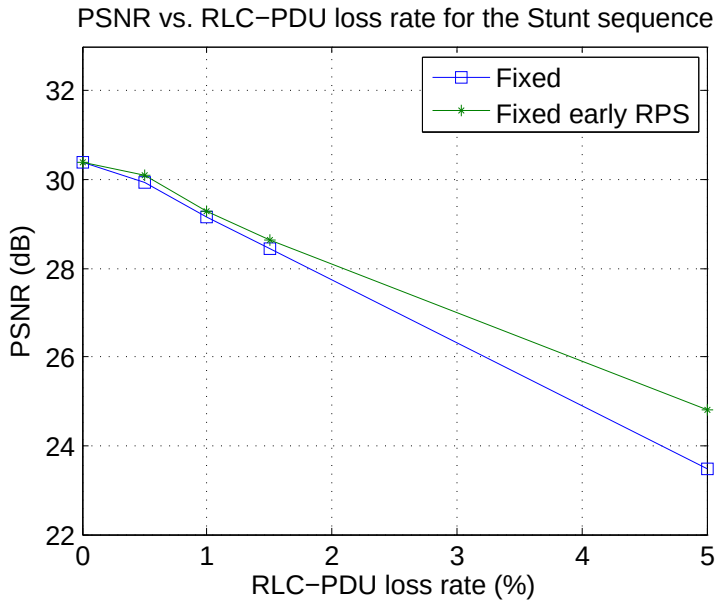


Figure 6.5: PSNR vs. RLC-PDU loss rate for the *Stunt* sequence using Raptor code.

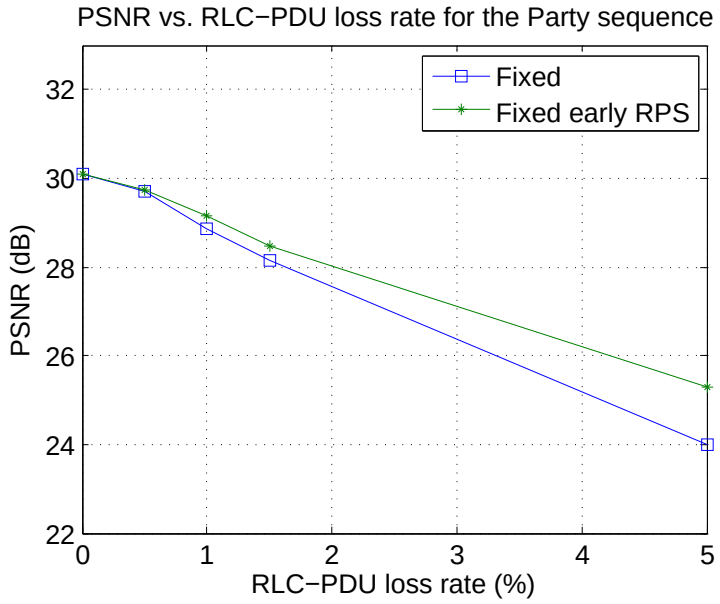


Figure 6.6: PSNR vs. RLC-PDU loss rate for the *Party* sequence using Raptor code.

The LT code PDVD results for the *Stunt* and *Party* video sequences are given in Figure 6.7 and Figure 6.8, respectively. The results show that as the RLC-PDU loss rate increases, the **Fixed early RPS** scheme achieves better PDVD results than the

Fixed scheme. At 5% RLC-PDU loss rate in each wireless link, the **Fixed early RPS** scheme achieves PDVD gain of 5.63% and 9.69% for *Stunt* and *Party* video sequences, respectively. The use of early reference picture selection reduces the number of video frames that are corrupted by spatio-temporal error propagation.

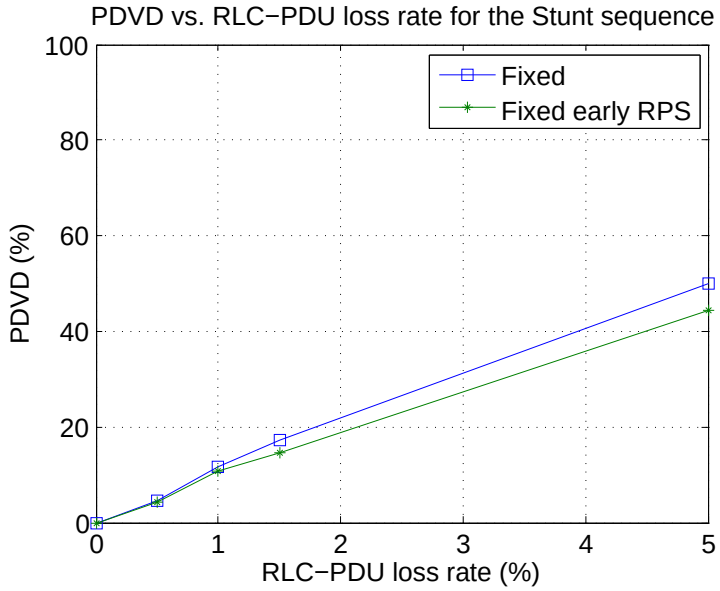


Figure 6.7: PDVD vs. RLC-PDU loss rate for the *Stunt* sequence using LT code.

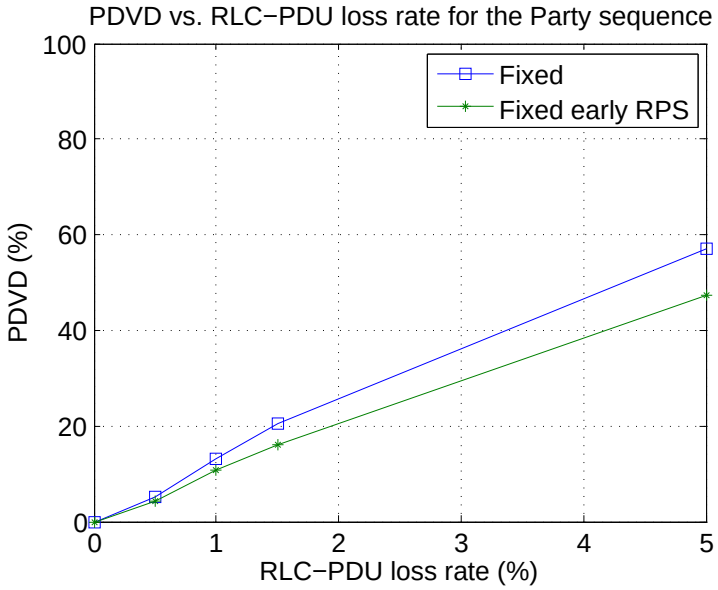


Figure 6.8: PDVD vs. RLC-PDU loss rate for the *Party* sequence using LT code.

The Raptor code PDVD results for the *Stunt* and *Party* video sequences are given in Figure 6.9 and Figure 6.10, respectively. The trend in the results is similar to the LT code results. At 5% RLC-PDU loss rate in each wireless link, the **Fixed early RPS** scheme achieves PDVD gain of 7.95% and 10.67% for *Stunt* and *Party* video sequences, respectively. Also, the PDVD results of the Raptor code are lower than the corresponding PDVD results of the LT code.

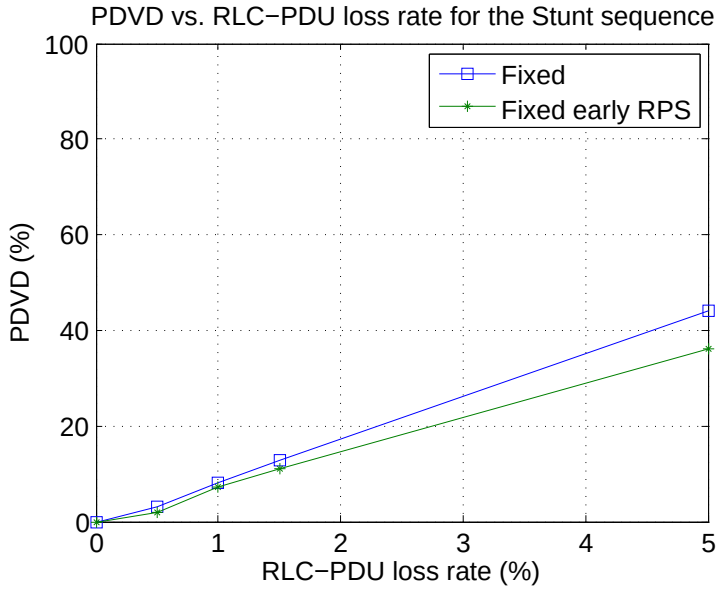


Figure 6.9: PDVD vs. RLC-PDU loss rate for the *Stunt* sequence using Raptor code.

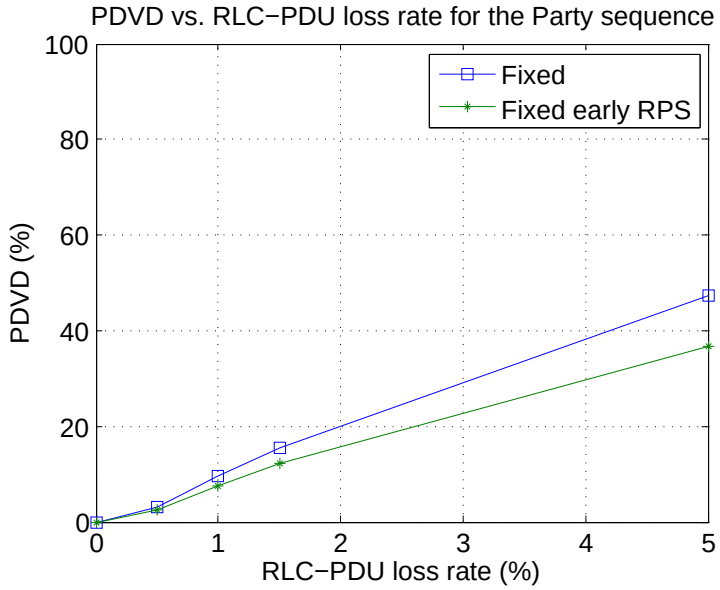


Figure 6.10: PDVD vs. RLC-PDU loss rate for the *Party* sequence using Raptor code.

The bit rate curves for the *Stunt* and *Party* video sequences for the LT code are given in Figure 6.11 and Figure 6.12, respectively. The **Fixed early RPS** scheme saves bandwidth compared to the **Fixed** scheme. The bandwidth saving is more at higher

RLC-PDU loss rate. At higher RLC-PDU loss rate, early RPS is invoked more often.

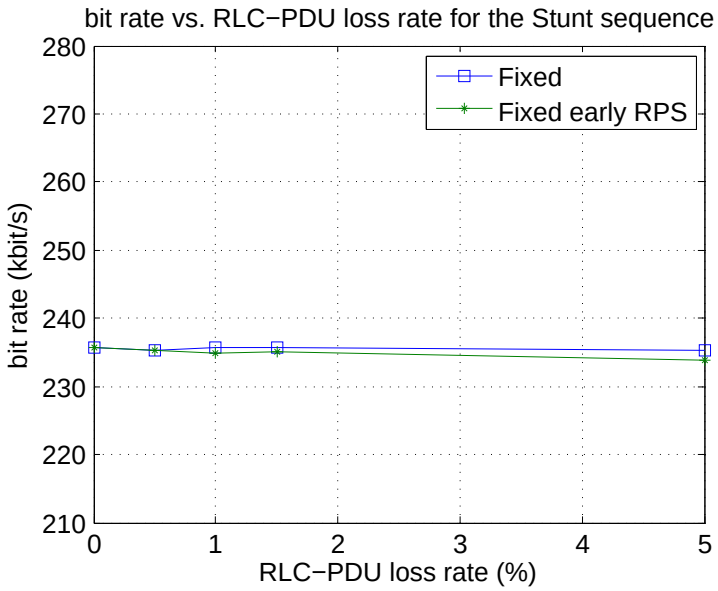


Figure 6.11: Bit rate vs. RLC-PDU loss rate for the *Stunt* sequence using LT code.

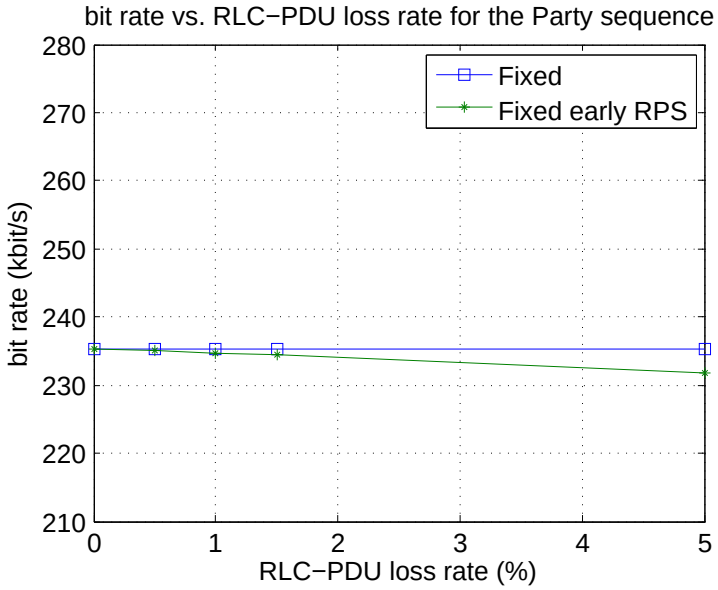


Figure 6.12: Bit rate vs. RLC-PDU loss rate for the *Party* sequence using LT code.

