UNIVERSITY OF SOUTHAMPTON

FACULTY OF ENGINEERING, SCIENCE AND MATHEMATICS

SCHOOL OF ELECTRONICS AND COMPUTER SCIENCE

Southampton SO17 1BJ

United Kingdom

Low-Bit-Rate Joint Source-Channel Decoding Aided Wireless Video Communications

by

Nasru Minallah

B.Eng., M.Sc.

A Doctoral thesis submitted in partial fulfilment of the requirements for the award of Doctor of Philosophy at the University of Southampton

June 2010

SUPERVISOR:

Professor Lajos Hanzo

M.Sc., Ph.D, FREng, DSc, FIEEE, FIET

Chair of Telecommunications

School of Electronics and Computer Science

Dedicated to

my beloved parents

for their tremendous patience and care

with all my love and respect · · ·

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING, SCIENCE AND MATHEMATICS
SCHOOL OF ELECTRONICS AND COMPUTER SCIENCE

Doctor of Philosophy

Low-Bit-Rate Joint Source-Channel Decoding Aided Wireless Video Communications

by

Nasru Minallah

Detailed wireless video structures employing novel channel coding schemes for enhancing the achievable performance are designed. Although there is a plethora of papers on both robust video transmission, iterative detection and video telephony, there is a paucity of up-to-date research studies on the unified treatment of the topic of near capcity multimedia communication systems using iterative detection aided joint source-channel decoding employing sophisticated transmission techniques. Therefore in this thesis we focus our attention not only on the source and channel coding but also on their iterative decoding and transmission. Initially, we investigated the H.264 codec's error sensitivity. The perceptually more important bits were provided with more strong protection relative to less important bits using Unequal Error Protection (UEP) by applying different-rate Recursive Systematic Convolutional (RSC) codes.

We then further improved the attainable performance of a Data-Partitioned (DP) H.264 coded video transmission system using UEP based IrRegular Convolutional Codes (IRCC). An iterative detection aided combination of IRCC and a rate-1 precoder was used to improve the overall BER performance and to enhance the objective video quality expressed in terms of the Peak Signal-to-Noise Ratio (PSNR)[1]. More specifically, we exploited the the innate UEP capability and high design flexibility of IRCCs, which are constituted by different-rate subcodes capable of maintaining an excellent iterative decoding performance. In contrast to regular convolutional codes, which encode the entire segment of the source signal using the same code, the IRCCs introduced encode the source signal by splitting it into segments having specifically designed lengths, each of which is encoded by a code having an appropriately designed code-rate. A novel Extrinsic Information Transfer (EXIT) chart matching procedure was used for the design of our specific IRCC which allowed us to design near-capacity schemes.

Additionally, we developed a novel Unequal Source-Symbol Probability Aided (USSPA) design, which is capable of further enhancing the subjective video quality by exploiting the residual redundancy that remains in the source-coded stream after encoding. Furthermore, we proposed a family of Short Block Codes (SBCs) designed for guaranteed convergence in Iterative Source-Channel Decoding (ISCD). The DP H.264 source coded video stream was used to evaluate the performance of our system using SBCs in conjunction with RSCs for transmission over correlated narrowband Rayleigh fading channels. The effect of different SBC schemes having diverse minimum Hamming distances ($d_{H,min}$) and code rates on the attainable system performance was quantified, when using iterative SBSD and channel decoding, while keeping the overall bit-rate budget constant

[1] PSNR is the most widely used and simple form of objective video quality measure, which represents the ratio of the peak to peak signal to the root mean squared noise [1].

by appropriately partitioning the total available bit rate budget between the source and channel codecs. EXIT charts were used for analysing the attainable system performance and it was observed from the EXIT-chart analysis that the convergence behaviour of ISCD is substantially improved with the aid of SBCs.

The above-mentioned investigations evolved further by designing more sophisticated non-coherent-detection aided space time coding based Multiple-Input Multiple-Output (MIMO) schemes for near-capacity video transmissions without the need for any high-complexity MIMO channel estimation. Space time coding constitutes an effective transmit diversity technique of compensating the effects of wireless channels by exploiting the independent fading of the signal transmitted from multiple antennas. Space-time coding is capable of achieving a substantial diversity and power gain relative to its single-input and single-output counterpart, which is attained without any bandwidth expansion. More specifically, we proposed a new near-capacity Sphere Packing (SP) modulation aided Differential Space Time Spreading (DSTS) design for the transmission of the video coded stream. SP modulation is a specific scheme, which maintains the highest possible Euclidean distance of the modulated symbols, while constitutes DSTS a low-complexity MIMO technique that does not require any channel estimation, since it relies on non-coherent detection.

Finally, in order to circumvent the BER floor imposed by conventional two-stage turbo-detection schemes, we considered jointly optimised three-stage source and channel decoding arrangements employing serially concatenated and iteratively decoded SBCs combined with a URC and multi-dimensional SP modulation. The mutual information between the SBC, URC and SP constituent components is iteratively exploited in a turbo process in order to improve the overall BER and objective video quality in terms of the PSNR. The resultant coded signal was transmitted using a non-coherently detected DSTS MIMO-aided transceiver designed for near capacity JSCD. The performance of the system was evaluated by considering interactive video telephony using the H.264/AVC source codec. Again, the convergence behaviour of the MIMO transceiver advocated was investigated with the aid of EXIT charts.

Acknowledgements

I will like to express my gratitude to my supervisor, Professor Lajos Hanzo for his kindness, guidance and support. I am really obliged to him for his gracious assistance, invaluable guidance and inspirational attitude during my supervision. I consider myself very fortunate to be a member of this learned research group and have greatly benefitted from its vast expertise, precise methods of research and the sincerity of its members towards the pursuit of knowledge.

I will like to appreciate the support provided to me by all my colleagues. I would also like to express my gratitude to Dr. Rob Maunder and Dr. Mohammed El-Hajjar for their support and sharing their knowledge. I am grateful to Dr. N. S. Othman for her assistance. Special thanks to my colleagues Dr. Raja Ali Riaz and Muhammad Fasih U Butt for their utmost kindness and overall for their friendship.

I also gratefully acknowledge the financial support I received from the University of Engineering & Technology, Peshawar under the auspices of the Higher Education Commission, Pakistan.

DECLARATION OF AUTHORSHIP

I, **Nasruminallah**,

declare that the thesis entitled

Low Bit-Rate Video Joint Source-Channel Decoding Aided Wireless Communication

and the work presented in the thesis are both my own, and have been generated by me as the result of my own original research. I confirm that:

- this work was done wholly or mainly while in candidature for a research degree at this University;
- where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
- where I have consulted the published work of others, this is always clearly attributed;
- where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
- I have acknowledged all main sources of help;
- where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
- parts of this work have been published as: [2–11].

Signed: *Nasru Minallah*
Date: 15 June 2010

List of Publications

Journal Papers:

1. **Nasruminallah** and Lajos Hanzo, “Short block codes for guaranteed convergence in soft-bit assisted iterative joint source and channel decoding”, IEE Electronics Letters, pp. 1315-1316, September 2008.

2. **Nasruminallah** and Lajos Hanzo, “EXIT-Chart Optimised Short Block Codes for Iterative Joint Source and Channel Decoding in H.264 Video Telephony”, IEEE Transactions on Vehicular Technology, April 2009.

Conference Papers:

1. **Nasruminallah** and M. El-Hajjar and Noor Othman and A.P. Quang and Lajos Hanzo, “Over-Complete Mapping Aided, Soft-Bit Assisted Iterative Unequal Error Protection H.264 Joint Source and Channel Decoding”, in Proceedings of the IEEE Vehicular Technology Conference IEEE VTC’08 (Fall), September 2008.

2. **Nasruminallah** and Lajos Hanzo, “Convergence Behaviour of Iteratively Decoded Short Block-Codes in H.264 Joint Source and Channel Decoding”, in Proceedings of the IEEE Vehicular Technology Conference IEEE VTC’08 (Fall), September 2008.

3. Du Yang and **Nasruminallah** and Lie-Liang Yang and Lajos Hanzo, ”SVD-aided Unequal-Protection Spatial Multiplexing for Wireless Video Telephony”, in Proceedings of the IEEE Vehicular Technology Conference IEEE VTC’08 (Fall), September 2008.

4. **Nasruminallah**, Robert G. Maunder,and Lajos Hanzo, “Iterative Detection Aided H.264 Wireless Video Telephony Using Irregular Convolutional Codes”, in Proceedings of the IEEE Vehicular Technology Conference IEEE VTC’09 (Spring), April 2009.

5. **Nasruminallah**, Mohammed El-Hajjar and Lajos Hanzo, “Robust Transmission of H.264 Coded Video Using Three-Stage Iterative Joint Source and Channel Decoding”, submitted in IEEE Globecom 2009 Wireless Communications Symposium, April 2009.

6. **Nasruminallah** and Lajos Hanzo, “Exploiting Redundancy In Iterative H.264 Joint Source and Channel Decoding For Robust Video Transmission”, in Proceedings of the IEEE Vehicular Technology Conference IEEE VTC’10 (Spring), April 2010.

7. **Nasruminallah**, Mohammad Fasih Uddin Butt, Soon Xin Ng and Lajos Hanzo, “H.264 Wireless Video Telephony Using Iteratively-Detected Binary Self-Concatenated Coding”, in Proceedings of the IEEE Vehicular Technology Conference IEEE VTC’10 (Spring), April 2010.

8. **Nasruminallah**, Mohammed El-Hajjar and Lajos Hanzo, “Iterative H.264 Source and Channel Decoding Using Sphere Packing Modulation Aided Layered Steered Space-Time Codes”, in Proceedings of the IEEE International Conference on Communications (IEEE ICC’10), May 2010.

Contents

List of Symbols

$Y_{ref}(i,j)$	The pixel values of reference frame.
$Y_{proc}(i,j)$	The pixel values of processing frame.
$H(x)$	The entropy of symbol x.
g	The generator polynomial of convolutional code.
f_D	The normalised Doppler frequency.
f_d	The doppler frequency.
T_s	The symbol duration.
x_i	The source coded bit-stream.
x_a	The partition A bit-stream.
x_b	The partition B bit-stream.
x_c	The partition C bit-stream.
$x_{a'}$	The partition A bit-stream after interleaving.
$x_{b'}$	The partition B bit-stream after interleaving.
$x_{c'}$	The partition C bit-stream after interleaving.
y_a	The RSC coded stream of Partition A.
y_b	The RSC coded stream of Partition B.
y_c	The RSC coded stream of Partition C.
$L_M(\hat{y}_i)$	The received soft values of the transmitted stream.
$L_M(\hat{y}_a)$	The received soft values of Partition A.
$L_M(\hat{y}_b)$	The received soft values of Partition B.
$L_M(\hat{y}_c)$	The received soft values of Partition C.
$\hat{x'}_a$	The RSC decoded Partition A stream.
$\hat{x'}_b$	The RSC decoded Partition B stream.
$\hat{x'}_c$	The RSC decoded Partition C stream.
$\hat{x}_a$	The deinterleaved RSC decoded Partition A stream.
$\hat{x}_b$	The deinterleaved RSC decoded Partition A stream.
$\hat{x}_c$	The deinterleaved RSC decoded Partition A stream.
$\hat{x}_i$	The estimated source signal.
Π	The interleaver.
Π^{-1}	The einterleaver.
n	The additive White Gaussian noise.
N_0	The noise power spectral density.
σ_n^2	The complex AWGN variance.
E_b	The bit energy.
$L_{.,a}$	The *a priori* information.
$L_{.,a}$	The *a posteriori* information.
$L_{.,a}$	The *extrinsic* information.
$d_{H,min}$	The minimum Hamming distance.

$V_{m,k}$	The source coded parameters.
$V'_{m,k}$	The bit-mapped source coded parameters.
V_m	The bit-mapped code book.
$\mid V_m \mid$	The bit-mapped code book size.
$\mid w_m \mid$	The length of mapped bit pattern.
$P(x)$	The probability mass value.
$L_{CD}^{[\text{ext}]}(x)$	The extrinsic information of channel decoder.
$L_{USSPA}^{[\text{ext}]}(x)$	The extrinsic information of source decoder.
$\theta(\text{x})$	The reliability of x.
I_A	The *a priori* information.
I_E	The *extrinsic* information.
$\oplus$	The XOR operation.
B_{x}^{o}	The overlapped block in MCTI.
B_{x}^{no}	The non-overlapped block in MCTI.
$H1$	The transformation matrix applied to all blocks of frame.
$H2$	The transformation matrix of Hadamard transform of size 4x4.
$H3$	The transformation matrix of Hadamard transform of size 2x2.
L	The number of legitimate Space-time signals.
R^4	The four dimensional real valued Euclidean Space.
S^l	The sphere packing symbols.
B_{sp}	The number of binary bits per sphere packing constellation symbol.
L	The number of modulation symbols in the sphere packing signaling alphabet.
V_k	The source signal parameter set.
M	The number of scalar source codec parameters per parameter set.
a	The fadding factor.
$L_{CD}^{[\text{ext}]}(x)$	The extrinsic information from artificial channel coding redundancy.
$L_{USSPA}^{[\text{ext}]}(x)$	The extrinsic information from natural residual source redundancy.
$S(l)$	The encoder trellis state for l-th input bit.
$\gamma_l(.)$	The reliability information for each valid trellis state transition.
$\alpha(.)$	The contribution from preceding trellis stages.
$\beta_l(.)$	The reliability information from succeeding trellis stages.
$D_{M,a}$	The soft information extracted from the demodulator.
K	The number of bits per symbols.
M	The number of 2^k-ary symbols.
P	The number of parity bits.
$d_{H,min}$	The minimum Hamming distance.
R	The coding rate.
I_t	The number of decoding iterations.
α^n	The fraction of encoded bit-stream generated by n-th component code.
CC^n	The n-th component convolutional code.

I_n	The number of constituent sub-codes.
I_{system}	The number of decoding system iteration.
I_{out}	The outer iteration.
I_{in}	The inner iteration.

$\mathrm{P}(x \mid z)$	The *a posteriori* probability for a single data bit x given the complete received sequence z.
$\mathrm{L}(x \mid z)$	The *a posteriori* L-value for a single data bit x given the complete received sequence z.
$I_{.,A}(x)$	The mutual information between *a priori* LLR value $L_a(x)$ and the corresponding symbol x.
$I_{.,E}(x)$	The mutual information between *extrinsic* LLR value $L_e(x)$ and the corresponding symbol x.

Chapter 1

Basics of Video Coding

1.1 History of Video Coding

Recent advances in the world of telecommunication and multimedia systems resulted in the design of improved transmission techniques. However, bearing in mind the volume of information produced by high definition multimedia communication systems and the limited availability of unoccupied bandwidth at carrier frequencies, where beneficial propagation conditions prevail, the design of efficient multimedia systems requires careful attention. Hence the design of improved video coding techniques is important for the successful implementation of various multimedia communication systems, in order to reduce the amount of information required for flawless interactive multimedia communications [1]. An overview of advances in the field of video coding is presented in Table 1.1.

1.2 Analogue Video

At the output of the video camera, the video signal is generated by scanning a two-dimensional scene. Normally, the scanning operation commences at the top left corner of the scene and ends at its bottom right corner. Each video clip is comprised of a number of scanned frames. The number of scan-lines per frame and pixels per line, has a direct relation to its resolution. Increasing the number of lines per frame results in increasing the video resolution and - provided that the compression ratio remain constant, the bandwidth is also increased. Additionally, the number of scanned frames per second determines the temporal resolution of the video, which should always be kept above the human eye's fusion frequency to avoid video flicker. There are two types of scanning methods:

1.2.1 Progressive and Interlaced Scanning

In progressive scanning, the video frame is formed from the natural scanning of the picture. By contrast, in interlace scanning [12], we could argue that two consecutive frames of the video scene are scanned jointly,

Table 1.1: History of Video Compression Standards

Year	Standard	Standardised by	Application Target bit-rate	Typical applications features
1984	H.120	ITU-T	2 Mb/s(Eu) or 1.544 Mb/s (North America)	Video Conf. Reasonable Spatial resolution/temporal quality
1990	H.261	ITU-T	384Kb/s (Multiples of 64Kbps)	Video Conf. ISDN Networks
1993	MPEG1	ISO	1.5 Mb/s (also higher)	Storage (CD, HD), progressive video coding
1996	MPEG2 (or H.262)	ISO, ITU-T	4-9Mb/s	Digital broadcast TV DVD, interlaced video coding, scalability
1996	H.263	ITU-T	10-64 Kb/s	Low bit rate coding, PSTN, Narrowband ISDN Extensions: H.263+, H.263++, H.263L
1998	MPEG4	ISO/IEC	Low and medium	Multimedia applications (scalability/error resilience/ object-based mixing of synthetic and natural)
2003	H.26L or H.264 or MPEG4-Part10 or AVC	ITU-T, ISO	wide range	Low bit rate video archival/streaming Advanced video coding for generic audiovisual services
2007	H.264/SVC	ITU-T, ISO	wide range	Low bit rate Streaming Conferencing Surveillance Broadcast Storage

leading to interleaved scan lines, as shown in Figure 1.1. This results in two fields, and joining them results in a complete frame. The basic idea behind the interlaced scanning is to provide a trade-off between the achievable spatial and temporal resolution, because for slowly moving objects it results in a high spatial resolution, since there is hardly any change between consecutive fields. By contrast, for fast moving scenes it avoids flicker, because the scene is displayed at field rate - i.e. at twice the frame-rate. Although, the spatial resolution is

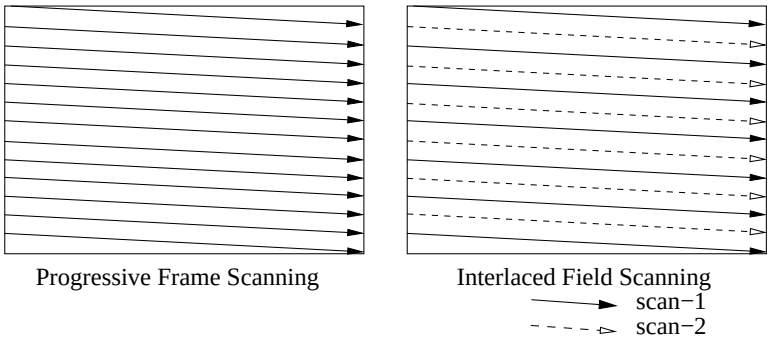

Figure 1.1: Progressive and interlaced scanning.

indeed reduced, this is not perceivable for rapidly moving objects. Albeit the employment of interlaced video is beneficial for digital television displays, it is not considered a suitable method of choice for computer displays, because they typically show fine-resolution information consisting of text and graphs.

1.3 Digital Video

Analogue video should be converted to a digital format, in order to facilitate video storage and compression [13]. This conversion is performed with the aid of the following three operations:

1. Filtering [14]
   Filtering is applied to the analogue signal prior to its sampling, in order to avoid aliasing.

2. Sampling [15]
   Analogue video is converted to a time-discrete form by sampling. The minimum rate at which sampling can be performed is specified by the Nyquist rate, which is twice the signal's bandwidth.

3. Quantisation [16]
   The discrete time-signal obtained after the process of sampling is then converted to its discrete-amplitude digital form, after the process of quantisation. Video broadcast applications typically use eight-bit per sample resolution for quantisation.

1.4 Colour Spaces

There are two types of colour formats, used to represent video signals, namely the Red-Green-Blue (RGB) [17] and the so-called YUV [18] formats:

1. RGB
   The video signals generated by the camera during the process of scanning represent the red, green and blue colour components.

2. YUV
   The three colour components of the RGB format are highly correlated. Therefore, to characterise colour components in terms of their relative importance and for the sake of compatibility with existing black and white video, another colour format was designed, which is known as the YUV colour format, where Y represents the luminance and U and V represent the chrominance or colour components of the video. The digital counterpart of YUV is YC_bC_r. Due to the sensitivity of human eye to the luminance component, it is given more importance in video processing, relative to the chrominance component.

The expected spatial and temporal resolution of broadcast-type video is specified by the Consultative Committee for International Radio (CCIR) communications under its recommendation CCIR-601 [19], consisting of two standards, namely:

1. CCIR-601/625 [20]
   This standard is followed in Europe, having a 625 scan-line per video frame spatial resolution and recorded/scanned at 25 frames per second temporal frame rate. In this system the number of active video lines per frame is 576.

2. CCIR-601/525 [20]
   This recommendation is followed in North America and in the Far East. It consists of 525 lines per frame and scanned at a frame rate of 30 frames per second. In this system the number of active video lines used for digital television is 480. The number of picture elements (pixels) per line in both these systems is 720. Therefore, the number of pixels generated per second by both of these systems is the same, although they have different number of lines per frame and a different frame rate. For the CCIR-601/625 system the total number of pixels per second is $25 \times 576 \times 720 = 10\,368\,000$. Similarly, for the CCIR-601/525 system the total number of pixels per second becomes $30 \times 480 \times 720 = 10\,368\,000$.

Considering eight bits per pixel, the resultant total bit rate generated by any one of these two systems becomes $10\,368\,000 \times 8 \times 3 = 248\,832\,000\ bits/s = 248.832\ Mbits/s$. Since this would require a significantly higher bandwidth than the corresponding analogue TV bandwidth, the need for efficient compression techniques becomes evident.

1.5 Spatial Sampling Patterns

The spatial sampling patterns employed specify the relative number of luminance (Y) and chrominance (C_b and C_r) pixels per video frame. The commonly used sampling patterns are [21]:

- 4:4:4
  This pattern represents an equal number of luminance and chrominance pixels per line, which results in an equal bandwidth required for each video component.

- 4:2:2

In this pattern the number of chrominance pixels per line is half of the luminance pixels, therefore, the resultant bandwidth required for the luminance component is double of the chrominance component.

- 4:2:0
  In 4:2:0 format, both the horizontal and vertical resolutions of the chrominance component are half of the luminance resolution. Therefore, the bandwidth requirement of the chrominance component is a quarter of the luminance component.

- 4:1:1
  In the sampling format for 4:1:1 the chrominance component has the same vertical resolution as that of the luminance component, while the horizontal resolution is quarter. Therefore, the number of pixels generated by this format is exactly the same as that of the 4:2:0 format.

1.6 Image Formats

Various applications necessitate different image formats [21], depending on their user requirements and affordability. For example, video telephony and video conferencing applications require image formats associated with a smaller screen size and lower resolution, relative to studio-quality applications, in order to provide a reasonable subjective video quality. Similarly, High Definition Television (HDTV) is typically associated with a high bandwidth and advanced display capability, hence requiring substantially improved image resolution and quality. Regardless of the specific video coding applications, the source input is typically provided by the video capturing device in CCIR-601 video format. Therefore, the different image formats employed in various video applications are defined relative to CCIR-601. Some of the frequently used image formats are given in Table 1.2 and are defined below.

1.6.1 Source Input Format [22]

The Source Input Format (SIF) type of image represents a lower resolution format for storage applications. The horizontal, vertical and temporal dimensions of the image stored in SIF image format are half of that in the CCIR-601 standard [23]. The required horizontal and vertical resolution is achieved by first filtering and then sub-sampling the image in both horizontal and vertical direction. Halving the temporal resolution can be achieved by either dropping alternate frames or by averaging consecutive pairs of frames. Therefore, the resultant SIF format relative to the European CCIR-601 standard consists of 360 pixels horizontal and 288 lines vertical resolution, where the temporal frame rate is 12.5 frames per second (i.e. 25 fields per second). By contrast, the North American and Far East standard consists 360, 240 and 15 frames per second (i.e. 30 fields per second).

1.6.2 Common Intermediate Format [22]

The Common Intermediate Format (CIF) constitutes a common intermediate representation of both the North American and Far Eastern versions of CCIR-601 video sources for ensuring the compatibility of worldwide video-conferencing applications [23]. This is achieved by selecting half of the largest possible vertical and temporal resolution in CCIR-601 sources. The resultant vertical resolution is given by half of the number of active picture lines (i.e. 576/2=288 lines) in the 625/50 system, while halving the temporal resolution of 525/60 results in 30 fields per second. Both the 625/50 and the 525/60 standards have a common horizontal resolution of 720 pixels per line, therefore, half of this results in a 360 pixels per line resolution for the CIF format. Due to this selection of intermediate values for the image format, it is referred to as the CIF.