The bit rate curves for the *Stunt* and *Party* video sequences for the Raptor code are given in Figure 6.13 and Figure 6.14, respectively. The Raptor code results are similar

to the LT code results.

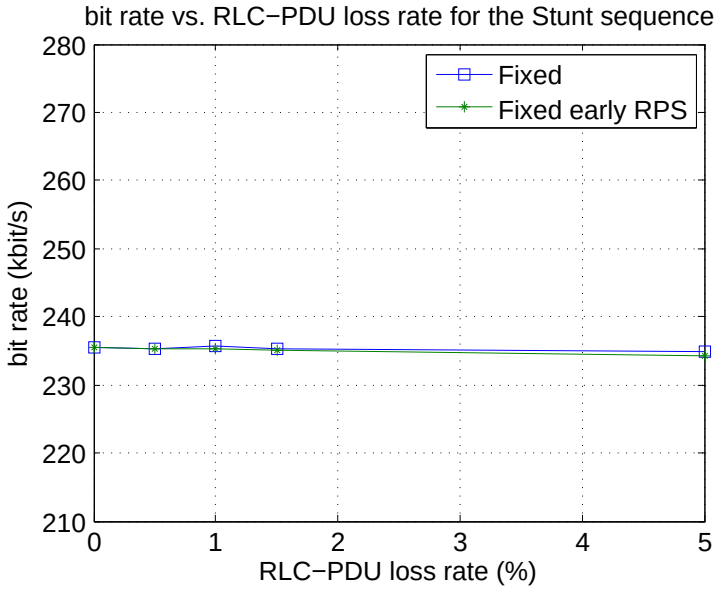


Figure 6.13: Bit rate vs. RLC-PDU loss rate for the *Stunt* sequence using Raptor code.

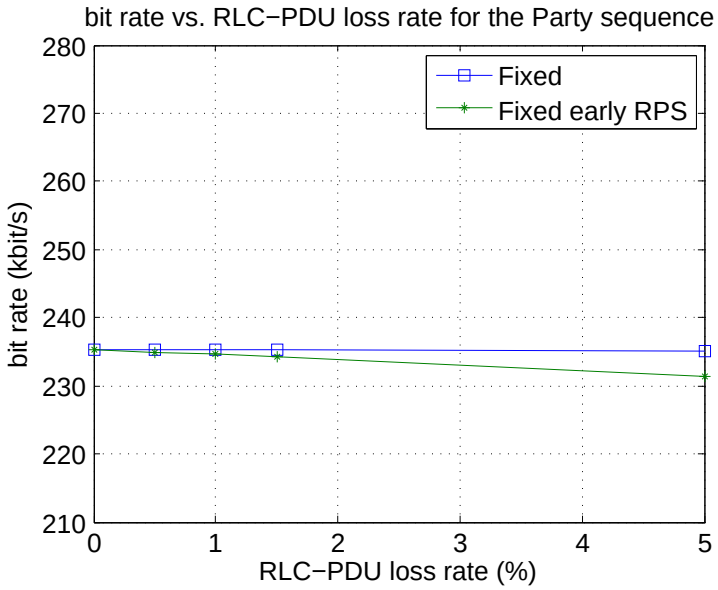


Figure 6.14: Bit rate vs. RLC-PDU loss rate for the *Party* sequence using Raptor code.

6.4 Channel adaptive rateless coding and early reference picture selection

The transmission strategy of the proposed method is shown in Figure 6.15. It is assumed that the transmitter and receiver clocks are synchronized. Each user equipment sends and receives video simultaneously. The transmission start time of the source block for both user equipments is the same. The transmission sending deadline and the transmission receiving deadline of a source block is determined as in Section 4.2.3. The maximum forward trip time is denoted by FTT and the maximum backward trip time is denoted by BTT. The number of initial encoded symbols k_s and the number of initial packets p are determined as in Section 5.2.2. The deadline scheme for the initial packets is also the same as in Section 5.2.2. The symbols lost threshold t_l is determined as in Section 6.3. The transmitter checks for the number of lost symbols at every TTI. If the number of lost symbols exceeds the symbols lost threshold t_l, then the reference picture selection is invoked at the transmitter as in Section 6.3. In Figure 6.15, this condition becomes true when NACK=4 is received at the transmitter. This is different from the case in **Adaptive** scheme given in Section 5.2, where the reference picture selection is invoked on reception of the reference picture selection feedback. Hence, the reference picture selection can be invoked earlier than the **Adaptive** method.

Early abort

If the condition for early reference picture selection becomes true, then the transmitter sends non-rateless packets till the end of the transmission sending deadline for the current source block. Since these packets do not contain any encoded symbols, bandwidth is saved. These packets are not shown in Figure 6.15 for simplicity.

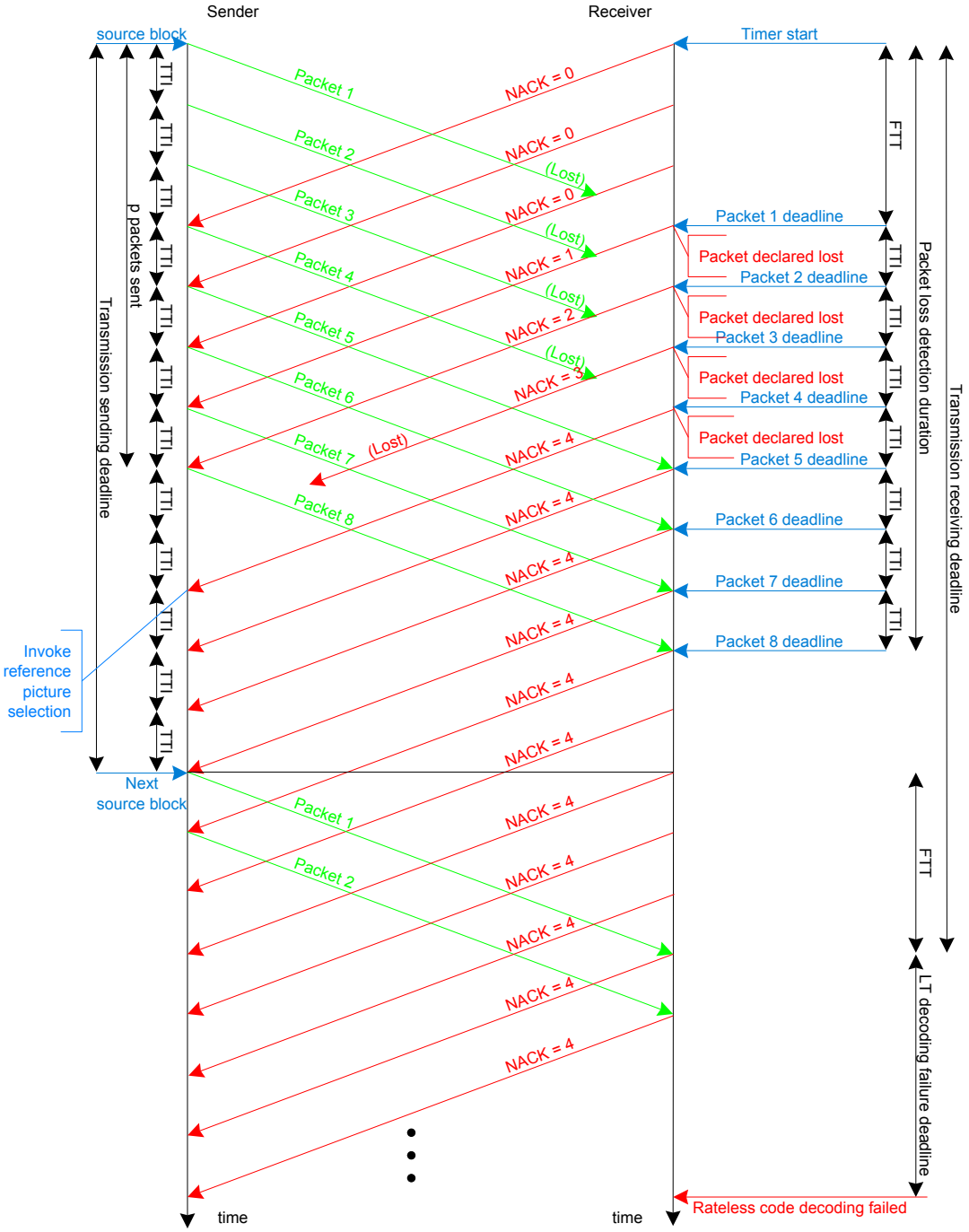


Figure 6.15: Transmission strategy of the proposed **Adaptive early RPS** method.

6.4.1 Proposed packetization

The packetization of the RTP/UDP/IP packet for the **Adaptive early RPS** scheme is the same as in Figure 6.2.

6.4.2 Experimental results

The video telephony scenario and network topology is the same as in Section 5.5. The initial LT and Raptor code redundancy is set to 17%. The rest of the parameters are the same as in Section 6.3.2. The goal of the experiments is to show that using early reference picture selection improves video quality.

The following methods are compared:

- **Adaptive** The LT and Raptor code redundancy is channel adaptive.
- **Adaptive early RPS** The LT and Raptor code redundancy is channel adaptive. Early Reference Picture Selection is used.

The LT code PSNR results for the *Stunt* and *Party* video sequences are given in Figure 6.16 and Figure 6.17, respectively. The results show that as the RLC-PDU loss rate increases, the **Adaptive early RPS** scheme achieves better PSNR results than the **Adaptive** scheme. At 5% RLC-PDU loss rate in each wireless link, the **Adaptive early RPS** scheme achieves PSNR gain of 0.86 dB and 1.14 dB for *Stunt* and *Party* video sequences, respectively. This shows that using early reference picture selection improves video quality.

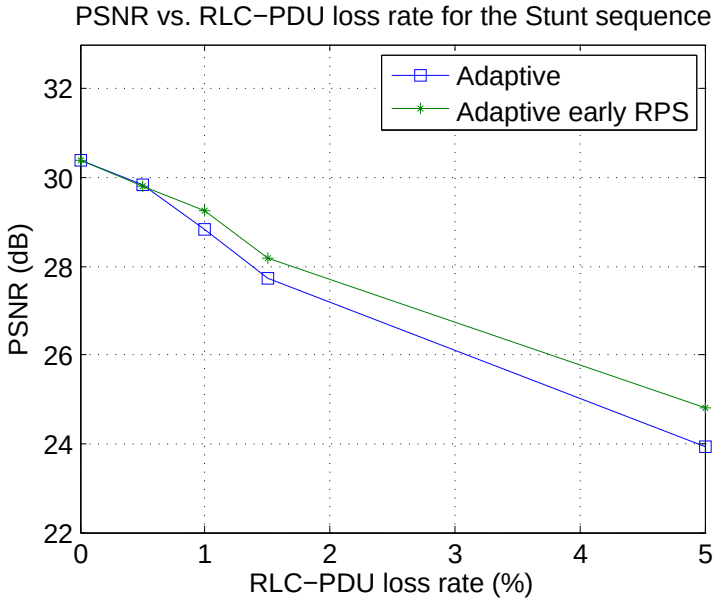


Figure 6.16: PSNR vs. RLC-PDU loss rate for the *Stunt* sequence using LT code.

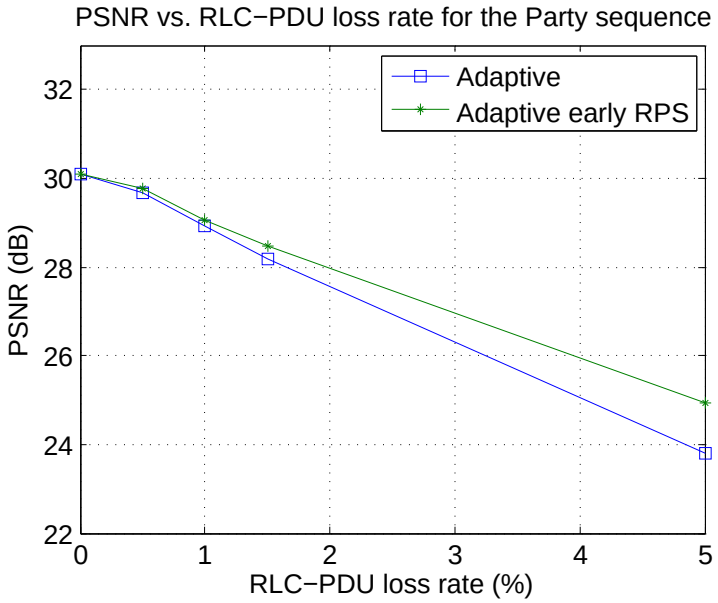


Figure 6.17: PSNR vs. RLC-PDU loss rate for the *Party* sequence using LT code.