1.6.3 QSIF, QCIF and SubQCIF

Some applications, such as video-telephony over low bit rate mobile networks [24] require the further reduction of the spatio-temporal resolution. For this reason the horizontal and vertical resolution of the SIF and CIF format image may be halved in each direction, and the resultant formats are referred to as Quarter Source Input Format (QSIF) and Quarter Common Intermediate Format (QCIF) formats, respectively. For certain applications, which necessitate even further reductions in the spatio-temporal resolution, a SubQCIF image format associated with the smallest standard image dimensions of 128 by 96 pixels is defined. For SubQCIF images the frame rate can be as low as 5 frames per second.

1.7 Video Quality Evaluation

There are two methods of video quality assessment, with reference to the original video, namely subjective and objective evaluation techniques, which are detailed below [25–27].

1.7.1 Subjective Video Quality Evaluation

In the spirit of this traditional method of video quality assessment [28], a group of subjects is presented with the processed video sequence that was subjected to possible impairments, and their opinion regarding the quality of the video sequence is sought with reference to the unimpaired reference video sequence. The subjective response is obtained by giving a score in the range of 1 to 100, representing bad, poor, fair, good and excellent quality. The objective association of the response is then obtained by classifying the responses into 20-point intervals (i.e. 1-20=bad, 21-40=poor, 41-60=fair, 61-80=good, 81-100=excellent). Finally, the average of the viewers scores defined as Mean Opinion Score (MOS) provides the measure of the video quality. About 20-25 non-expert users are required to record reliable MOS scores for subjective video quality assessment. In order to have a fair subjective video quality evaluation, we introduced two methods for analysing the relative performance of different systems, namely the

- Average Subjective Video Quality;

- Cumulative-Error Subjective Video Quality [29].

Table 1.2: Different image formats

Format	Resolution	
	Europe	North America
SIF	360x288	360x240
CIF	360x288	360x288
QSIF	180x144	180x120
QCIF	176x144	176x144

1.7.1.1 Average Subjective Video Quality

According to this method [28] the process of source coded video transmission and decoding is performed N times, in order to obtain N decoded video clips, each associated with a different error distribution when transmitted through the transmission system. Then the average of the N decoded sequences is obtained upon performing pixel by pixel averaging of the N decoded video sequences.

1.7.1.2 Cumulative-Error Subjective Video Quality

Similarly to the average subjective video quality method, the source coded video is transmitted through the transmission system considered N times. Then the final cumulative-error subjective video quality is evaluated by copying all the channel-induced transmission errors into each of the N decoded video frames, in order to obtain a single video sequence from the error accumulation of N video clips.

1.7.2 Objective Video Quality Evaluation

The Peak Signal-to-Noise Ratio (PSNR) is the most widely used form of objective video quality evaluation [25], which represents the ratio of the highest possible video signal power of the 8 bits/pixel luminance signal, i.e. 255^2 to the video reconstruction error power formulated as [1]:

$$PSNR = 10log_{10}\left[\frac{255^2}{(1/N)\sum_i\sum_j(Y_{ref}(i,j)-Y_{proc}(i,j))^2}\right][dB]. \quad (1.1)$$

More explicitly, the video reconstruction error power is represented by the square of the pixel-by-pixel difference between the pixel values of the reference frame $Y_{ref}(i,j)$ and the pixel values of the frame under evaluation, namely of $Y_{proc}(i,j)$. The rationale behind this definition is that we would like to avoid that the Signal to Noise Ratio (SNR) becomes dependent on the brightness/darkness of the test video sequence, this is why the peak-value of 255^2 is used instead of the true signal power.

Although there are some weaknesses of the PSNR based objective video quality evaluation method, owing to its appealingly simple evaluation process it is considered the most popular method of choice [30].

To elaborate a little further, one of the weaknesses of the PSNR method of evaluation is its inability to characterise the subjective human interpretation of certain types of distortion, which is perceived by the human vision system differently in different parts of the image. For example, a small block of severely distorted pixels in the subjectively most sensitive eye-catching areas of the frame, such as the human face attracts the viewer's attention and hence it is perceived as a substantial distortion in the frame, although it might hardly affect the PSNR. These artifacts generally result in large discrepancies among the subjectively and objectively evaluated image qualities. However, under similar conditions if one system has a better PSNR performance than the other, then the same system will also have better subjective video quality, under the same conditions relative to the other system. Hence the PSNR is the most frequently used metric for evaluating the performance of various video processing and communication systems.

1.8 Fundamental Building Blocks of Video Compression

The main focus of video compression theory [1] is based on the removal of redundancy from the video sequence, which exhibits itself in the form of both spatial and temporal correlation of the video frames. Additionally, the removal of certain information without this becoming subjectively objectionable by the human vision system can be exploited for further video bit-rate reduction. In terms of video coding the spatial correlation reduction process is typically referred to as intra-frame coding [31], while the temporal correlation reduction technique is referred to as inter-frame coding [32]. Furthermore, lossless Variable Length Coding (VLC) techniques [33] may be used to reduce the redundancy between the coded data symbols, which is referred to as entropy coding. Therefore, the resultant three fundamental building blocks of video compression schemes employed for redundancy reduction are:

1. Intra-frame coding;
2. Inter-frame coding;
3. Entropy coding.

1.8.1 Intra-frame Coding [31]

The main objective of intra-frame coding is to reduce the spatial redundancy among the pixels within the frame. The key compression techniques involved in intra-frame coding are predictive coding, transform coding and quantisation.

1.8.1.1 Predictive Coding [34]

The most simple and basic method of spatial redundancy reduction is to predict the values of the pixels based on the previously coded values and then to encode and transmit the prediction error. This method is referred

to as Differential Pulse Code Modulation (DPCM) [35]. The block diagram of DPCM is shown in Figure 1.2, where the prediction error values are quantised and entropy coded before transmission. As seen in Figure 1.2, at the receiver the inverse-quantised prediction error signal is then added to the predicted signal, in order to get back the original signal. The neighbouring pixels within the same frame or the previous frame can be used for prediction. Again, invoking spatial prediction based on the pixels within the frame is referred to as intra-frame predictive coding, while temporal prediction based on the previous frame is referred to as inter-frame predictive coding. A combination of inter- and intra-frame predictive coding is often referred to as hybrid predictive coding [36]. The minimum number of bits that can be assigned to each prediction error value is one, therefore, the compression achieved by this method in its own might in isolation from other techniques does not satisfy the requirements of low bit-rate video coding, unless a block of pixels is predictively coded together. Having a low complexity and low latency is one of the advantages of DPCM, and this technique is often used in coding the motion vectors during the process of video coding.

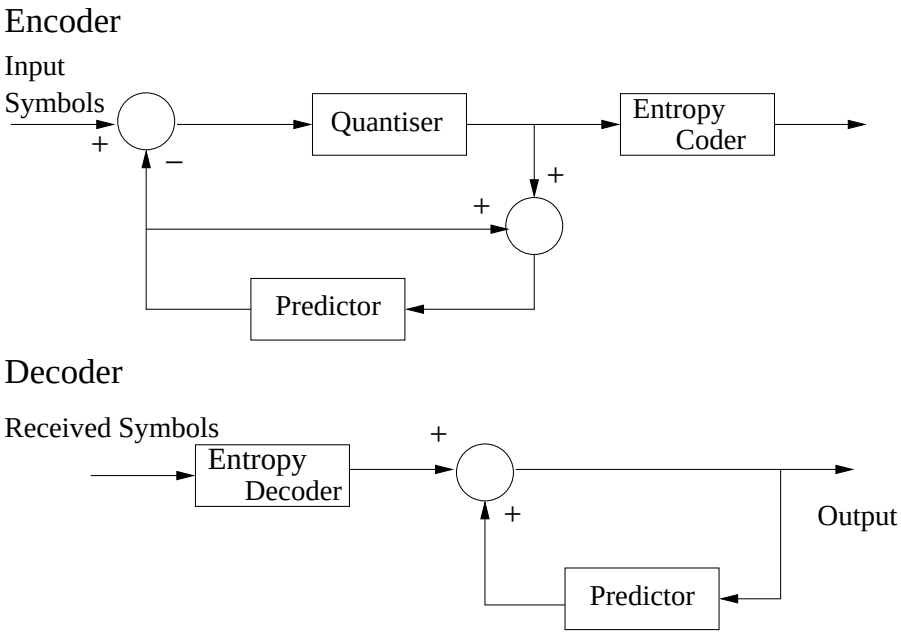

Figure 1.2: DPCM schematic often used for coding various video codec parameters.

1.8.1.2 Transform Coding [37]

In this process the original video frame is transformed into the spatial frequency domain, where most of the image energy is concentrated in the low frequency transform coefficients. Therefore, it does not result in excessive video distortion, if we encode only these low-frequency coefficients, while discarding the less significant low-energy, high-frequency coefficients throughout the process of quantisation. Hence we arrive at an acceptable reconstruction quality at a low data rate.

1.8.1.3 Quantisation [38]

The transformation of the video frames alone does not actually result in any compression. It basically transforms the 64 pixel values of an 8x8 video block to 64 transform coefficient values and the total energy in both the original and transformed domains remains the same, and therefore results in no compression. However, provided that the original video blocks to be transformed exhibited spatial correlation [39], the advantage of the transformation is that most of the image energy is concentrated in few low-frequency transform coefficients, while the remaining coefficients have a low energy, hence they become zero after quantisation. Compression is basically achieved through the process of quantisation and VLC. Additionally, by exploiting the characteristics of the human vision system, which is less sensitive to high frequency coefficients, further compression can be achieved by applying coarser quantisation to the high frequency coefficients, which forces more coefficients to zero and as a result an increased compression is achieved, while more fine quantisation can be applied to the low frequency coefficients. Therefore, the quantisers are broadly divided into two categories, namely Uniform Quantisers (UQ) [38] and Non-uniform Quantisers (NUQ) [40]. The step size q in UQs remains fixed across the entire dynamic range of the quantiser. Typically, a so-called quantiser deadzone is employed, for example to eliminate small noise-like samples. The width of this deadzone is represented in Figure 1.3 by the threshold 'T' of the quantiser. Normally, a zero dead zone is used to quantise the intra-frame DC coefficients, while a non-zero deadzone is used to quantise the inter-frame prediction error AC and DC coefficients.

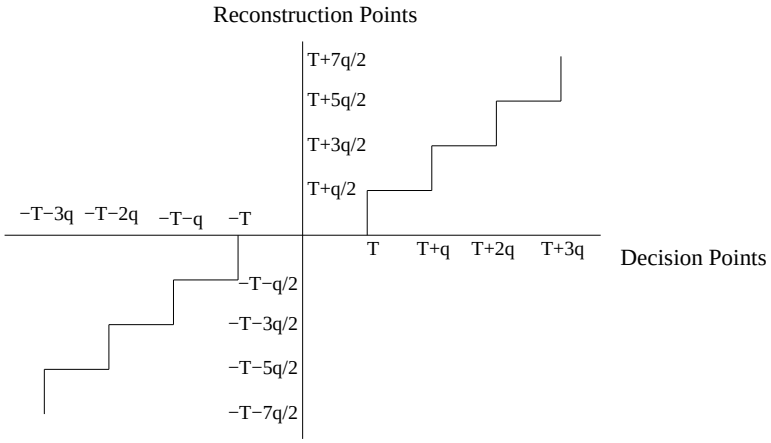

Figure 1.3: Quantiser having a deadzone width of T.

1.8.2 Inter-frame Coding [41]

Inter-frame coding is used to reduce the temporal redundancy. By taking the difference between the successive images, the temporal redundancy is significantly reduced. The static background portions of the images typically result in zero difference, while those associated with motion result in significant errors and hence are coded for transmission.

Figure 1.5 shows the frame difference between the 4th and 5th frame of the CIF-format (288x352) "foreman" video sequence, which was obtained using the contrast-stretching function of Figure 1.4, in order to make it visible for the naked eye. The contrast-stretching function of Figure 1.4 is used to constrain the input levels

i lower than w to a darker level z in the output image. Similarly, it constrains the values above m to lighter values, which results in an output image having an improved contrast. The function shown in Figure 1.4 may be expressed as:

$$z = T(i) = \frac{1}{1 + (w/i)^X}, \tag{1.2}$$

where i represents the intensities of the input image, z is the corresponding intensity values in the output image, and X controls the slope of the function. More specifically, in the contrast stretching example of Figure 1.5 we considered w=225 and X=100.

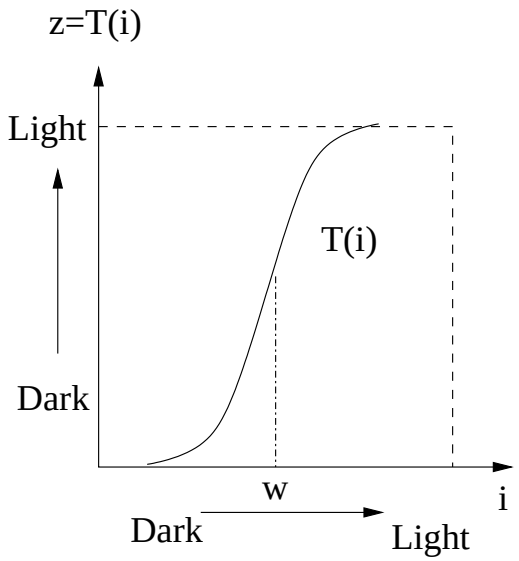

Figure 1.4: Contrast stretching transformation function.

However, this simple frame-differencing based technique does not cater for the fact that different parts of the video frame might move in different directions. In order to cater for this scenario, each (8x8)-pixel block may be allowed to move to an arbitrary position with its own search-scope surrounding it, before the Motion Compensated Error Residual (MCER) is computed. As seen in Figure 1.7, a pair of horizontal and vertical Motion-Vectors (MVs) are required to signal these positions to the decoder for each individual (8x8) block, before the original block may be reconstructed. As an example, Figure 1.6 shows the frame difference of the 5th frame from its motion compensated counterpart. It can be observed that the MV-based motion compensated frame difference exhibits a considerably reduced inter-frame difference.

1.8.2.1 Motion Compensation [42]

To elaborate a little further on the subject of motion compensation, the motion of the object have to be estimated, which is commonly performed by using Block Matching Algorithms (BMA) [43], which typically assume that the frame is usually partitioned into blocks of $N \times N$ pixels [44]. Given the maximum motion displacement of x pixels per frame shown in Figure 1.7, the block of pixels considered is matched within the corresponding search-scope of $(2 \times x + N)$ pixels of the previous frame. The best match within the previous frame is assumed to be the motion displacement of the current block relative to the same block in the previous frame. Again, this motion displacement vector is referred to as the motion vector [45]. Various matching criteria, such as the

Figure 1.5: Inter-frame difference.

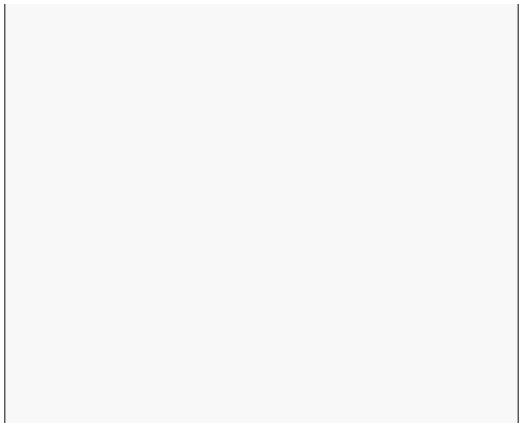

Figure 1.6: Motion compensated inter-frame difference.

Cross Correlation Function (CCF) [46], Mean Squared Error (MSE) [47], Mean Absolute Error (MAE) [48] and PSNR [49] can be used to identify best matching blocks, which are defined as follows:

$$MSE(i,j) = \frac{1}{N^2}\sum_{m=1}^{N}\sum_{n=1}^{N}(Y(m,n) - Y_{ref}(m+i,n+j))^2, \quad -w \leq i,j \leq w \tag{1.3}$$

$$MAE(i,j) = \frac{1}{N^2}\sum_{m=1}^{N}\sum_{n=1}^{N}|Y(m,n) - Y_{ref}(m+i,n+j)|, \quad -w \leq i,j \leq w \tag{1.4}$$

$$PSNR(i,j) = 10log_{10}\left[\frac{255^2}{(1/N)\sum_{m=1}^{N}\sum_{n=1}^{N}(Y(m,n) - Y_{ref}(m+i,n+j))^2}\right], \quad -w \leq i,j \leq w \tag{1.5}$$

The evaluation of these block-matching metrics imposes different complexities. Various fast motion search methods have been introduced in the literature [50]. The idea of fast motion search methods is to identify best matching blocks by considering the lowest number of search points or comparisons. Some of these methods include Two-dimensional Logarithmic (TDL) search [51], Three-Step Search (TSS) [51], the Cross Search Algorithm (CSA) [51], Orthogonal Search [51] and Diamond search [52], which are briefly highlighted below.

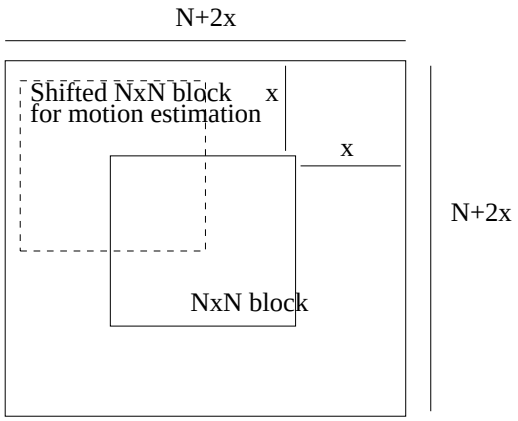

Figure 1.7: Motion estimation search window.

1. FULL-SEARCH [53]:
   Naturally, full-search is the most complex motion-search technique, where the motion vectors are calculated using full search by exhaustively calculating the matching criterion for the entire search window. Considering an (NxN)-pixel block size and an (N+2x)$\times$(N+2x) pixel search window size, the full search requires $(2x+1)^2$ evaluations of the specific block matching criterion employed. The evaluation of the video encoder's complexity shows that potentially 50-70 % of the encoder's complexity is imposed by the motion estimation. Therefore, the employment of a computationally efficient motion estimation algorithm is desirable.

2. DIAMOND SEARCH [54]:
   According to this search algorithm the motion vectors are found using two diamond-shaped search patterns. The first pattern is referred to as the Large Diamond Search Pattern (LDSP) and consists of nine search points, where eight points surround the center one, while obeying a diamond shape. The second pattern consist of the Small Diamond Search Pattern (SDSP), which is constituted by five check points forming a small diamond shape, as shown in Figure 1.8. This algorithm may be described as follows.

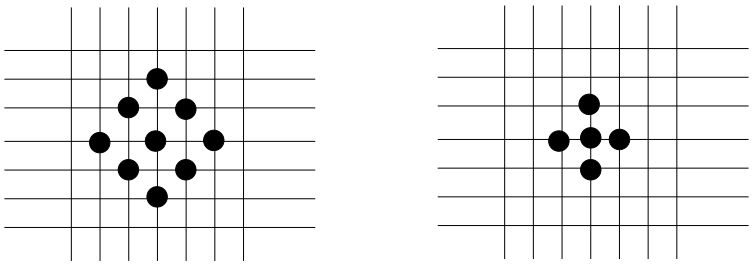

Figure 1.8: Diamond search patterns.

- Step-1:
  The LDSP is centered at the origin of the search window, and the matching criterion is applied only at the nine check points. If the best match is found at the centre of the search pattern, then the algorithm

proceeds to step-3, otherwise to step-2.

- Step-2:
  The best match found in the previous step is considered as the origin of the new LDSP and Step-1 is repeated.

- Step-3:
  The search procedure then switches to the SDSP considering the origin as the center of the previous LDSP. The best match found in this step becomes the final motion vector.

1.8.3 Entropy Coding [39]

Entropy coding is a well-established compression technique, which achieves further compression by reducing the redundancy among arbitrary symbols, provided they exhibit different probability of occurance. In video codecs typically the transform coefficients and motion vectors are entropy coded using appropriate VLC techniques. To elaborate a little further, the idea of VLC is to assign a lower number of bits to highly probable values, while long code words are assigned to symbols associated with a low probability. The number of bits assigned to a VLC symbol and its probability are inversely proportional. The number of bits required to code these symbols is $1/\log_2\ p$, where p is the probability of the symbol. Therefore, the entropy of the symbols, which represents the minimum average number of bits required to code the symbol is given as:

$$H(x) = -\sum_{i=1}^{n} p_i\ log_2\ p_i. \tag{1.6}$$

The two most commonly used VLC methods are Huffman coding [1], and Arithmetic Coding [1]. Huffman coding is relatively simple VLC technique. Due to the employment of an integer number of bits per symbol the performance of Huffman coding cannot be as low as the entropy. By contrast, arithmetic coding may approach the entropy, because the symbols are not coded individually.

1.8.4 A Generic Low Bit-Rate Video Coding System

In recent years there has been significant interest in low bit-rate video coding [55]. Several low bit-rate standards emerged as a result of academic and industrial efforts in this area. Various efficient compression techniques have emerged as a result of these efforts [56]. These different standards include the Joint Photographic Experts Group (JPEG) recommendations [57], the International Telecommunications Union - Telecommunication (ITU-T) codecs H.261, H.263 [58] and the ISO/IEC Motion Picture Experts Group type 1 (MPEG-1) and Motion Picture Experts Group type 2 (MPEG-2) codecs [59], for example. These different standards were designed for different application areas, detailed in Table 1.1. They also have different requirements in-terms of their picture quality, bit-rate, error resilience, complexity and delay [60]. The coding algorithms of these different standards portray the video scene at an acceptable quality. A subjectively pleasing video quality is

achieved at a frame-size-dependent bit-rate. Some of the sophisticated low bit-rate video coders are capable of providing acceptable picture quality at such low bit-rates as 64Kbps, but below this rate these standards typically result in various annoying artifacts, such as blocking, staircase and mosquito noise artifacts. Similarly, at such low bit-rates the codec has to operate at low frame rates, which results in a low temporal resolution and a potential loss of lip-synchronisation. The requirement for enhanced compression and increased robustness against channel errors arises due to the considerable increase in demand for video telephony, video conferencing, security monitoring, interactive games, teleshopping, multimedia messaging and other value added services, while the transmission rates over wireless and wired Public Switched Telephone Networks (PSTN) are still rather limited [61]. This imposes significant challenges on digital video communications. Owing to the bit-rate limitations of wireless communications, there is considerable interest in low bit-rate video coding, requiring further compression improvements in order to reach an acceptable picture quality at low bit rates.

1.8.5 Building Blocks of Video Communications Systems

The building blocks involved in wireless video communications systems are presented in Figure 1.9. The functionalities of these blocks are highlighted below.

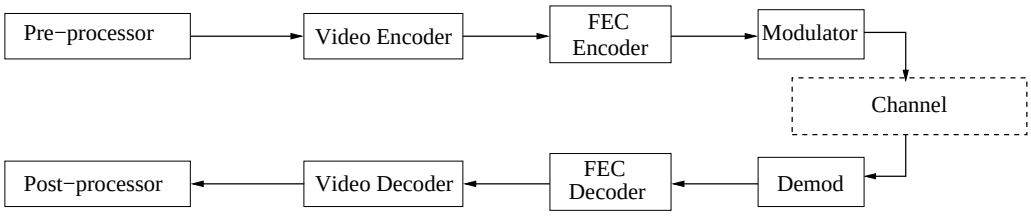

Figure 1.9: Generic Video Coding System.

1.8.5.1 Pre-Processor

The pre-processor represents the first processing unit in the low bit-rate video communication system. The different blocks used in the pre-processor are shown in Figure 1.10, which are used to generate a low bit rate image sequence.

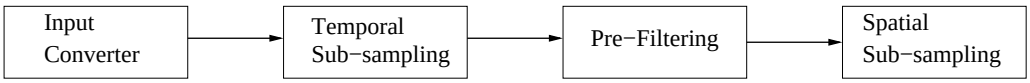

Figure 1.10: Design blocks of pre-processor.

The functionality of the pre-processor design blocks presented in Figure 1.10 are detailed below.