The Raptor code PSNR results for the *Stunt* and *Party* video sequences are given in Figure 6.18 and Figure 6.19, respectively. The trend in the results is similar to the LT code results. At 5% RLC-PDU loss rate in each wireless link, the **Adaptive early**

RPS scheme achieves PSNR gain of 0.89 dB and 0.77 dB for *Stunt* and *Party* video sequences, respectively.

The PSNR results of the Raptor code are higher than the corresponding PSNR results of the LT code.

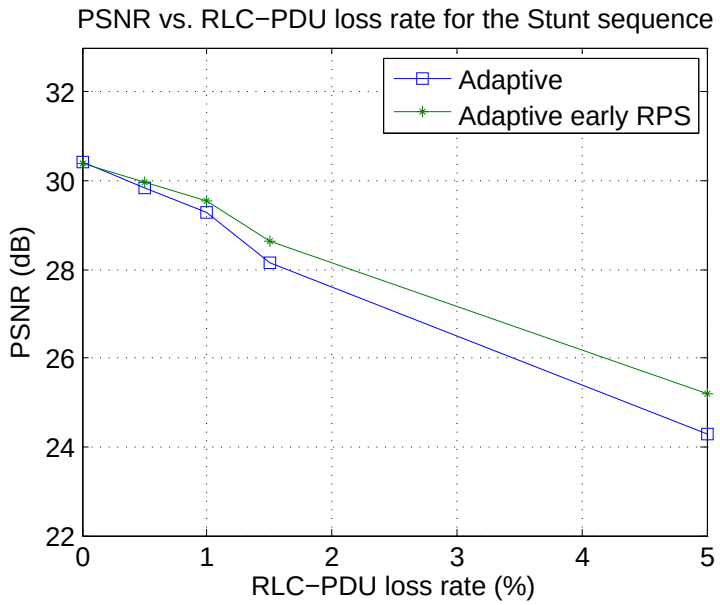


Figure 6.18: PSNR vs. RLC-PDU loss rate for the *Stunt* sequence using Raptor code.

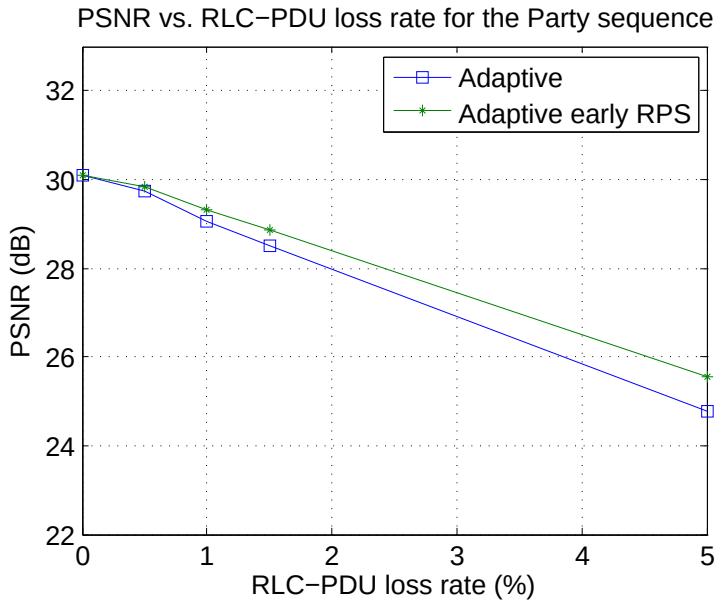


Figure 6.19: PSNR vs. RLC-PDU loss rate for the *Party* sequence using Raptor code.

The LT code PDVD results for the *Stunt* and *Party* video sequences are given in Figure 6.20 and Figure 6.21, respectively. The results show that as the RLC-PDU loss rate increases, the **Adaptive early RPS** scheme achieves better PDVD results than the **Adaptive** scheme. At 5% RLC-PDU loss rate in each wireless link, the **Adaptive early RPS** scheme achieves PDVD gain of 6.47% and 7.87% for *Stunt* and *Party* video sequences, respectively. The use of early reference picture selection reduces the number of video frames that are corrupted by spatio-temporal error propagation.

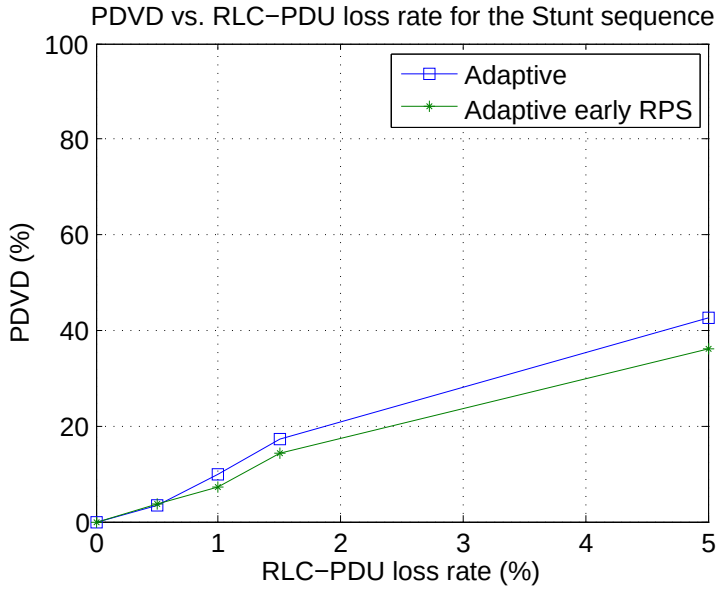


Figure 6.20: PDVD vs. RLC-PDU loss rate for the *Stunt* sequence using LT code.

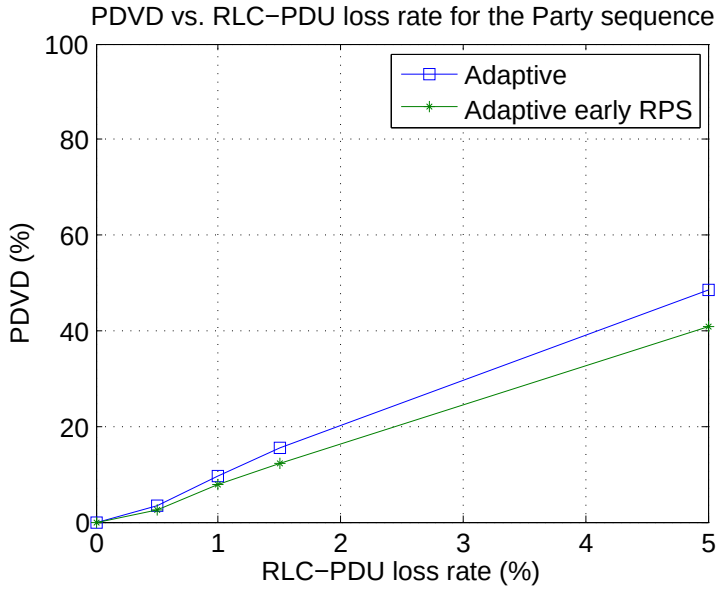


Figure 6.21: PDVD vs. RLC-PDU loss rate for the *Party* sequence using LT code.

The Raptor code PDVD results for the *Stunt* and *Party* video sequences are given in Figure 6.22 and Figure 6.23, respectively. The trend in the results are similar to the LT code results. At 5% RLC-PDU loss rate in each wireless link, the **Adaptive**

early RPS scheme achieves PDVD gain of 6.74% and 7.15% for *Stunt* and *Party* video sequences, respectively. Also, the PDVD results of the Raptor code are lower than the corresponding PDVD results of the LT code.

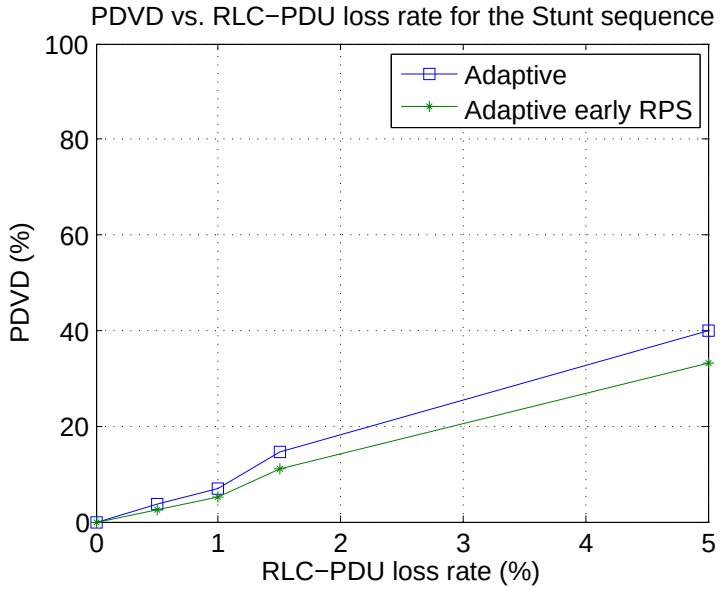


Figure 6.22: PDVD vs. RLC-PDU loss rate for the *Stunt* sequence using Raptor code.

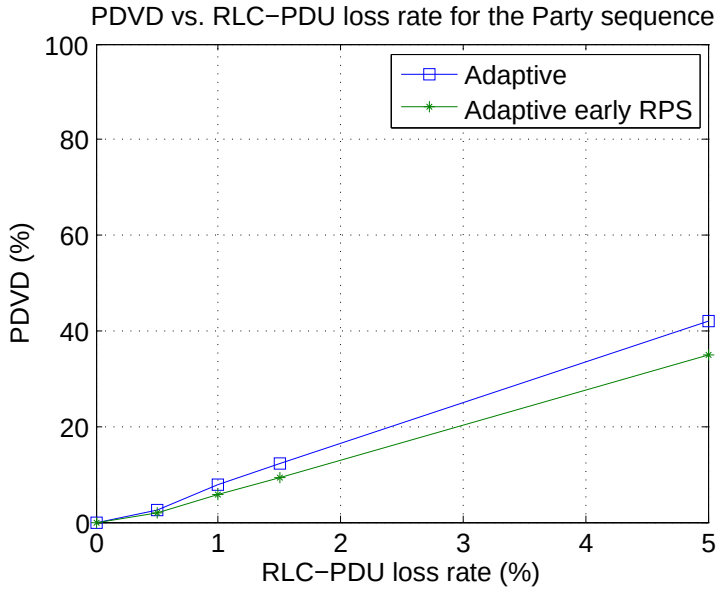


Figure 6.23: PDVD vs. RLC-PDU loss rate for the *Party* sequence using Raptor code.

The bit rate curves for the *Stunt* and *Party* video sequences for the LT code are given in Figure 6.24 and Figure 6.25, respectively. The **Adaptive early RPS** scheme saves bandwidth compared to the **Adaptive** scheme. The bandwidth saving is more at higher RLC-PDU loss rate. At higher RLC-PDU loss rate, the early RPS is invoked more often. Also, as the RLC-PDU loss rate increases, the bit rate also increase. This is because of more retransmissions at higher RLC-PDU loss rates.

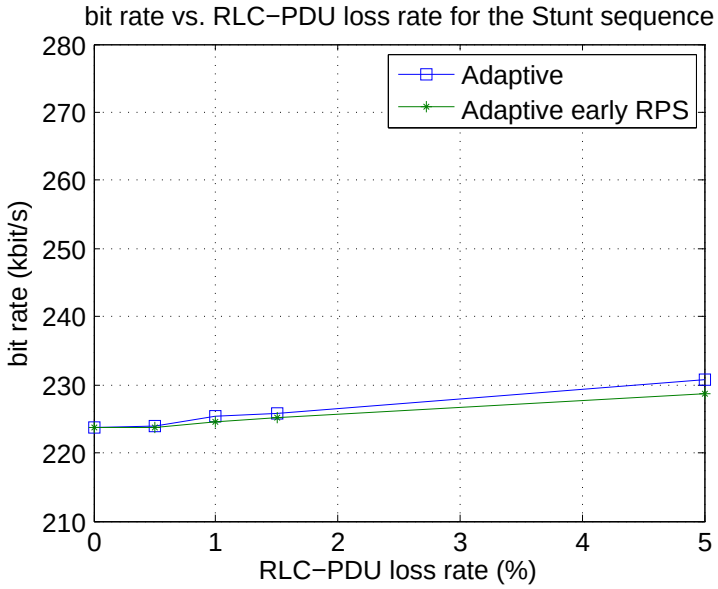


Figure 6.24: Bit rate vs. RLC-PDU loss rate for the *Stunt* sequence using LT code.