1. Input Converter

   The basic aim of the input converter is to convert the input video sequence represented in formats such as the Portable Pixel Map (PPM), RGB and YUV format into the YUV(4:2:0) image format, which is then used to perform different operations in the process of low bit-rate video coding.

2. Temporal Sub-sampling

   This process is used to sub-sample the input video sequence in the temporal domain, for the sake of arriving at a reduced frame rate, which in turn results in reduced bit-rate for the coded video sequence. Nonetheless, this results in a reduced correlation between the consecutive frames, which hence partially erodes the bit-rate savings of the reduced frame-rate.

3. Pre-Filtering

   The goal of pre-filtering is to remove high-frequency spatial information as much as possible, without degrading the perceptual image quality of the resultant video sequence. It involves a sequence of operations, which basically smoothenes image details, which are less important from a perceptual perspective. It also avoids the effects of aliasing throughout the subsequent step of spatial sub-sampling.

4. Spatial Sub-sampling

   Spatial sub-sampling is used to reduce the spatial redundancy of the input image sequence and to simultaneously reduce its spatial dimensions. Spatial sub-sampling is typically performed in both the vertical and horizontal direction in order to reduce the bit rate of the resultant video sequence. As an example, the resultant frame after applying spatial sub-sampling to the 240x360-pixel SIF resolution frame of Figure 1.11 is shown in Figure 1.12, which has a 120x180-pixel QSIF resolution.

Figure 1.11: SIF (240x360) format 'Mom' frame.

Figure 1.12: Resultant sub-sampled QSIF (120x180) format 'Mom' frame.

This low-resolution video obtained after pre-processing is then input to the encoder for further processing.

1.8.5.2 Video Encoder

The encoder of Figure 1.13 constitutes the basic element of low bit rate video coding. It performs diverse operations on the input bit stream in order to convert it into low bit rate video data. After decoding we regenerate the resultant output video clip. The block diagram of a generic inter-frame video encoder is shown in Figure 1.13. This encoder architecture is used in most of the standard video codecs, such as H.261, H.263, H.264, MPEG-1, MPEG-2 and MPEG-4 [62]. The different elements of this codec are described in the following paragraphs.

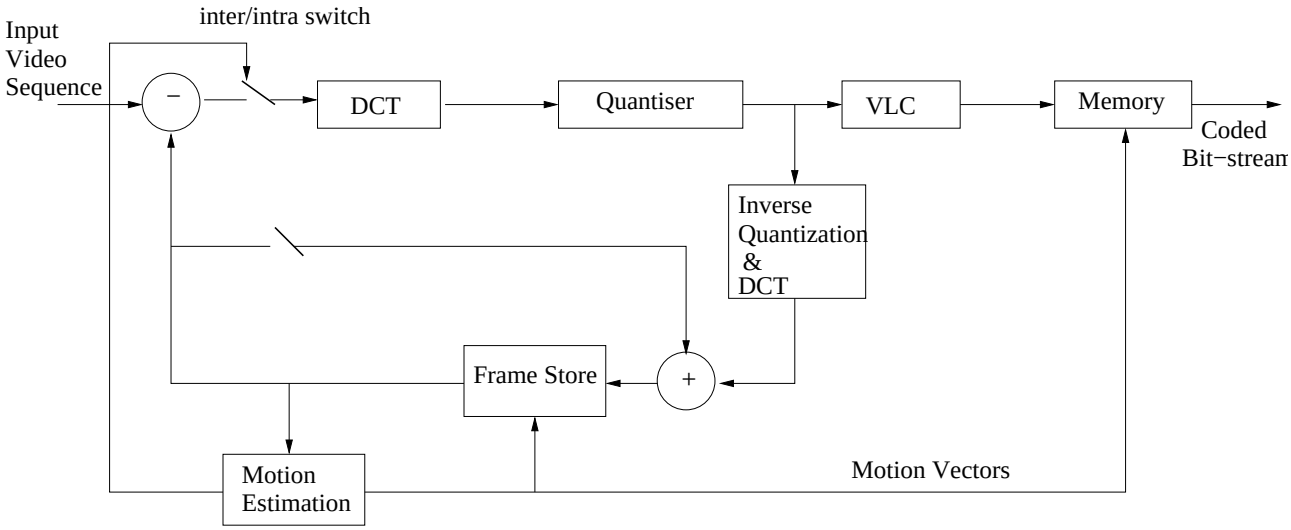

Figure 1.13: Generic inter-frame video encoder.

1. Inter-Frame Loop

   In the inter-frame loop, the prediction error is calculated by taking the difference of the current frame and its prediction from the previous frame. At the receiver this error signal is added to the corresponding previously decoded frame values in order to recover the current frame values. Generally, the magnitude of the error signal is directly related to the accuracy of the predictor, hence the more accurate the predictor, the lower the generated error signal. For still video frame segments the corresponding pixels in the previous frame are used as the predicted value, while for moving segments or objects the pixels in the previous frame displaced by the motion vectors are used for prediction.

2. Motion Estimation

   In standard video codecs typically each (16x16)-pixel Macro-Block (MB) has its own motion vector. Furthermore, only the luminance component of the image is used for motion estimation and the corresponding scaled version of the motion vector is used for the chrominance component.

3. Inter-/Intra-Frame Coding Switch

   The inter-/intra-frame coding switch of Figure 1.13 is used to code the MB either in inter- or intra-frame mode. The decision about the inter-/ ntra-coding of the MBs specifically depends upon the coding technique applied. More details about the H.264 codec's specific inter/intra-coding algorithms are provided in the following chapter.

4. Discrete Cosine Transform block

   The DCT [63] of Figure 1.13 is used to transform the video from the pixel/spatial domain to frequency/transform domain. In Y:U:V 4:2:0 format each MB consists of four (8x8) luminance pixel blocks and two (8x8) chrominance pixel blocks, which are transformed to blocks of 64 luminance coefficients and two blocks of 64 chrominance coefficients.

5. Quantisation Block

   Again, the transformation of the video to the frequency domain does not actually result in any compression. It basically transforms the 64 8-bit/pixel values to 64 DCT coefficient values and due to the property of orthonormality the energy in both the original and transformed domains remains the same. Compression is basically achieved with the aid of the process of quantisation and VLC, as seen in Figure1.13. However, the advantage of the DCT is that most of the image energy is concentrated in a few low-frequency DCT coefficients, while the remaining coefficients have a low energy, which can be converted to zero by the quantiser. Additionally, since the human vision system is less sensitive to high-frequency coefficients, a higher compression can be achieved by applying coarser quantisation to the high-frequency coefficients relative to the low-frequency coefficients.

6. Variable Length Coding

   A variable length code is a lossless data compression technique which maps source symbols to a variable number of bits. After quantisation the DCT coefficients are variable length coded and transmitted to the receiver. Additionally, the motion vectors and the MB addresses are also VLC coded before transmission.

1.8.5.3 Decoder

The basic function of the decoder seen in Figure 1.14 is to perform a series of inverse operations with respect to those performed at the encoder side, in order to transform the received bit stream back to a form comparable to the original-input stream.

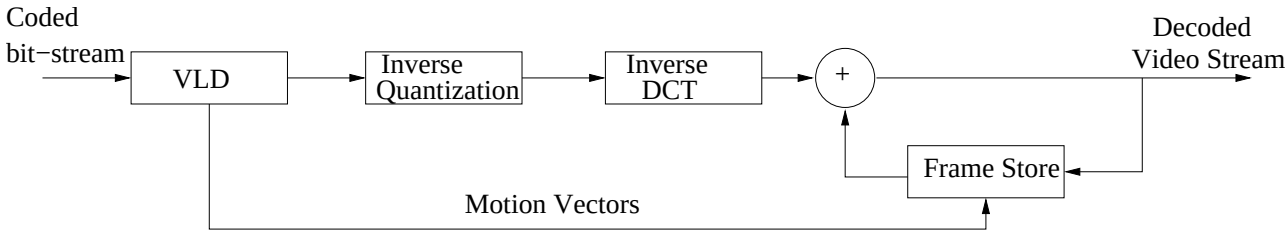

Figure 1.14: Generic inter-frame video decoder.

The different building blocks of decoder are as follows.

Inverse VLC, Inverse Quantiser and Inverse DCT At the decoder side seen in Figure 1.14 the received data stream is first Variable Length Decoded (VLD), and then inverse quantised, followed by the inverse DCT, in order to regenerate the MCER for inter-frame coding. These prediction errors are then added to the previous

motion-compensated picture values in order to generate the decoded output picture. A copy of this decoded picture is then stored in the frame-store seen in Figure 1.14 for prediction of the next frame for decoding.

1.8.5.4 Post-Processor

The various video coding algorithms may result in visually annoying artifacts at low bit rates, which degrades the perceptual quality of video. In this scenario post-processing algorithms provide an attractive solution for maintaining an acceptable perceptual quality of the decoded video sequence. The basic building blocks of the post-processor are shown in Figure 1.15.

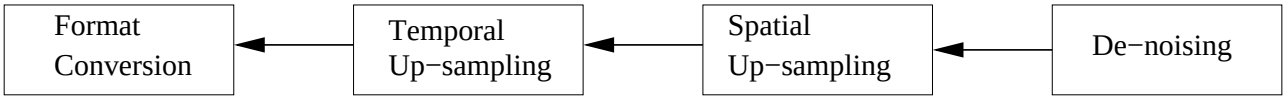

Figure 1.15: Design blocks of Post-processor.

De-noising is used to remove the various types of artifacts, such as blocking artifacts and mosquito-noise from the decoded image. For example, filtering can be used for the removal of these artifacts, which are typically caused by the error introduced by the noise in the communication channel. For spatial up-sampling various methods, such as bi-linear interpolation [39], can be used to generate the interpolated pixels.

Similarly, temporal interpolation of video sequences is used, whenever there is a need for temporal up-sampling. Up-sampling of video sequences is required in a variety of applications, such as low bit rate video coding, when video decompression is performed and also when frame-rate up-conversion is required for improved display. For temporal up-sampling new frames can be generated by simple interpolation. For example, for the 'Mom' sequence of Figure 1.11 the intermediate frame obtained by temporal up-sampling after taking the simple average of two frames is shown in Figure 1.16. This averaging operation reduces the 'crispness' of the image. Note however the loss of sharpness in the image in Figure 1.16, which is most visible around the eyes.

However, simple interpolation methods, such as frame repetition, averaging and bi-linear interpolation employed for the reconstruction of skipped frames in a video sequence may introduce undesirable artifacts. Similarly, the common technique of Motion Compensated Temporal Interpolation (MCTI) [64] may be employed to interpolate video frames, but blocking artifacts may appear at some places in the output video. Therefore, techniques such as Unidirectional Motion Compensated Temporal Interpolation (UMCTI) [65] and Overlapped Motion Compensation Temporal Interpolation (OMCTI) [42] have been proposed to remove this blocky effect from the interpolated frames. These concepts are described below in more detail.

Motion Compensated Temporal Interpolation

In the early stages of video-compression research, temporal interpolation was based on frame repetition or linear interpolation. However, when this was applied to two consecutive original frames, it resulted in a number of

Figure 1.16: The interpolated frame generated by averaging between two upsampled frames.

artifacts [66]. Using an improved MCTI was deemed to be crucial for improving the video quality [27, 67], which involves generating and inserting a frame between the kth and (k-1)st frame. Accordingly, all frames are divided into (NxN)-pixel blocks. The motion vectors are then found for all the blocks by using both forward- and backward-prediction aided block based motion estimation. In this way multiple motion vectors may be generated for each block and the final motion vector is deemed to be the one resulting in the lowest MCER.

There are drawbacks and some constraints associated with this technique. For example, blockiness may appear in the up-sampled video clip using MCTI. However, the resultant interpolated image still has a better perceptual quality relative to the interpolated frame generated using a simple averaging method. Additionally, the blockiness in the MCTI frame can be removed using spatial filtering. As an example, Figure 1.20 presents the average of the interpolated frame between the first two frames of the Foreman video sequence, which are shown in Figure 1.18 and Figure 1.19. Additionally, the interpolated frame generated using MCTI is shown in Figure 1.21. It can be observed from the generated results that the MCTI frame of Figure 1.21 has a better perceptual quality relative to the average interpolated frame of Figure 1.20. Additionally, the blockiness in the MCTI frame of Figure 1.21 is removed using spatial filtering, as shown in Figure 1.22. Furthermore, in order to more clearly visualise the capability of interpolation methods, we consider frame interpolation between the first and fifth frame of the Foreman video sequence, where we clearly have a more substantial spatial difference as shown in Figure 1.23 and Figure 1.24. The MCTI interpolated frame of Figure 1.26 has a better quality relative to the average interpolated frame of Figure 1.25. Furthermore, the filtered version of Figure 1.26 is presented in Figure 1.27.

MCTI is also a computationally intensive task, which requires a large memory, since both the forward- and backward-oriented motion estimates have to be computed. In order to circumvent these drawbacks and constraints, Unidirectional Motion Compensated Temporal Interpolation (UMCTI) was proposed in [65].

Unidirectional Motion Compensated Temporal Interpolation

According to this method, first the k*th*, (k-1)*st* and the inserted frame are divided into perfectly tiling blocks of size (NxN)-pixels. Only forward oriented motion estimation is employed for determining the motion vector for each block, instead of carrying it out bi-directionally. Its advantage is a reduced amount of memory and a reduced computational complexity. The Mean Absolute Difference (MAD) is an often-used criterion for searching for the best match of a block in the (k-1)*st* frame to a block in the k*th* frame within the search area of (2x+1)x(2x+1)-pixels. Exhaustive search may be used in order to find the best match. Let us consider Figure 1.17, where we carried out an exhaustive search in the (k-1)*st* frame, and found that B_2 is the best match for the block B_1 in the k*th* frame, which is located at the position of $(x+d_x,y+d_y)$ and the corresponding motion vector is (d_x,d_y). Assuming linear motion translation, block B_3 is a candidate along with other blocks in the inserted frame and the motion vector for B_3 is $(d_x/2,d_y/2)$. The block B_3 is not considered in the inserted frame, since other interpolated blocks may appear over the same location or with some overlap with B_3. Consider B_4 as an example, which overlaps with the previously determined block B_3. The technique followed here is that the new block B_4 is considered as having two parts. Let the one, which overlaps with the previously determined block be B_4^o and the one which does not overlaps be B_4^{no}. The block B_4^o is determined by calculating the average between the corresponding overlapping portions of blocks B_1 and B_2, whereas B_4^{no} is simply unidirectionally predicted. For any location in the inserted frame, B_4^o and B_4^{no} are calculated as:

$f_{insert}(i,j) = [f_{k-1}(i+d_x/2, j+d_y/2)+ f_k(i-d_x/2,j-d_y/2)]/2$ for B_4^o

$f_{insert}(i,j) = f_{k-1}(i+d_x/2, j+d_y/2)$ for B_4^{no}.

The advantage of this technique is the reduced amount of calculations and its reduced memory requirement.

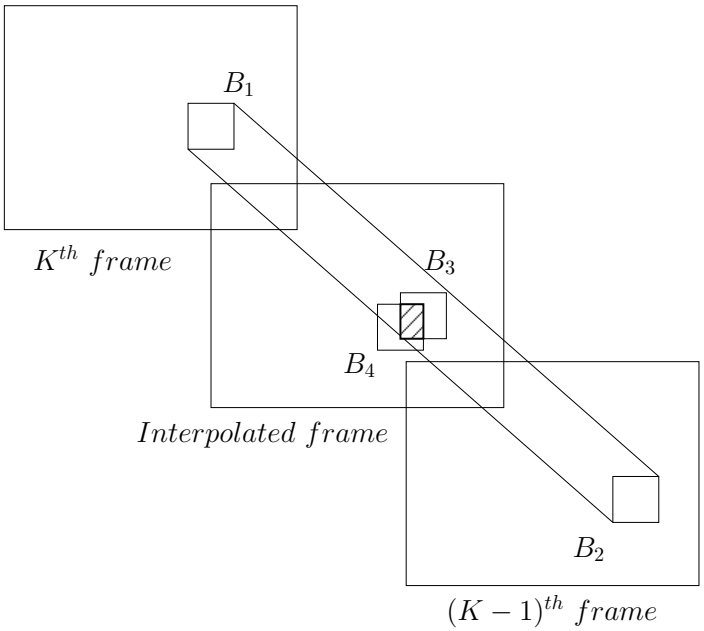

Figure 1.17: Unidirectional Motion Compensated Temporal Interpolation (UMCTI).

Fast Unidirectional Motion Compensated Temporal Interpolation

The novel technique of Fast Unidirectional Motion Compensation Temporal Interpolation (FUMCTI) was proposed in [64]. The only difference with respect to the UMCTI technique is that instead of carrying out an exhaustive full search, the MAD is first calculated for assuming a zero motion vector. If the value of the MAD is below some predefined threshold, it is considered to be a stationary block; otherwise full search is carried out across the entire search area.

UMCTI and FUMCTI with OVERLAPPING (UMCTIO and FUMCTIO):

Similarly to the MCTI and UMCTI techniques, FUMCTI may also exhibit some artifacts. Although the complexity may have been substantially reduced, blockiness may appear in the up-sampled video. In order to reduce this blockiness, an overlapping block-based algorithm has been proposed in [42] for MCTI, UMCTI and FUMCTI. In this algorithm, the motion vectors of the neighboring blocks are also considered along with the motion vector of the current block. The weighted average of these blocks is then determined, in order to reduce the blockiness.

Figure 1.18: First frame of Foreman video sequence.

Figure 1.19: Second frame of Foreman video sequence.

Figure 1.20: Interpolated frame between first two frames using averaging.

Figure 1.21: Interpolated frame between first two frames using MCTFI.

Figure 1.22: Interpolated frame between first two frames using MCTFI with (3×3) averaging.

Figure 1.23: First frame of Foreman video sequence.

Figure 1.24: Fifth frame of Foreman video sequence.

Figure 1.25: Interpolated frame between first and fifth frame using averaging.

Figure 1.26: Interpolated frame between first and fifth frame using MCTFI.

Figure 1.27: Interpolated frame between first and fifth frame using MCTFI with (3×3) averaging.

1.9 Video Compression Standards

Diverse video compression standards have been ratified by three different standardisation organisations, namely by the [68]

1. International Telecommunications Union (ITU, CCITT)

   Telecommunications Section (ITU-T);

2. International Standarization Organisation (ISO)

   Motion Picture Experts Group (MPEG);

3. Joint Photographic Experts Group (JPEG)

   (collaboration of ITU-T and ISO).

The video standardisation activities are presented pictorally in Figure 1.28. Additional details about the video standardisation efforts are provided in Table 1.1.

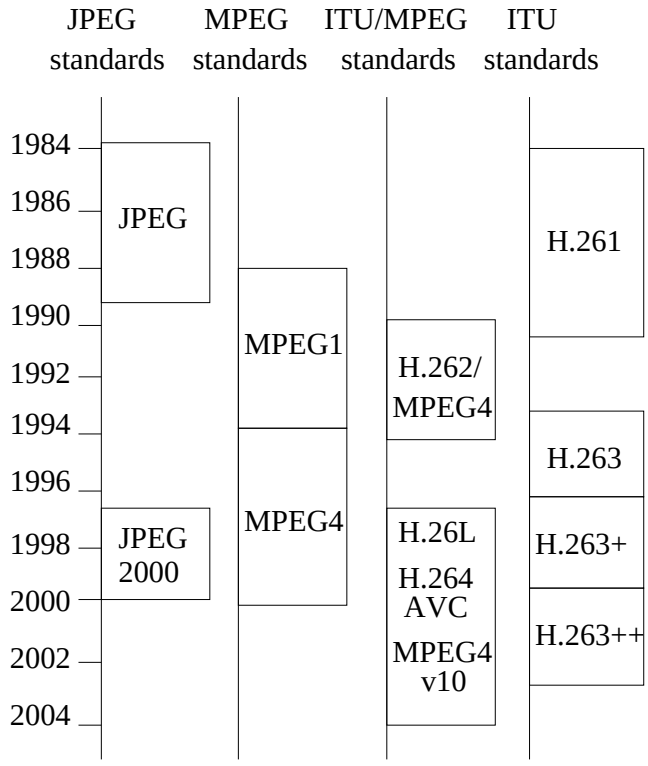

Figure 1.28: Video Standardisation Activities.

1.9.1 Outline of Thesis

Figure 1.29 gives an overview of the thesis. Chapter 1 commences with a rudimentary introduction to the different commonly used concepts of video coding. Section 1.2 outlines the commonly used analogue video formats. The analogue to digital conversion procedure is outlined in Section 1.3. The concept of colour spaces used to represent the video signal was discussed in Section 1.4, followed by the commonly used video sampling patterns in Section 1.5. The specifications of the various image formats obtained using a digitisation process were described in Section 1.6. Furthermore, a typical sequence of operations used for the removal of redundancy from the video sequence was described in Section 1.8. Finally, the different video compression standards were summarised in Section 1.9.

Chapter 2 provides details about the H.264/AVC video coding standard, while Chapters 3 to 5 will then provide the details of our beneficial transceiver structures employing novel channel coding schemes for enhancing the achievable error-resilience of diverse video transmission systems. Each chapter constitutes an evolutionary improvement of the previous chapter.

To elaborate a little further, Chapter 2 deals with the H.264 video coding standard, which describes an efficient and network-friendly video codec capable of supporting a variety of multimedia applications in Section 2.1. The evolution of the base-line codec to the H.264 AVC scheme is outlined in Section 2.2. The fundamental building blocks of video compression first detailed in Section 1.8 are systematically shown revisited in Figure 2.2 of Section 2.3 in order to explain the H.264 hybrid video coding concept, followed by the outline of its main video coding features in Section 2.4. Section 2.5 provided details about the different video coding techniques used by H.264/AVC for enhancing the video coding efficiency. The concept of block-based inter-frame coding was detailed in Section 2.5.2. Furthermore, the block-based transform coding concept was discussed in Section 2.5.3. Moreover, the concept of deblocking filters was reviewed in Section 2.5.4. The two alternative entropy coding methods used in the H.264 standard were explained in Section 2.6. Section 2.7 described rate-distortion based optimisation of the H.264 codec. Finally, Section 2.8 provided details about the H.264/AVC video coding profiles and levels.

Chapter 3 commences with a rudimentary introduction to the different techniques used in literature for the efficient transmission of video, followed by the concepts behind iterative detection schemes in Section 3.2. In Section 3.3, we introduced the concept of binary Extrinsic Information Transfer (EXIT) chart analysis. The procedure used to calculate the EXIT characteristics of the inner and outer decoder is briefly summarised in Section 3.3.1 and Section 3.3.2. The concept of different component codes used in the iterative detection schemes is described in Section 3.4, followed by an overview of the Recursive Systematic Convolutional (RSC) Codes in Section 3.4.1. The detailed description of irregular coding and its various features used in digital multimedia communications is given in Section 3.4.2 and Section 3.4.3. The video source signal coded by the H.264/AVC standard video codec of Section 2.3 and provided as input to our proposed systems is presented in Section 3.5. The detailed hierarchical structure of the H.264 coded video codec is presented in Section 3.6. The bit-stream syntax of the H.264/AVC coded video is detailed in Section 3.7. In Section 3.8 we describe the H.264 data-partitioning concept. Furthermore, in Section 3.9 we presented our H.264 error sensitivity study based on the objective video quality evaluation criterion detailed in Section 1.7.2. In Section 3.10 we

investigated an Unequal Error Protection (UEP) aided video system using RSC codes, which relied on the architecture presented in Figure 3.23. A novel iterative detection aided H.264 wireless video telephony scheme using IrRegular Convolutional Codes (IRCC) was proposed in Section 3.11. In contrast to Section 3.10, where different component codes were utilised to provide UEP for the different portions of the bit-stream, the innate UEP capability of IRCCs was exploited in Section 3.11 to provide UEP for the coded video-stream. Finally, the performance of the proposed system was analysed using EXIT charts in Section 3.11.4 and our results were discussed in Section 3.11.5.