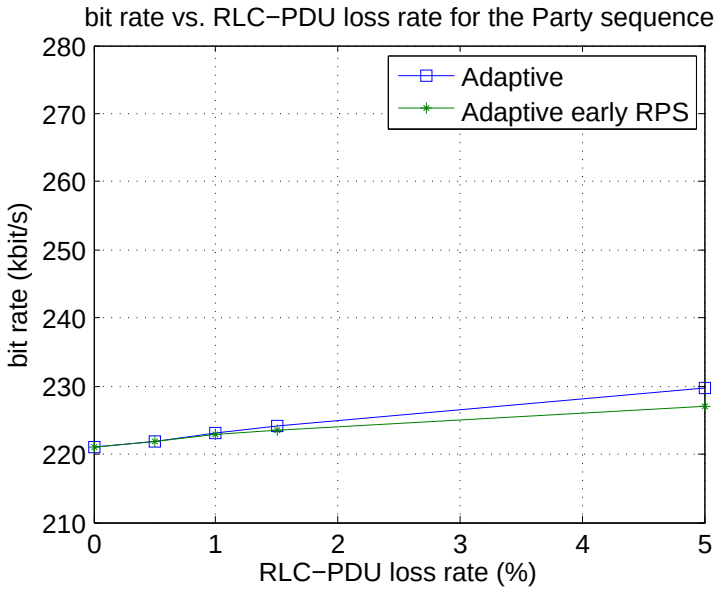


Figure 6.25: Bit rate vs. RLC-PDU loss rate for the *Party* sequence using LT code.

The bit rate curves for the *Stunt* and *Party* video sequences for the Raptor code are given in Figure 6.26 and Figure 6.27, respectively. The trend in the Raptor code results are similar to the LT code results.

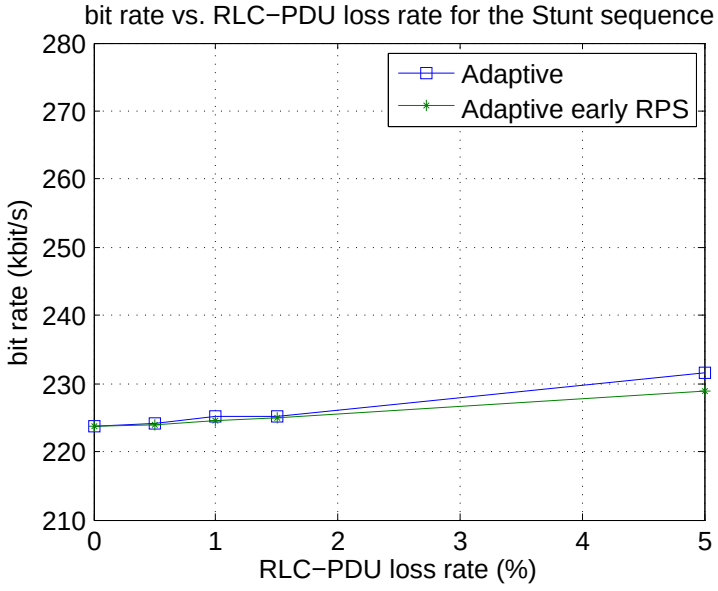


Figure 6.26: Bit rate vs. RLC-PDU loss rate for the *Stunt* sequence using Raptor code.

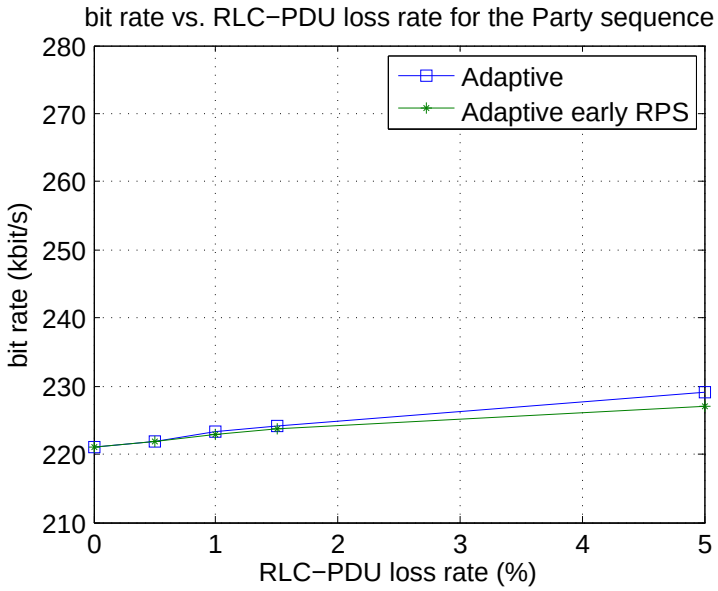


Figure 6.27: Bit rate vs. RLC-PDU loss rate for the *Party* sequence using Raptor code.

It is clear that the **Adaptive early RPS** technique achieves better PSNR and PDVD results at less bandwidth usage, compared to the **Fixed early RPS** scheme. Hence, **Adaptive early RPS** is the best scheme overall.

6.5 Timing diagram and end-to-end delay analysis

The timing diagram is given in Figure 6.28. It is assumed that the video coding sequence is in the form IPPPP....The first video frame is encoded as I frame and rest of the frames in the video sequence are encoded as P frames. The end-to-end delay components are the same as in Section 4.4. The source block consists of NAL units corresponding to four video frames. The frame buffering delay for the first source block is 3/FPS. The frame buffering delay for the subsequent source blocks is 4/FPS. FPS is the video frame rate in frames per second. The transmission sending deadline is 4/FPS. The end-to-end delay components and their values are given in Table 6.2.

The total end-to end-delay given in Table 6.2 is higher than 400 ms, which is the maximum end-to-end delay specified by the 3GPP for video telephony. The major component of the end-to-end delay is the LT decoding delay. The implementation of LT code used in the experiments was not optimized for speed. As already mentioned in Section 4.4, if an optimized implementation of LT code or Raptor code is used, then

the end-to-end delay can be below 400 ms.

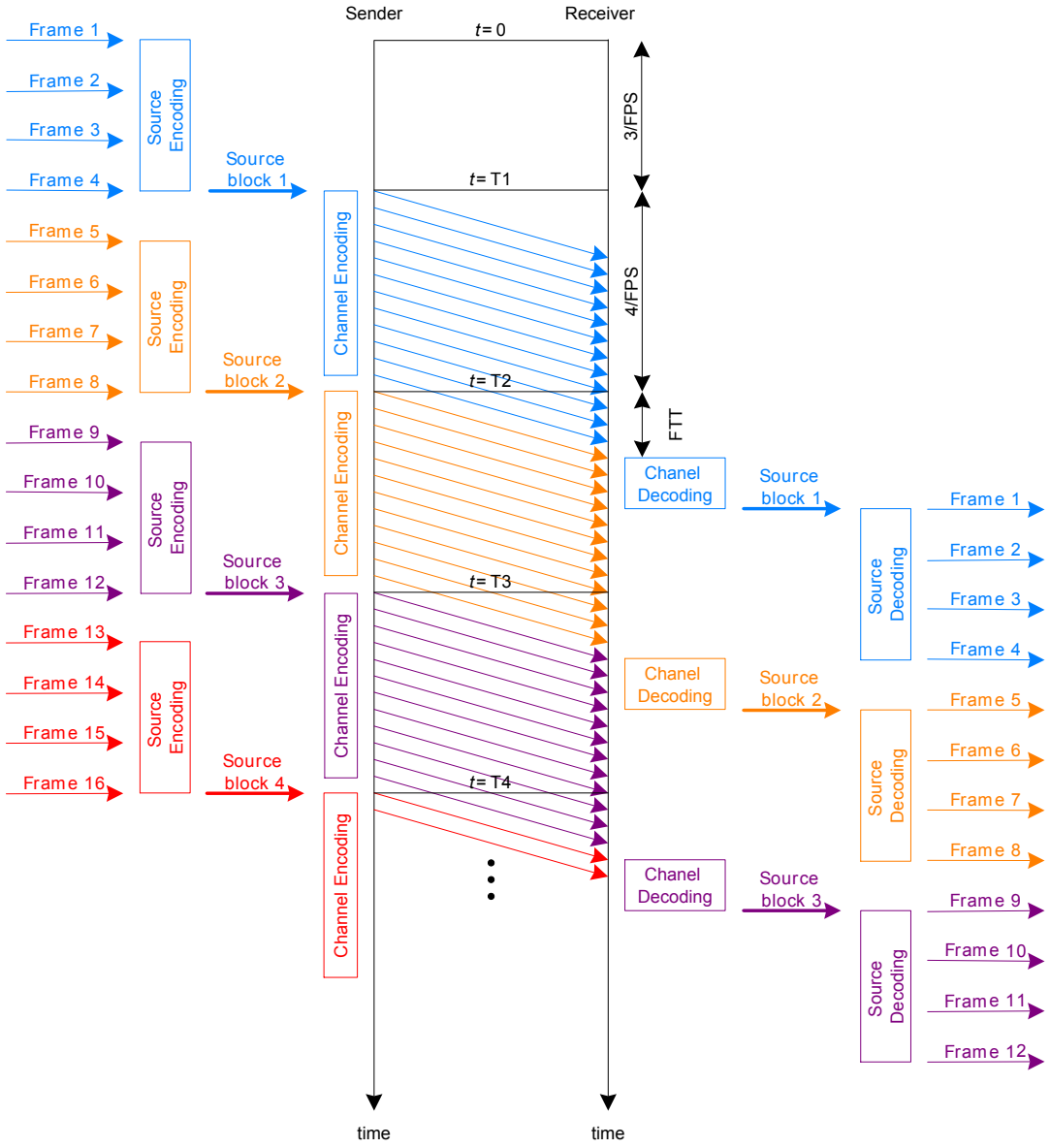


Figure 6.28: Timing diagram and end-to-end delay components. The frame buffering delay for the first source block is 3/FPS. The frame buffering delay for subsequent source blocks is 4/FPS. The transmission sending deadline (transmission delay) is 4/FPS. The total end-to-end delay is the sum of the end-to-end delay components.

Delay Component	*Stunt* Delay (ms)	*Party* Delay (ms)
Frame buffering	99.9	125
H.264 encoding	5.0	5.0
LT encoding	< 1	< 1
Transmission delay	133.3	166.66
Propagation delay	40	40
LT decoding	125	213
H.264 decoding	1.6	1.6
Total delay	405.8	552.26

Table 6.2: End-to-end delay components for *Stunt* and *Party* video sequences for Expanding Factor 8. The total end-to-end delay is the sum of the individual end-to-end delay components.

6.6 Summary

In this chapter, the main contribution is the combination of early reference picture selection technique with each of **Fixed** and **Adaptive** methods to improve the performance of the **Fixed** and the **Adaptive** methods. The combination of early reference picture selection with **Fixed** and **Adaptive** methods improves video quality. The early reference picture selection and early abort is achieved by including a packet deadline scheme and symbols lost threshold criteria. From the results, the best performing technique is the **Adaptive early RPS** technique.

Chapter 7

Exploiting channel history

7.1 Introduction

It has been shown in [38] that RLC-PDU losses in the Rayleigh fading wireless channel, under certain conditions, are highly correlated. This correlation can be exploited to improve the performance of the **Adaptive early RPS** method. As given in equation 2.7, the degree of correlation depends on the value of f_dT. Smaller values (< 0.1) of f_dT result in stronger correlation (burst errors). For LTE system under consideration, the value of f_dT is small enough to result in strong correlation.

7.2 Proposed method

The system block diagram is the same as in Figure 4.1. The transmission sending deadline and transmission receiving deadline are the same as in Section 6.4. The proposed scheme is a variation of the **Adaptive early RPS** scheme given in Section 6.4. As investigated in [38], RLC-PDU losses in the Rayleigh fading wireless channel, may be highly correlated. In the proposed method, the RLC-PDU loss correlation is exploited. The initial redundancy of the rateless code is dynamically adjusted, based on the average RLC-PDU (RTP/UDP/IP packet) loss rate in the previous few consecutive source blocks. This scheme is called **Adaptive past channel early RPS** scheme.

For the **Adaptive past channel early RPS** method, the total initial redundancy of the rateless code is adjusted dynamically as:

$$\hat{r} = r_f + l_r$$

where $\hat{r}$ is the total initial redundancy of the rateless code, r_f is the fixed initial redundancy of the rateless code, and l_r is the RLC-PDU loss rate in the last few consecutive source blocks.

The packetization scheme for this method is given in Section 7.3. The average loss rate for the initial round packets is determined using a deadline scheme as in Section 6.4. The average loss rate for both second round and non-rateless packets is determined using same deadline scheme. This is possible, since the transmitter sends a packet at every transmission opportunity. Thus, the past history information about the RLC-PDU (RTP/UDP/IP packet) losses in the Rayleigh fading wireless channel is used to improve the performance of the **Adaptive early RPS** method.

7.3 Proposed packetization

The packetization for the proposed method is given in Figure 7.1. Compared to the packetization scheme given in Figure 6.3.1, there are two NACK fields. These NACK fields are NACK1 and NACK2. NACK1 contains NACK for the initial round packets. NACK2 contains NACK for the second round and non-rateless packets. The size of each NACK field is 1 byte. The first four bits of each NACK field contains the NACK value of the current source block. The last 4 bits of each NACK field contains the NACK value for the previous source block. Each NACK value is retained for the duration of two source blocks. The total size of the rateless code information, reference picture selection information and NACK information fields in the packetization is 7 bytes.

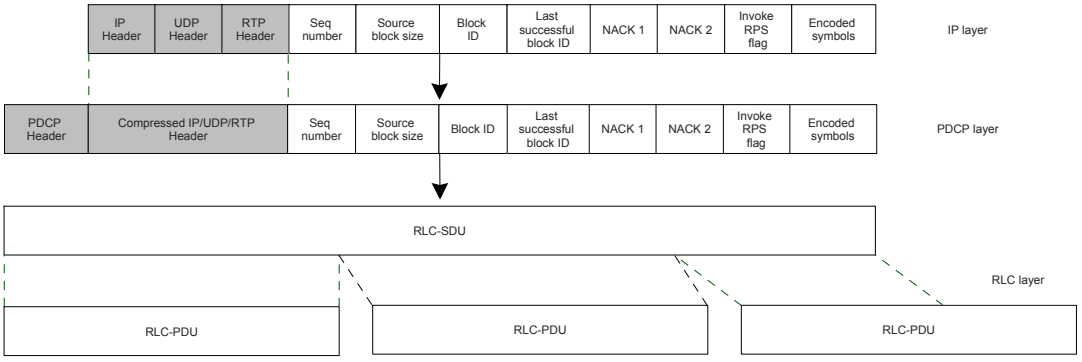


Figure 7.1: Proposed packetization of RTP/UDP/IP packet for the **Adaptive past channel early RPS** scheme.

7.4 Experimental results

The mobile video telephony scenario and network topology is the same as in Section 6.3.2. Video and transmission parameters are the same as in Section 6.4.2. The goal of the experiments is to show that using channel history improves video quality.

The Raptor code failure probability model as in [109] is used. r_f is fixed to 17%. l_r is the RLC-PDU loss rate in the last 6 consecutive source blocks. l_r value of 6 gives the best video quality, when the two schemes are compared at the same bit rate.

The initial redundancy of the **Adaptive early RPS** method is adjusted, so that the bit rate for the **Adaptive early RPS** method is the same as the **Adaptive past channel early RPS** method at a given RLC-PDU loss rate. This is done to compare the two methods at the same bit rate.

The following error resilience schemes are compared:

- **Adaptive early RPS:** The **Adaptive early RPS** scheme given in Section 6.4.
- **Adaptive past channel early RPS:** This is the proposed scheme. The initial rateless code redundancy is dynamically adjusted, based on the RLC-PDU losses in the previous few source blocks.

The PSNR vs. RLC-PDU loss rate curves for the *Stunt* sequence are given in Figure 7.2. At RLC-PDU loss rate of 1.5% or lower, the improvement in the **Adaptive past channel early RPS** scheme is not significant. Between RLC-PDU loss rates of 1.5% and 5%, there is some improvemnt in results. Between RLC-PDU loss rate of 5% and 10%, the improvement is significant. At RLC-PDU loss rate of 10% in each wireless link, the **Adaptive past channel early RPS** scheme achieves PSNR gain of 0.72 dB compared to the **Adaptive early RPS** scheme. This gain is achieved at the same bit rate for **Adaptive past channel early RPS** and **Adaptive early RPS** schemes. The bit rate curves are given in Figure 7.4.

The PDVD vs. RLC-PDU loss rate curves for the *Stunt* sequence are given in Figure 7.3. As the RLC-PDU loss rate increases, the **Adaptive past channel early RPS** scheme achieves better results than the **Adaptive early RPS** scheme. At RLC-PDU loss rate of 10% in each wireless link, the **Adaptive past channel early RPS** scheme achieves PDVD gain of 5.39% compared to the **Adaptive early RPS** scheme.

The PSNR vs. RLC-PDU loss rate curves for the *Party* sequence are given in Figure 7.5. At RLC-PDU loss rate of 1.5% or lower, the performance of the two schemes are similar. At RLC-PDU loss rate of 10% in each wireless link, the **Adaptive past channel early RPS** scheme achieves PSNR gain of 0.41 dB compared to the **Adaptive early RPS** scheme.

The PDVD vs. RLC-PDU loss rate curves for the *Party* sequence are given in Figure 7.6. At RLC-PDU loss rate of 1.5% or lower, the performance of the two schemes are similar. At RLC-PDU loss rate of 10% in each wireless link, the **Adaptive past channel early RPS** scheme achieves PDVD gain of 1.74% compared to the **Adaptive early RPS** scheme.

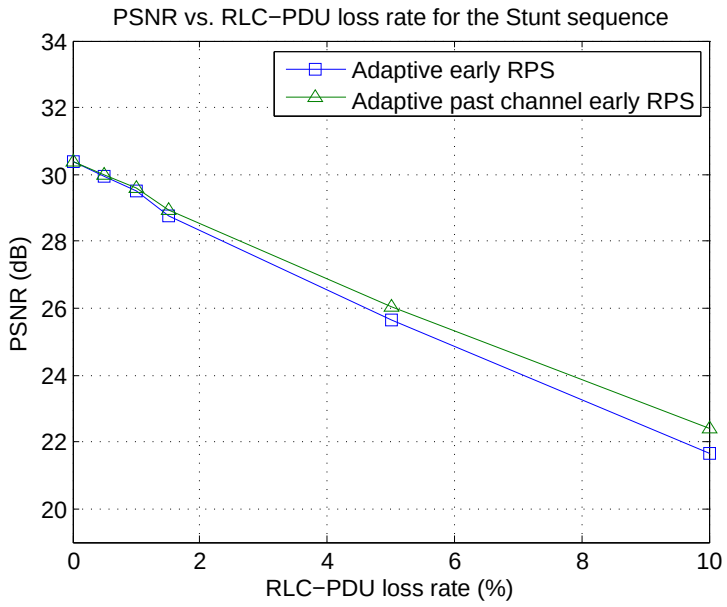


Figure 7.2: PSNR vs. RLC-PDU loss rate results for the *Stunt* sequence.

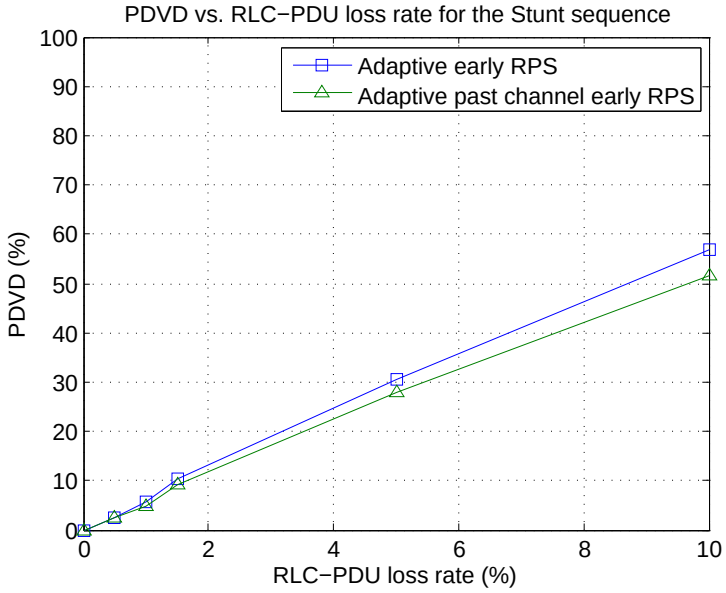


Figure 7.3: PDVD vs. RLC-PDU loss rate results for the *Stunt* sequence.

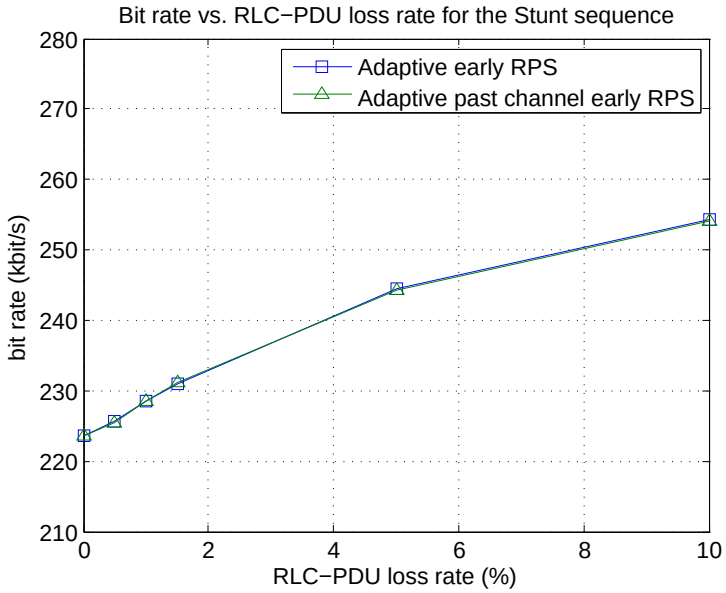


Figure 7.4: Bit rate vs. RLC-PDU loss rate results for the *Stunt* sequence.

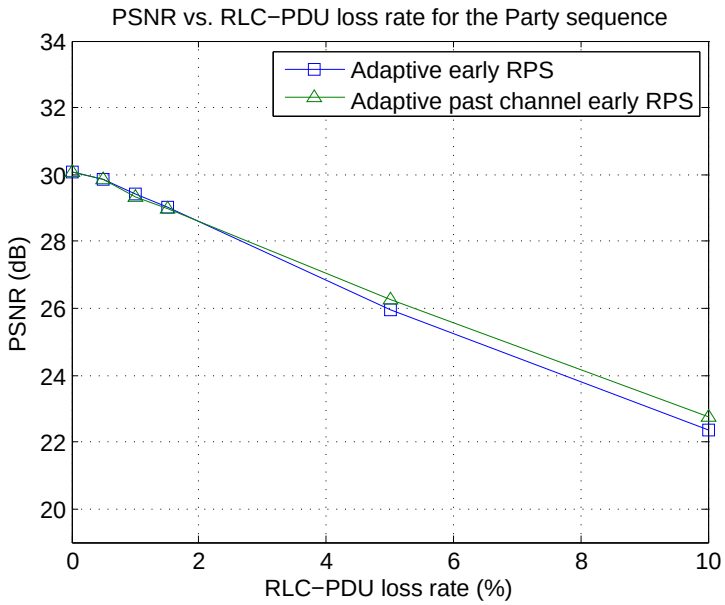


Figure 7.5: PSNR vs. RLC-PDU loss rate results for the *Party* sequence.

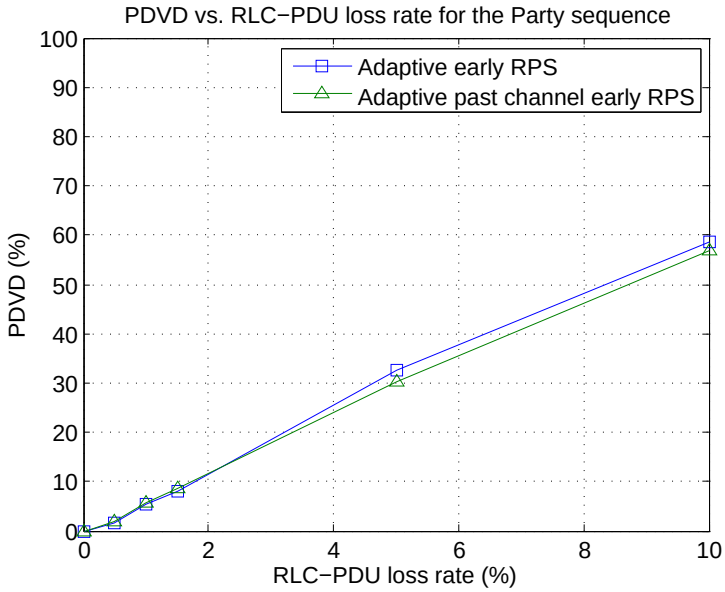


Figure 7.6: PDVD vs. RLC-PDU loss rate results for the *Party* sequence.