In Section 3.11 we found that iterative detection schemes result in beneficial BER reductions and objective video quality improvements of the coded video sequence, therefore in Chapter 4 we focused our attention on finding the most appropriate component codes for iterative detection. Chapter 4 commenced with an overview of the transmitter and receiver in Section 4.2 and Section 4.3. In Section 4.4 a design example was provided in order to quantify the iterative source-channel decoder's achievable performance improvements. Additionally, the concept of Short Block Codes (SBCs) was introduced in Section 4.5. In Section 4.5.5 we observed from the EXIT-chart analysis procedure of Section 3.3 that the convergence behaviour of Iterative Source-Channel Decoding (ISCD) is substantially improved with the aid of SBCs. The beneficial effects of SBCs combined with rate-1 inner codes on the achievable performance was presented in Section 4.6. From the EXIT analysis of Section 4.6.3 we concluded that the performance of the iterative source-channel codec improved as a result of increasing the minimum Hamming distance $d_{H,min}$ of the SBCs employed. Additionally, the concept of Redundant Source Mapping (RSM) was described in Section 4.7 along with its EXIT-chart analysis and the associated performance results. We observed from the EXIT curves portrayed in Section 4.7.3 that the convergence behaviour of the SBCs depends on identifying their most appropriate code-rate.

In Chapter 4 we considered various iterative decoding strategies. As the transceiver considered in Chapter 4 was composed of a classic Single-Input, Single-Output (SISO) scheme, in Chapter 5 our discussions evolved further and we focused our attention on finding the most appropriate diversity aided Multiple-Input Multiple-Output (MIMO) scheme with the aim of designing more sophisticated near-capacity video transmission systems. In Section 5.2 we evaluated the performance of the data-partitioned H.264 coded video transmission system of Figure 5.3 using different RSC based UEP schemes, while employing a SP modulation aided Differential Space Time Spreading (DSTS) transmission scheme. The UEP scheme considered is similar to that of Sections 3.10 and 4.4, but instead of a SISO transceiver we considered SP modulation aided DSTS based MIMO transmission. When using UEP, the perceptually more important bits were provided with more strong protection relative to less important video bits. Additionally, in Section 5.3 an Unequal Source-Symbol Probability Aided (USSPA) design was incorporated, which improved the error correction capability and hence enhanced the subjective video quality by exploiting the residual redundancy that remains in the coded stream after encoding. The resultant USSPA coded bit-stream was transmitted using a DSTS aided SP modulation scheme for attaining a diversity gain without the need for any high-complexity MIMO channel estimation. In Section 5.4 an iterative detection aided combination of a RSC and a SBC was used to improve the overall BER performance, which enhanced the objective video quality expressed in terms of PSNR. Like any classic FEC codes, the SBCs of Section 5.4 introduced intentional redundancy in the source coded bit-stream, which assisted the outer decoder's EXIT curve in reaching the $(1,1)$ point of perfect convergence to a vanishingly

low BER. The effects of the different error protection schemes of Table 5.10(a) on the attainable system performance was demonstrated in Figure 5.14, while keeping the overall bit-rate budget constant for the transmission of Data-Partitioned (DP) H.264 source coded video over correlated narrowband Rayleigh fading channels. Additionally, in Section 5.5 EXIT charts were utilised in ordered to analyse the effect of the SBC coding rate on the achievable performance of the UEP iterative JSCD strategies of Table 5.14(a). Furthermore, in Section 5.6.2 we considered jointly optimised three-stage source and channel decoding arrangements employing serially concatenated and iteratively decoded SBCs combined with a URC and multi-dimensional SP modulation, in order to circumvent the BER floor imposed by the conventional two-stage turbo-detection schemes of Section 4.6. The resultant coded signal was transmitted using a non-coherently detected DSTS MIMO-aided transceiver designed for near capacity JSCD, as shown in Figure 5.34. The performance of the system was evaluated in Figure 5.34 by considering interactive video telephony using the H.264/AVC source codec. The output bit-stream generated by the state-of-the-art H.264/AVC video codec typically contains limited natural residual redundancy. Therefore, to improve the error-resilience of the ISCD scheme of Figure 5.34, SBCs were incorporated in order to impose additional FEC redundancy on the source coded bit-stream. The natural residual redundancy of source coding and the artificial redundancy imposed by SBCs was iteratively exploited in the architecture of Figure 5.34 in a turbo process to improve the overall BER, as characterised in Figure 5.37. The associated objective video quality performance improvements were discussed in the context of Figure 5.38 in terms of the PSNR. In Section 5.6.6 the convergence behaviour of the MIMO transceiver advocated was investigated with the aid of EXIT charts.

Finally, the main conclusions of the thesis and a range of future research ideas were detailed in Chapter 6.

1.9.2 Novel Contributions

This dissertation is based on the following publications and manuscript submissions [2–11], where the main novel contributions can be summarised as follows:

- In Section 3.11 a serially concatenated and iteratively decoded EXIT chart-optimised IRCC is amalgamated with a unity-rate precoded DP H.264 coded video transmission system. When using UEP, the perceptually more important bits are provided with more strong protection relative to the less important bits. An iterative detection aided combination of IRCCs and a rate-1 precoder was used to improve the overall BER performance and to enhance the objective video quality expressed in terms of PSNR. The effect of different error protection schemes on the attainable system performance is demonstrated, while keeping the overall bit-rate budget constant for the transmission of DP H.264 source coded video over correlated narrowband Rayleigh fading channels. More specifically, we exploited the high design flexibility of IRCCs, which constitute a family of different-rate subcodes, while maintaining an excellent iterative decoding convergence performance. Additionally, due to the employment of different-rate subcodes, IRCCs have the capability of providing UEP for the H.264 coded video stream. An EXIT chart matching procedure was used for the design of our specific IRCC [7].

- From the systematic study of Section 3.11 we found that UEP aided iterative detection schemes results in BER reduction and beneficial objective quality improvement of the coded video stream. Hence, in

order to achieve further performance gain, we focused our attention on finding most appropriate component codes for iterative detection process. Therefore, in Section 4.5 a powerful, yet low-complexity algorithms for EXIT-chart Optimised Short Block Codes (EOSBC) are presented, which can be used to generate EOSBCs for a variety of code rates associated with diverse $d_{H,min}$ values that are applicable to wide-ranging multimedia services [2]. Additionally, instead of modelling the sources with the aid of their correlation, the practically achievable interactive video performance trends are quantified when using state-of-the-art video coding techniques, such as H.264/AVC. More explicitly, instead of assuming a specific source-correlation model, we based our system design examples on the simulation of the actual H.264/AVC source coded bit-stream. The EOSBC-aided schemes are utilised for protecting the H.264 coded bit-stream using RSC codes [69]. The EOSBC coding scheme is incorporated by carefully partitioning the total available bit-rate budget between the source and channel codecs, which results in an improved performance [3].

- Ashikhmin *et al* proved in [70, 71] that the inner code of a serially concatenated turbo scheme must be of rate = 1 to be capacity achieving. Therefore, in contrast to Section 4.5 composed of diverse combination of different rate EOSBCs and RSCs for UEP of video stream, in Section 4.6 the performance of an iterative detection aided combination of EOSBC assisted source decoding and a rate-1 precoder designed for the transmission of data-partitioned H.264 source coded video over correlated narrowband Rayleigh fading channels is analysed. It is demonstrated the effects of different EOSBCs having diverse minimum Hamming distances [$d_{H,min}$] but identical coding rates on both the overall BER performance as well as on the objective video quality expressed in terms of the PSNR [5].

- In Section 4.6 we found that the performance of EOSBC improves upon increase in $d_{H,min}$ and result in open EXIT tunnel at lower $\mathrm{E}_b/\mathrm{N}_0$ value relative to identical rate EOSBC with lower $d_{H,min}$. Therefore, in order to improve the performance of outer code in iterative detection process, we focused our attention on finding outer codes with further improved $d_{H,min}$ and hence decoding performance. In Section 4.7, we proposed the joint optimisation of iterative joint source and channel decoding with the aid of our proposed EXIT chart optimised RSM scheme, which is designed for guaranteed convergence to achieve an infinitesimally low Bit Error Ratio (BER). The performance of our system was evaluated, while considering an iterative combination of RSM assisted source decoding and RSC for transmission over correlated narrowband Rayleigh fading channels. EXIT charts were utilised to analyse the effect of redundancy using different RSM schemes on the attainable system performance, while keeping the overall bit-rate budget constant [9].

- In Section 3.11 we found that the UEP scheme outperforms the benchmarker EEP scheme. Additionally, we utilised different iterative detection schemes in Section 4.5, 4.5 and 4.5, while employing classic SISO transceiver design. Therefore, in Section 4.4 we focused our attention on finding an appropriate UEP scheme using RSC codes, employing the most appropriate diversity or MIMO scheme. A novel USSPA source and channel decoding arrangement is proposed for the transmission of an H.264/AVC coded video bit-stream. This USSPA coding scheme is utilised for the UEP of the H.264/AVC coded bit-stream using RSC codes, while exploiting the different relative importance of the bits in the H.264/AVC coded bit-stream. The resultant USSPA assisted UEP aided bit-stream is transmitted using a DSTS aided

Sphere Packing (SP) modulation scheme [72] for attaining a diversity gain without the need for any high-complexity MIMO channel estimation. Additionally, we incorporated a novel EOSBC scheme by partitioning the total available bit-rate budget between the source and channel codecs, which results in an improved performance relative to the identical-rate benchmarker scheme dispensing with EOSBC, when ISCD is employed [4].

- In Section 4.6 we found that the the iterative detection scheme results in the beneficial BER performance improvement of the video transmission system. Similarly, the SBC assisted RSC coded video transmission scheme is analysed in Section 5.4, which resulted in subjective video quality improvement of the transmitted video sequence, while employing sophisticated SP modulation aided DSTS transmission system. However the conventional two-stage iterative detection schemes suffer from BER floor problem. In order to circumvent the BER floor imposed by conventional two-stage turbo-detection schemes, we focused our attention on finding the near-capacity three-stage iterative detection scheme. Therefore, in Section 5.6.2, we considered jointly optimised three-stage iterative joint source and channel decoding, while employing a novel combination of serially concatenated and iteratively decoded EOSBCs combined with a Unity Rate Code (URC) and multi-dimensional SP modulation. The resultant coded signal is transmitted over non-coherently detected MIMO DSTS designed for near capacity Joint Source-Channel Decoding (JSCD). The performance of the system was evaluated by considering interactive video telephony using the H.264/AVC source codec. The natural residual redundancy after source coding and the artificial redundancy due to EOSBC coding is iteratively exploited in a turbo process to improve both the overall BER and the objective video quality performance quantified in terms of the PSNR. The convergence behaviour of the advocated MIMO transceiver is investigated with the aid of bit-based EXIT charts [8].

1.10 Chapter Conclusions

This chapter has the following findings,

1. The video [73] signal is generated by the video camera in the analogue format described in Section 1.2 by scanning a two-dimensional scene and each video clip is comprised of a number of scanned scenes referred to as a video frame. The number of lines per frame has a direct relation to its bandwidth and resolution. Additionally, the number of frames per second is another factor affecting the temporal resolution of the video, which should always be kept above some specified value to avoid video flicker.

2. In order to facilitate video storage, compression, and error correction, the analogue video is converted to a digital format, using a series of steps consisting of filtering, sampling and quantisation, as described in Section 1.3.

3. From the subjective and objective video quality evaluation methods presented in Section 1.7 it may be concluded that the PSNR evaluation is unable to characterise the human interpretation of distortion, which is perceived by human vision system differently at different parts of the image. For example a

small block of severely distorted pixels in the subjectively most eye-catching areas of the frame, such as the human face attracts the viewer's attention and hence it is perceived as large distortion in the frame, although it may hardly affect the objective quality metric.

4. The video compression process consists of three important coding components including intra-frame coding, inter-frame coding and entropy coding as described in Section 1.8.

5. The concept of entropy coding detailed in Section 1.8.3 is used to achieve further compression by reducing the redundancy among the symbols, using various VLC techniques. The idea of VLC is to assign lower number of bits to high-probability values, while long code words are assigned to symbols with low probability.

6. The concept of MCTI is presented in Section 1.8.5.4 and from the results obtained it is observed that despite the high compression efficiency achieved by the employment of MCTI, hardly any perceptual difference is noticed among the interpolated frames of Figure 1.17 with reference to the original frame presented in Figure 1.11.