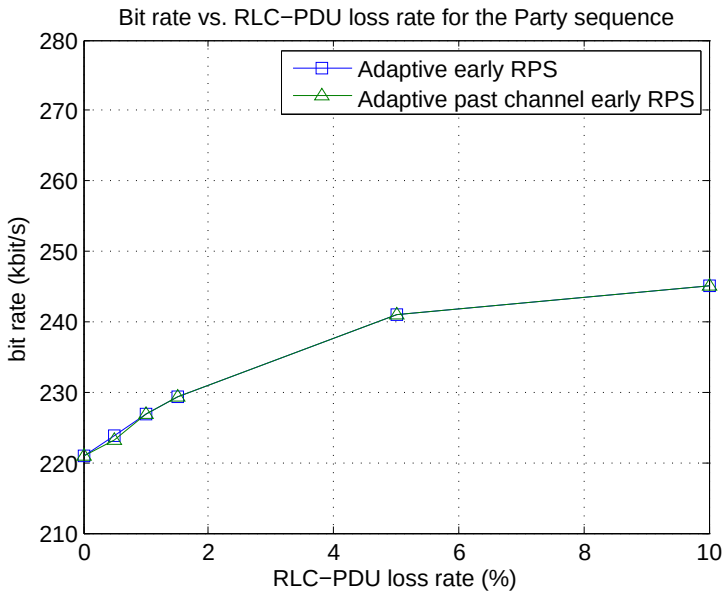
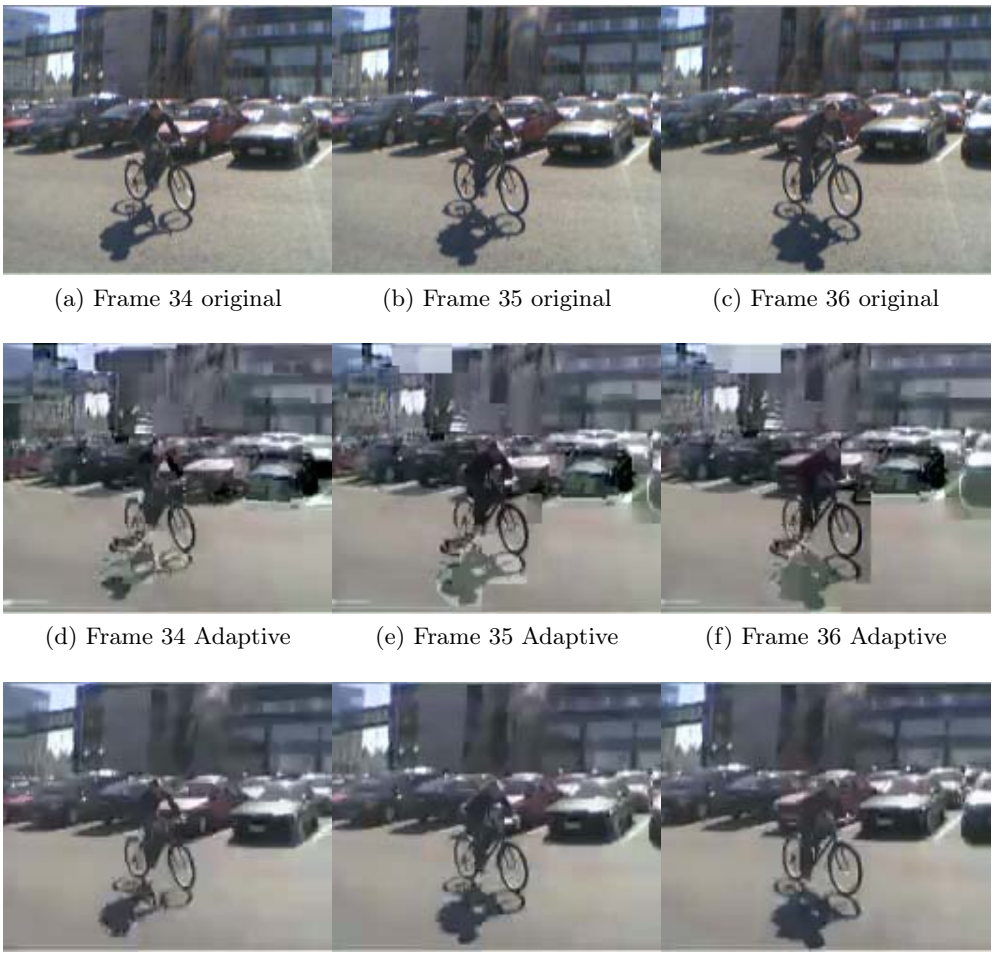


Figure 7.7: Bit rate vs. RLC-PDU loss rate results for the *Party* sequence.

7.5 Visual results

Visual results are given in Figure 7.8 and Figure 7.9. The RLC-PDU loss rate is 10% in each wireless link. The figures show that the **Adaptive past channel early RPS** method can achieve better visual results than the **Adaptive** method. The PSNR and PDVD results of the two schemes are close to each other but the visual results of the **Adaptive past channel early RPS** is much better than the **Adaptive** method. This is because for the **Adaptive** method, the spatio-temporal error has propagated to frames 34, 35 and 36 due to LT decoding failure before these frames. However, the LT decoding is successful for source block before these frames for the **Adaptive past channel early RPS method**.

The visual results are given for 10% RLC-PDU loss rate. At this loss rate, the **Adaptive past channel early RPS** method achieves significant objective results than the **Adaptive** method. There can be large differences between the improvement in the objective and visual results because visual results are given for the case in which the LT decoding is successful for **Adaptive past channel early RPS** method and failed for the **Adaptive** method.

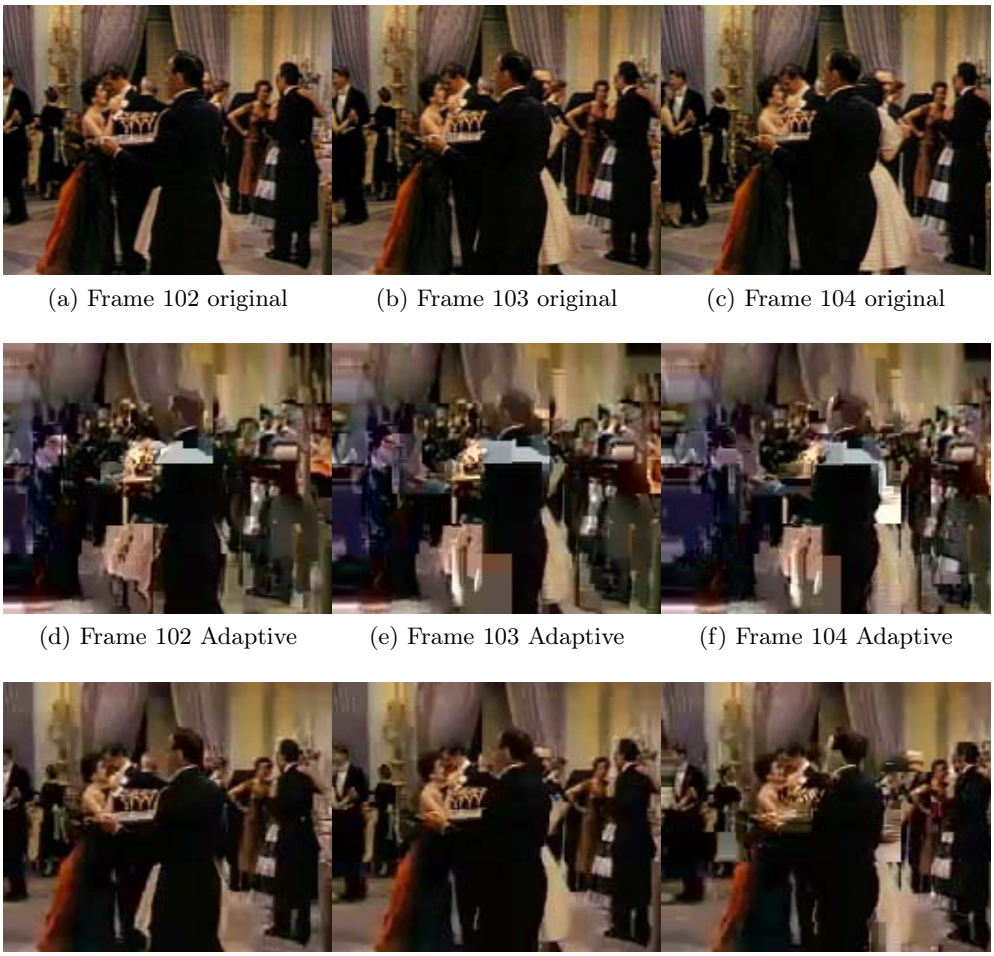


(a) Frame 34 original (b) Frame 35 original (c) Frame 36 original

(d) Frame 34 Adaptive (e) Frame 35 Adaptive (f) Frame 36 Adaptive

(g) Frame 34 Adaptive past channel early RPS (h) Frame 35 Adaptive past channel early RPS (i) Frame 36 Adaptive past channel early RPS

Figure 7.8: Visual results for the *Stunt* sequence



(a) Frame 102 original (b) Frame 103 original (c) Frame 104 original

(d) Frame 102 Adaptive (e) Frame 103 Adaptive (f) Frame 104 Adaptive

(g) Frame 102 Adaptive past channel early RPS (h) Frame 103 Adaptive past channel early RPS (i) Frame 104 Adaptive past channel early RPS

Figure 7.9: Visual results for the *Party* sequence

7.6 Timing diagram and end-to-end delay analysis

The timing diagram and end-to-end delay analysis is the same as in Section 6.5.

7.7 Summary

In this chapter, a new error resilience technique for packet switched mobile video telephony is proposed. The proposed error resilience technique is a combination of application layer forward error correction using rateless codes, automatic repeat request,

packet deadline based transmission strategy and information about the past history of RLC-PDU losses in the wireless channel. The proposed error resilience technique, called **Adaptive past channel early RPS** achieves better results than the **Adaptive early RPS** technique, at same bit rate. Also a RTP/UDP/IP level packetization scheme is proposed, to facilitate the implementation of the **Adaptive past channel early RPS** scheme in practical scenarios.

Chapter 8

Discussion, suggestion for further work and conclusion

In this dissertation, application layer error resilience techniques for packet switched video telephony over 3G networks have been proposed. The main idea is to combine Rateless codes, reference picture selection, cross layer optimization and automatic repeat request.

It has been shown in this dissertation that application layer Rateless codes can be used to achieve good video quality in packet switched mobile cellular video telephony.

The proposed error resilience techniques can be used in practical scenarios. The target 3GPP standards are HSPA and LTE. HPSA is currently the most widely used packet switched 3G standard. LTE is the latest version of 3GPP standard which supports only packet switched traffic. LTE has higher bit rate and lower latency than HSPA standard.

The first contribution is a combination of application layer Rateless codes, reference picture selection and cross layer optimization. A real software implementation of LT codes was used in this work. The proposed method achieved better PSNR and PDVD results than a state of the art error resilience technique known as IEC [13, 14] for *Stunt* and *Party* video sequences.

The second contribution is a channel adaptive error resilience method which combines Rateless codes, reference picture selection and automatic repeat request. Au-

tomatic repeat request retransmits lost packets containing LT encoded symbols. The number of retransmissions depends on the packet loss rate in the wireless channel. The channel adaptive method presented in Chapter 5 achieved better results than the fixed method when higher radio bearer bit rate was available. Better results were obtained for the channel adaptive method at lower bit rate than the fixed rate method.

The third contribution is an error resilience technique which combines Rateless codes, automatic repeat request and early reference picture selection. The reference picture selection was invoked early, if the transmitter was certain that the LT decoding would fail. This decision to invoke reference picture selection was based on the cumulative negative feedback about the lost packets. The improvement with early reference picture selection methods was obtained at video frame rates of 24 and 30 fps. The results were obtained under simulated and controlled conditions. The simulation conditions (RLC-PDU loss rates) were varied systematically. This was done so that the results could be related to the simulation conditions.

The fourth contribution is an error resilience technique which combines Rateless codes, automatic repeat request, early reference picture selection and exploitation of past history of the wireless channel. The initial redundancy of the Rateless code was dynamically adjusted according to the RLC-DPU loss rate in the last six consecutive source blocks. Improved results were obtained when exploiting past history of the wireless channel. The improvement was significant, when the packet loss rate in the two wireless links was high. The packet loss rate was determined using packet deadline schemes for the initial round as well as for the subsequent packets in that source block.

RTP/UDP/IP packetization schemes are also presented. These packetization schemes can be used to implement the proposed error resilience techniques in practical scenarios.

8.1 Limitations and future work

The proposed error resilience techniques were tested for two transmission bit rates only. These bit rates correspond to the ones usually used in video telephony over 3G networks. A broader evaluation should consider more bit rates.

Similarly, results were provided for two video sequences only (Stunt and Party). More video sequences should be used for a better evaluation of the proposed techniques.

These sequences should contain different levels of motion and visual content.

To evaluate video quality, PSNR, PDVD, and visual results were presented. Other objective video quality metrics such as VQM and SSIM could be used in the future. Also, subjective video quality metrics such as MOS under standard testing conditions would provide a better visual assessment.

The simulation scenario and setting were simple and could be expanded in the future. For example,

- Cross traffic and core network congestion were not considered;
- The maximum forward trip time and maximum backward trip time for each packet were assumed to be identical;
- wired link was assumed to be error and congestion free;
- The background traffic was subject to the same RLC-PDU loss rate as the forward traffic.

The packetization scheme was designed such that one RLC-PDU contains exactly one RTP/UDP/IP packet. The goal is to minimize packet loss at the transport layer when data is lost at the RLC layer. However, there is no guarantee that this strategy minimizes the distortion for a given transmission bit rate. More research is needed to find the optimal tradeoff between RTP/UDP/IP packet size and rate-distortion performance under various bearer parameters (TTI, RLC-PDU size) and video bit rates.

The results were provided for non-systematic Rateless codes. Better results are expected by using a systematic Raptor code.

A major contributor to the end-to-end delay was the Rateless code decoding delay. This delay can be reduced by using a more efficient implementation of the decoder. For example, Rateless coding and decoding require only exclusive-OR operation. Thus, they can easily be implemented in hardware. More time can be saved by generating the random numbers needed in the encoding and decoding offline. This would further reduce the encoding and decoding time and decrease the end-to-end delay below 400 ms.

References

[1] 3GPP, 3GPP Technical Specification 22.105 (v 8.4.0), *Service aspects; Services and service capabilities*, 2007.

[2] Stockhammer, T., Hannuksela, M.M. and Wiegand, T., H.264/AVC in Wireless Environments. *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 657-673, 2003.

[3] Girod, B. and Farber, N., Feedback-Based Error Control for Mobile Video Transmission, *Proc. IEEE, Special Issue on Video for Mobile Multi-media*, vol. 97, no. 10, pp. 1707-1723, 1999.

[4] Khan, F., *LTE for 4G Mobile Broadband - Air Interface Technologies and Performance*, Cambridge University Press, New York, NY, 2009.

[5] Lescuyer, P. and Lucidarme, T., *Evolved Packet System (EPS): The LTE and SAE Evolution of 3G UMTS*, John Wiley & Sons, Ltd. 2008.

[6] 3GPP, 3GPP Technical Specification 23.107 (v 8.0.0), *Quality of Service (QoS) concept and architecture*, 2008.

[7] Postel, J., *User Datagram Protocol*, Request for Comments 768, Internet Engineering Task Force, 1980.

[8] Wang, Y., Wenger, S., Wen, J. and Katsaggelos, A. K., Review of Error Resilient Coding Techniques for Real-Time Video Communications, *IEEE Signal Processing Magazine*, vol. 17, no. 4, pp. 61-82, 2000.

[9] Luby, M., LT-Codes, *Proc. 43rd Annual IEEE Symposium on Foundations of Computer Science*, Vancouver, BC, Canada, pp. 271280, 2002.

[10] Shokrollahi, A., Raptor codes, *IEEE Transactions on Information Theory*, vol. 52. no. 6. pp. 2551-2567, 2006.

[11] Dawood, M., Hamzaoui, R., Ahmad, S. and Al-Akaidi, M., Error-resilient packet switched H.264 mobile video telephony with LT coding and reference picture selection, *Proc. 17th European Signal Processing Conference, EUSIPCO 09*, Glasgow, UK, pp. 2211-2215, Aug. 2009.

[12] Dawood, M., Hamzaoui, R., Ahmad, S. and Al-Akaidi, M., Error resilient packet-switched video telephony with adaptive rateless coding and reference picture selection, *Proc. MESM 2010, Middle Eastern Multiconference on Simulation and Modelling*, Alexandria, Egypt, Dec. 2010.

[13] Zia, W., Diepold, K. and Stockhammer, T., Complexity Constrained Robust Video Transmission for Hand-Held Devices, *Proc. IEEE International Conference on Image Processing 2007 (ICIP 2007)*, San Antonio, TX, USA, vol. 4, pp. 261-264, 2007.

[14] Zia, W., et al., Interactive Error Control for Mobile Video Telephony, *Proc. IEEE International Conference on Communications 2007 (ICC 2007)*, Glasgow, Scotland, pp. 1797-1802, 2007.

[15] Holma, H. and Toskala, A. (Eds.), *WCDMA for UMTS - HSPA Evolution and LTE, 4th ed.*, John Wiley & Sons, Ltd. 2007.

[16] Chakraborty, S., Peisa, J., Frankkila, T. and Synnergren, P., *IMS Multimedia Telephony over Cellular Systems: VoIP Evolution in a Converged Telecommunication World*, John Wiley & Sons, Ltd. 2007.

[17] Holma, H. and Toskala, A. (Eds.), *HSDPA/HSUPA for UMTS High Speed Radio Access for Mobile Communications*, John Wiley & Sons, Ltd. 2006.

[18] Hannu, H., et al, *RObust header compression (ROHC): Framework and four profiles: RTP, UDP, ESP, and uncompressed*, Request for Comments 3095, Internet Engineering Task Force, 2001.

[19] 3GPP, 3GPP Technical Specification 25.323 (v 8.3.0), *Packet Data Convergence Protocol (PDCP) specification*, 2008.

[20] Etoh, M. and Yoshimura, T., Advances in Wireless Video Delivery, in *Proc. IEEE*, vol. 93, no. 1, pp. 111-122, 2005.

[21] 3GPP, 3GPP Technical Specification 26.236 (v 7.3.0), *Packet switched conversational multimedia applications; Transport protocols*, 2008.

[22] 3GPP, 3GPP Technical Specification 26.235 (v 8.0.0), *Packet switched conversational multimedia applications; Default codecs*, 2008.

[23] Schulzrinne, H., Casner, S., Frederick, R. and Jacobson, V., *RTP: A Transport Protocol for Real-Time Applications*. Request for Comments 3550, Internet Engineering Task Force, 2003.

[24] Wiegand, T., Sullivan, G.J., Bjontegaard, G. and Luthra, A., Overview of the H.264/AVC video coding standard, *IEEE Transactions on Circuits and Systems for Video Technology*, 13, pp. 560-576, 2003.

[25] Richardson, I.E.G., *H.264 and MPEG-4 Video Compression: Video Coding for Next-generation Multimedia*, John Wiley & Sons, Ltd. 2003.

[26] JVT (Joint Video Team) of ISO/IEC and ITU-T VCEG, *Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC)*, JVT-G050r1, May 2003.

[27] Steinbach, E., Farber, N. and Girod, B., Standard compatible extension of H.263 for robust video transmission in mobile environments, *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, no. 6, pp. 872-881, 1997.

[28] Byers, J.W., Luby, M. and Mitzenmacher, M., A Digital Fountain Approach to Asynchronous Reliable Multicast, *IEEE Journal on Selected Areas in Communications*, vol. 20. no. 8. pp. 1528-1540, 2002.

[29] Hamzaoui, R., Ahmad, S., Al-Akaidi, M., Video Streaming: Reliable Wireless Streaming with Digital Fountain Codes, *Encyclopedia of Wireless and Mobile Communications*, Taylor & Francis, 2007.

[30] Ahmad, S., Hamzaoui, R. and Al-Akaidi, M., Unequal error protection using LT codes and block duplication, *Proc. Middle Eastern Multiconference on Simulation and Modelling*, Amman, 2008.

[31] *Ns-2*, http://www.isi.edu/nsnam/ns/

[32] *Eurane Ns-2 extensions*, http://eurane.ti-wmc.nl/eurane/

[33] *NSMIRACLE*, http://www.dei.unipd.it/wdyn/?IDsezione=3966

[34] *Seawind*, http://www.cs.helsinki.fi/research/iwtcp/seawind/

[35] *NCTUNS*, http://nsl.csie.nctu.edu.tw/nctuns.html.

[36] *Opnet*, http://www.opnet.com

[37] 3GPP, *Permanent document on test components*, 3GPP Permanent Document S4-060515, 2006.

[38] Zorzi, M., Rao, R.R. and Milstein, L.B., On the accuracy of a first order markov model for data transmission on fading channels, *Proc. Fourth IEEE International Conference on Universal Personal Communications*, Tokyo, Japan, pp. 211-215, 1995.

[39] Wang, H.S. and Chang P.-C., On verifying the first-order Markovian assumption for a Rayleigh fading channel model, *IEEE Transactions on Vehicular Technology*, vol. 45, no. 2, pp. 353-357, May 1996.

[40] Kang, K. and Shin, H., Reduced Data Rates for Energy-Efficient Reed-Solomon FEC on Fading Channels, *IEEE Transactions on Vehicular Technology*, vol. 58, no. 1, pp. 176-187, 2009.

[41] Zhang, Q., Zhu, W. and Zhang, Y.-Q., Channel-Adaptive Resource Allocation for Scalable Video Transmission Over 3G Wireless Network, *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 8, pp. 1049-1063, 2004.

[42] Marcum, J. I., *Table of Q Functions*, U.S. Air Force Project RAND Res. Memo. M-339, ASTIA Document AD 1165451, Rand Corp., Santa Monica, CA, Jan. 1950.

[43] Zorzi, M., Rao, R.R. and Milstein, L.B., Error statistics in data transmission over fading channels, *IEEE Transactions on Communications*, vol. 46, no. 11, pp. 1468-1477, 1998.

[44] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, Image quality assessment: From error visibility to structural similarity, *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600-612, Apr. 2004.

[45] Z. Wang, L. Lu, A. C. Bovic, Video quality assessment using structural distortion measurement, *Signal Processing: Image Communication, special issue on Objective video quality metrics*, vol. 19, no. 2, pp. 121-132, February 2004.

[46] D. Niranjan et al., Image Quality Assessment Based on a Degradation Model, *IEEE Transaction on Image Processing*, vol. 9, no. 4, April 2000.

[47] C. J. Branden Lambrecht and O. Verscheure, Perceptual Quality Measure using a Spatio-Temporal Model of the Human Visual System, *Proc. SPIE*, vol. 2668, p. 450-461, March, 1996.

[48] M.H. Pinson, S.Wolf, A New Standardized Method for Objectively Measuring Video Quality, IEEE Transactions on Broadcasting, vol. 50, pp. 312-322, Sep. 2004.

[49] VQEG, Final Report From the Video Quality Experts Group on the Validation of Objective Models of Video Quality Assessment, Mar. 2000, Available: http://www.vqeg.org/.

[50] http://www.mathworks.com/help/toolbox/vipblks/ref/psnr.html

[51] 3GPP, 3GPP Technical Recommendation 26.902 (v 7.1.2), *Video codec performance*, 2008.

[52] Wang, Y. and Zhu, Q.-F., Error control and concealment for video communication: a review, *Proc. IEEE*, vol. 86, no. 5, pp. 974-997, 1998.

[53] Devadoss, J., et al., Evaluation of Error Resilience Mechanisms for 3G Conversational Video, *Proc. Tenth IEEE International Symposium on Multimedia*, Berkeley, CA, USA, pp. 378-383, 2008.

[54] Karande, S. and Radha, H., Rate-constrained adaptive FEC for video over erasure channels with memory, *Proc. International Conference on Image Processing, 2004. ICIP '04*, Singapore, vol. 4, pp. 2539, 2005.

[55] Xunqi, Y., Modestino, J.W. and Bajic, I.V., Performance analysis of the efficacy of packet-level FEC in improving video transport over networks, *Proc. International Conference on Image Processing, 2005. ICIP '05*, Genoa, Italy, vol. 2, pp. II, 2005.

[56] Liu, X., Li, Z.-C., Chen, Z.-F. and Wang, L., Perceptual quality optimized FEC for video streaming, *Proc. 2010 International Conference on Audio Language and Image Processing (ICALIP)*, Shanghai, China, pp. 98, Nov. 2010.

[57] Tan, Y., Wang, H., Wang, X. and Zhang, Q., A video transmission algorithm over the Internet based on FEC and Kalman, *IEEE International Symposium on IT in Medicine and Education, 2008. ITME 2008*, Xia'men, China, pp. 263, Dec. 2008.

[58] Neckebroek, J., Bruneel, H., Moeneclaey, M., Application layer ARQ for protecting video packets over an indoor MIMO-OFDM link with correlated block fading, *IEEE Journal on Selected Areas in Communications*, vol. 28, pp. 467, April, 2010.

[59] Zhou, J., Wu, T. and Li, Z., Cross-Layer QoS for Wireless Video Based on Priority-ARQ, *Proc. 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM) 2010*, Shenzhen, China, pp. 1, 2010.

[60] Bucciol, P., Davini, G., Masala, E., Filippi, E. and De Martin, J.C., Cross-layer perceptual ARQ for H.264 video streaming over 802.11 wireless networks, *Proc. IEEE Global Telecommunications Conference, 2004. GLOBECOM '04*, Texas, USA, vol. 5, pp. 3027, Dec. 2004.

[61] Srisawaivilai, N. and Aramvith, S., Improved H.264 rate-control using channel throughput estimate for ARQ-based wireless video transmissions, *Proc. International Symposium on Intelligent Signal Processing and Communication Systems, 2005. ISPACS 2005*, pp. 13-16, Dec. 2005.

[62] Bucciol, P., Masala, E. and De Martin, J.C., Perceptual ARQ for H.264 video streaming over 3G wireless networks, *Proc. 2004 IEEE International Conference on Communications*, Paris, vol. 3, pp. 1288, June, 2004.

[63] Wang C.-H., Chang, R.-I., Ho, J. -M. and Hsu, S.-C., Rate-sensitive ARQ for real-time video streaming, *Proc. IEEE Global Telecommunications Conference, 2003. GLOBECOM '03*, vol. 6, pp. 3361, Dec. 2003.

[64] Ding, X. and Roy, K., A novel bitstream level joint channel error concealment scheme for realtime video over wireless networks, *Proc. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies INFOCOM 2004*, vol. 3, pp. 2163, 2004.

[65] Zhou, J., Yan, B. and Gharavi, H., Efficient Motion Vector Interpolation for Error Concealment of H.264/AVC, *IEEE Transactions on Broadcasting*, vol. 57, no. 1, pp. 75, March 2011.

[66] Zhai, G., Yang, X., Lin, W. and Zhang, W., Simultaneous deblocking and error concealment for decoded visual signal, *Proc. 2010 IEEE International Symposium on Circuits and Systems (ISCAS)*, Paris, pp. 2626, May 2010.

[67] Li, H. and Zhong, Y., A Hermite Interpolation Based Motion Vector Recovery Algorithm for H.264/AVC, *Proc. Second International Conference on Communication Software and Networks, 2010. ICCSN '10*, Singapore, pp. 63, Feb. 2010.

[68] Wu, G. -L., Chen, C. -Y., Wu, T. -H. and Chien, S. -Y., Efficient Spatial-Temporal Error Concealment Algorithm and Hardware Architecture Design for H.264/AVC,

IEEE Transactions on Circuits and Systems for Video Technology, vol. 20, no. 11, pp. 1409, Nov. 2010.

[69] Cote, G. and Kossentini, F., Optimal intra coding of blocks for robust video communication over the Internet, *Signal Processing: Image Communication, Special Issue on Real-time Video*, vol. 15, pp. 25-34, 1999.

[70] Sullivan, G. J. and Wiegand, T., Rate-Distortion Optimization for Video Compression, *IEEE Signal Processing Magazine*, vol. 15, no. 6, pp. 74-90, 1998.

[71] Zhang, Y., Gao, W., Sun, H., Huang, Q. and Lu, Y., Error Resilient Video Coding in H.264 Encoder with Potential Distortion Tracking, *Proc. IEEE International Conference on Image Processing, ICIP 2004*, Singapore, vol. 1, pp. 163-166, October 2004.

[72] Zhang, R., Regunathan, S.L. and Rose, K., Video coding with optimal inter/intra-mode switching for packet loss resilience, *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 6, pp. 966-976, 2000.

[73] Wang, Y.-K., Hannuksela, M.M. and Gabbouj, M., Error-Robust Inter/Intra Macroblock Mode Selection Using Isolated Regions, *Proc. 13th Packet Video Workshop*, Nantes, 2003.

[74] Chen, Q., Chen, Z., Gu, Z. and Wang, C., Attention Based Adaptive Intra Refresh For Error Resilient Video Coding, *Proc. 25th Picture Coding Symposium 2006*, Beijing, China, April 2006.

[75] Wang, X., Kodikara, C., Sadka, A.H. and Kondoz, A.M., Robust GOB Intra Refresh Scheme for H.264/AVC Video Over UMTS, *Proc. 6th IEE International Conference on 3G and Beyond*, London, UK, pp. 1-4, November 2005.

[76] Liang, Y.J., El-Maleh, K. and Manjunath, S., Upfront intra-refresh decision for low-complexity wireless video telephony, *Proc. 2006 IEEE International Symposium on Circuits and Systems (ISCAS 2006)*, Island of Kos, Greece, pp. 4, 2006.

[77] Haskell, P. and Messerschmitt, D., Resynchronization of Motion Compensated Video Affected by ATM Cell Loss, *Proc. 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 92*, San Francisco, CA, USA, vol. 3, pp. 545-548, 1992.

[78] Yoon, D.H., et al., Spiral intra macroblock refresh with motion vector restriction for low bit-rate video telephony over a 3G network, *IEEE Transactions on Consumer Electronics*, vol. 50, no. 4, pp. 1038-1043, 2004.

[79] Liu, L., Zhang, S., Ye, X. and Zhang, Y., Error Resilience Schemes of H.264/AVC for 3G Conversational Video Services, *Proc. The Fifth International Conference on Computer and Information Technology*, Shanghai, China, pp. 657, 2005.

[80] Yu, H.-B., Wang, C. and Yu, S., A novel error recovery scheme for H.264 video and its application in conversational services, *IEEE Transactions on Consumer Electronics*, vol. 50, no. 1, pp. 329-334, 2004.

[81] Wiegand, T., Farber, N., Stuhlmuller, K. and Girod, B., Error-resilient video transmission using long-term memory motion-compensated prediction, *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 6, pp. 1050-1062, 2000.

[82] Tu, W. and Steinbach, E., Proxy-based reference picture selection for real-time video transmission over mobile networks, *Proc. 2005 IEEE International Conference on Multimedia and Expo (ICME 2005)*, Amsterdam, The Netherlands, pp. 309-312, July 2005.

[83] Rhee, I. and Joshi, S., Error Recovery for Interactive Video Transmission over the Internet, *IEEE Journal on Selected Areas in Communications, Special Issue on Error Robust Transmission of Images and Video*, vol. 18. no. 6. pp. 1033-1049, 2000.

[84] Vadapalli, S.C., Shetiya, H. and Sethuraman, S., Efficient Alternative to Intra Refresh using Reliable Reference Frames, *Proc. IEEE International Conference on Multimedia and Expo 2007*, Beijing, China, pp. 124-127, August 2007.

[85] Zheng, J. and Chau, L.P., Error-resilient coding of H.264 based on periodic macroblock, *IEEE Transactions on Broadcasting*, vol. 52, no. 2, pp. 223-229, 2006.

[86] Chang, P.C. and Lee, T.H., Precise and fast error tracking for error-resilient transmission of H.263 video, *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, no. 4, pp. 600-607, 2000.

[87] Tu, W. and Steinbach, E., Proxy-based error tracking for H.264 based real-time video transmission in mobile environments, *Proc. 2004 IEEE International Conference on Multimedia and Expo (ICME 2004)*, Taipei, Taiwan, vol. 2, pp. 1367-1370, 2004.

[88] Choi, B.S., Chae, S.I. and Ra, J.B., A feedback channel based error compensation method for mobile video communications using H.263 standard, *Proc. 1999 International Conference on Image Processing*, Kobe, Japan, vol. 2, pp. 555-559, October 1999.

[89] Wada, M., Selective recovery of video packet loss using error concealment, *IEEE Journal on Selected Areas in Communications*, vol. 7, no. 5, pp. 807-814, 1989.

[90] Baccichet, P., Rane, S., Chimienti, A. and Girod, B., Robust Low-Delay Video Transmission using H.264/AVC Redundant Slices and Flexible Macroblock Ordering, *Proc. IEEE International Conference on Image Processing 2007 (ICIP 2007)*, San Antonio, TX, USA, vol. 4, pp. IV-93-IV-96, November 2007.

[91] Katz, B., et al., New Error-Resilient Scheme Based on FMO and Dynamic Redundant Slices Allocation for Wireless Video Transmission, *IEEE Transactions on Broadcasting*, vol. 53, no. 1, pp. 308-319, March 2007.

[92] Qu, Q., Pei, Y. and Modestino, J.W., Robust H.264 video coding and transmission over bursty packet-loss wireless networks, *Proc. IEEE Vehicular Technology Conference 2003*, Orlando, Florida, USA, vol. 5, pp. 3395-3399, October 2003.

[93] Chung-How, J.T.H. and Bull, D.R., Robust H.263+ video for real-time Internet applications, *Proc. International Conference on Image Processing 2000 (ICIP 2000)*, Vancouver, BC, Canada, vol. 3, pp. 544-547, September 2000.

[94] Zhai, F., et al., Rate-Distortion Optimized Hybrid Error Control for Real-Time Packetized Video Transmission, *Proc. IEEE International Conference on Communications 2004*, Paris, France, vol. 3, pp. 1318-1322, June 2004.

[95] Zhu, C., et al., Error resilient video coding using redundant pictures, *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 1, pp. 3-14, Jan. 2009.

[96] Wang, T.-C., Fang, H.-C. and Chen, L.-G., Low-delay and error-robust wireless video transmission for video communications, *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 12, pp. 1049-1058, 2002.

[97] Chen, M. and Wei, G., Multi-Stages Hybrid ARQ with Conditional Frame Skipping and Reference Frame Selecting Scheme for Real-Time Video Transport Over Wireless LAN, *IEEE Transactions on Consumer Electronics*, vol. 50, no. 1, pp. 158-167, Feb. 2004.

[98] Zhang, X-. W. and Hu. G., Strategies of improving QOS for Video Transmission over 3G Wireless Network, *Proc. 2nd International Conference on Applications and Systems of Mobile Technology*, Guangzhou, China, pp. 7, Nov. 2005.

[99] Lin, S., Costello Jr., D.J. and Miller, M.J., Automatic-repeat-request error-control schemes, *IEEE Communications Magazine*, vol. 22, no. 12, pp. 517, Dec. 1984.

[100] Aramvith, S., Lin, C.-W., Roy, S. and Sun, M.-T., Wireless video transport using conditional retransmission and low-delay interleaving, *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 6, pp. 558-565, 2002.

[101] Wen, J., Dai, Q. and Jin, Y., Channel-adaptive hybrid ARQ/FEC for robust video transmission over 3G, *Proc. IEEE International Conference on Multimedia and Expo 2005*, Amsterdam, The Netherlands, pp. 4, July 2005.

[102] S. Khan, Y. Peng, E. Steinbach, M. Sgroi, W. Kellerer, Application-driven cross-layer optimization for video streaming over wireless networks, *IEEE Communications Magazine*, vol. 44, no. 1, pp. 122-130, 2006.

[103] T. Stockhammer, Robust System and Cross-Layer Design for H.264/AVC-Based Wireless Video Applications, *EURASIP Journal on Applied Signal Processing*, vol 2006, pp 1-15, 2006.

[104] M. van der Schaar, D. S. Turaga, Cross-Layer Packetization and Retransmission Strategies for Delay-Sensitive Wireless Multimedia Transmission, *IEEE Transactions on Multimedia*, vol. 9, no. 1, pp. 185-197, 2007.

[105] Mills, D.L., *Network Time Protocol (Version 3) Specification, Implementation and Analysis*, Request for Comments 1305, Internet Engineering Task Force, 1992.

[106] Cataldi, P., Shatarski, M.P., Grangetto, M. and Magli, E., Implementation and Performance Evaluation of LT and Raptor Codes for Multimedia Applications, *Proc. International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, Pasadena, California, USA, 2006.

[107] Holma, H. and Toskala, A. (Eds.), *LTE for UMTS - OFDMA and SC-FDMA Based Radio Access*, John Wiley & Sons, Ltd. 2009.

[108] Dahlman, E., Parkvall, S., Sköld, J. and Beming, P., *3G Evolution: HSPA and LTE for Mobile Broadband*, Academic Press, 2007.

[109] Luby, M., Gasiba, T., Stockhammer, T. and Watson, M., Reliable Multimedia Download Delivery in Cellular Broadcast Networks, *IEEE Transactions on Broadcasting*, vol. 53. no. 1. pp. 235-246, 2007.