1.11 Chapter Summary

In this chapter we provided a basic introduction to the different commonly used concepts of video coding. Initially, the video was generated in its analogue format at the output of the video camera by scanning a two-dimensional scene, as described in Section 1.2. In order to benefit from video storage and compression, the analogue video was converted into a digital format as described in Section 1.3, using a sequence of operations, such as filtering, sampling and quantisation. The concept of colour spaces used to represent the video signal was highlighted in Section 1.4, followed by the commonly used video sampling patterns in Section 1.5, used to specify the relative number of luminance (Y) and chrominance (C_b and C_r) pixels per video frame. Then we have detailed the various image formats obtained using a digitisation process, such as the SIF, the CIF and the Quarter SIF and Quarter CIF etc in Section 1.6. In Section 1.7, the different video quality evaluation methods used in the litrature were presented. Furthermore, a sequence of operations used for the removal of redundancy from the video sequence was described in Section 1.8, which included motion compensation, intra-frame coding, inter-frame coding, transform coding, quantisation and entropy coding. The information related to the building blocks of a video coding system was detailed in Section 1.8.5. Further details about the different video compression standards was provided in Section 1.9. The outline of the thesis was presented in Section 1.9.1, while its novel contributions were summarised in Section 1.9.2. Finally, the chapter is concluded in Section 1.10.

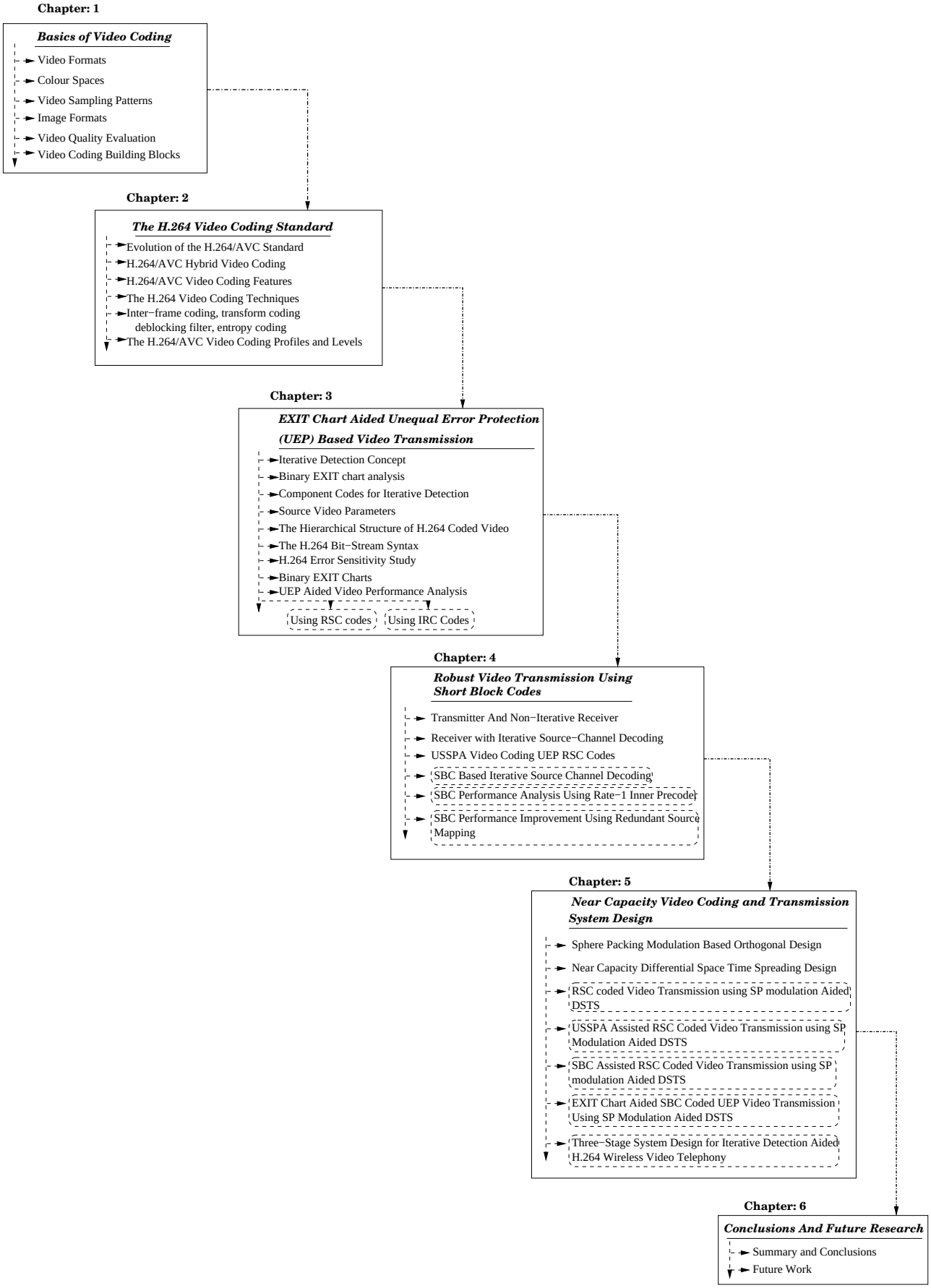

Figure 1.29: Thesis Outline.

Chapter 2

The H.264 Video Coding Standard

2.1 Introduction

The combined efforts of the ITU-T Video Coding Experts Group (VCEG) and ISO/IEC Motion Picture Experts Group (MPEG) resulted in the H.264/AVC video coding standard [74]. The main aim of this standardisation activity was to design an efficient and network-friendly video codec capable of supporting a variety of applications, including both real-time interactive applications such as video conferencing, video telephony and non-real-time applications, such as video streaming and digital television broadcast. Considerable research efforts have been dedicated to the design of H.264 video codec [75–78]. H.264/AVC provides the best performance in terms of its rate-distortion efficiency amongst the existing standards [23, 79].

The main tasks involved in modern video communication systems are summarised in Figure 2.1. The video scene is captured by a video capturing device, such as a camera and additional preprocessing may be applied such as format conversion and video quality enhancement. The video encoder then encodes the frames and converts them into a bit stream, which is then transmitted over the channel. The decoder decodes the received video which is displayed on the screen after optional post-processing, such as filtering, format conversion, error concealment and video enhancement [74].

2.2 Evolution of the H.264/AVC Standard

In 1980s ITU-T recommended the employment of combined interframe DPCM and Discrete Cosine Transform (DCT) for a codec suitable for videoconferencing type of applications. This recommendation showed reasonable improvements over existing video codecs, and was capable of providing a reasonable quality for video conferencing applications at 384 Kbits/s coding rate and high-quality video at 1 Mbit/s. This standard was officially completed in 1989 and was referred to as the H.261 standard. In the 1990s the MPEG initiated a project to investigate different video coding techniques for the storage of video on various digital storage media, such as CD-ROMs with a quality comparable to that of Video Cassette Recorders (VCRs). For this project the

basic coding structure of H.261 was used as a reference. This project was referred to as MPEG-1. This standard fulfilled the industrial needs to store video data on storage media other than conventional analogue VCR. The MPEG-1 standard is based on progressive [80] scanning and therefore can't decode interlaced [80] video, because interlaced video has to be de-interlaced before decoding. The decoded image may then be converted back to the interlaced format for displaying. The MPEG-1 and H.261 standards became successful but there was a need for an advanced codec for a wide variety of applications. Considering the similarities between H.261 and MPEG-1, ITUT and ISO agreed to make a concerted effort to design a generic video codec. This generic codec was finalised in 1995, and was termed as MPEG-2/ H.262, more commonly known as MPEG-2. The MPEG-2 standard is capable of with supporting interlaced video coding and is widely used for transmission of Standard Definition (SD) and High Definition (HD) TV signals over a variety of medias' including satellite, cable terrestrial channel as well as of storage of high-quality video signals on Digital Versatile Discs (DVDs). However, other transmission media, including cable modems, Digital Subscriber Loops (xDSL) and the Universal Mobile Telecommunications System (UMTS) offer lower data rates in comparison to broadcast channels. Therefore, the growing popularity of High Definition Television (HDTV) created a need for higher video coding efficiency, in order to provide higher quality video representation for transmission over these digital transmission media [23].

Hence, video coding research evolved through the development of the ITU-T H.261, H.262 (MPEG-2), and H.263 video coding standards along with the enhancement of H.263, leading to H.263+ and H.263++. Throughout this development continuous efforts were made to improve the achievable coding efficiency and robustness for transmission over diverse network types [31].

In heterogeneous state-of-the-art communication environments apart from the achievable coding efficiency the adaptability of the encoded video rate and quality is also of high importance in packet-based best effort Internet streaming and in wireless networks supporting diverse mobile receivers. The video adaptation procedures proposed for different transport networks are typically defined in separate standards, such as H.320 and H.324. However the best-possible performance of the video communication system can only be ensured by the innate integration of the video coding and network adaptation layers.

In early 1998 the VCEG issued a call for proposals in the context of a project referred to as H.26L. Its primary goal was to achieve an improved coding efficiency and network friendliness [74], with the ambitions target of doubling the coding efficiency (i.e. to half the bit rate) in comparison to the existing video coding standards for a broad variety of applications. The first draft of this standard was completed in October 1999. In December 2001 VCEG and MPEG formed a Joint Video Team (JVT), with the common goal of finalising a new draft of the H.264/AVC video coding standard in March 2003, which would cover all common video applications, ranging from mobile services and video conferencing to IPTV, HDTV and HD video storage.

The design of H.264/AVC was defined in form of two layers known as the Video Coding Layer (VCL) and Network Abstraction Layer (NAL) [76]. The core compression engine of H.264/AVC is based on VCL, which consists of different sub-levels known as blocks, macro blocks and slices. It was designed to be as network-independent as possible and consists of various coding so-called encoding tools' which improve the attainable coding efficiency and error robustness of the coded video stream. The NAL formates the stream generated by VCL to a specific format suitable for various transport layers in multiplex environments. The

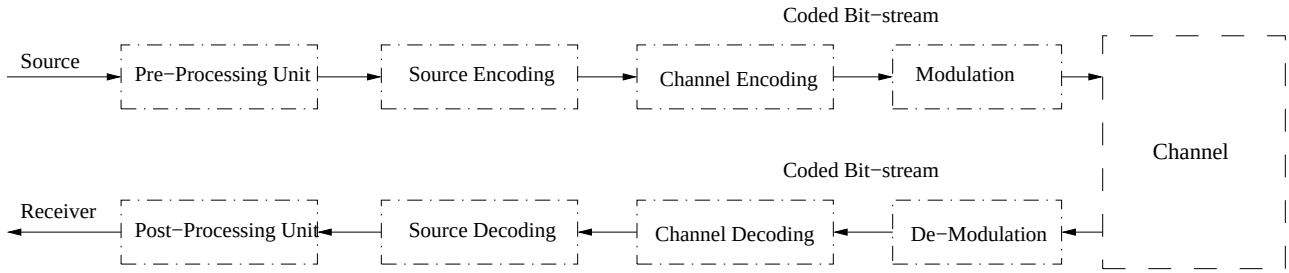

Figure 2.1: Video system broadview.

NAL's design is suitable for adaptation to various circuit switched and packet switch transport layers. For circuit switched transport layers, such as H.320, H.324M or MPEG-2 the NAL delivers the coded video as a simple structure consisting of an ordered stream of bytes containing certain start codes, for example the Picture Start Code (PSC), which allows us to resynchronise at the commandment of a new video frame as well as to interpret the structure of the bit-stream. By contrast, for the packet switched networks Transmission Control Protocol (TCP)/User Datagram Protocol (UDP)/Real Time Protocol (RTP)/Internet Protocol (IP) the NAL delivers the coded video packets [81].

Wireless systems are typically constrained owing to the availability of limited bandwidth and battery power. Therefore, the drive for increased compression efficiency has to be carefully balanced against the increased power consumption of high-complexity signal processing in video coding standards. Furthermore, the integration of the channel coded video system into different types of communication networks, while maintaining an enhanced error-resilience are equally important design aspects of wireless video and multimedia applications [82, 83]. In this context the H.264/AVC coding standard may be considered as an attractive candidate for all wireless applications, including Multimedia Messaging Services (MMS), Packet-Switched Streaming Services (PSS) and real-time conversational applications [76].

2.3 H.264/AVC Hybrid Video Coding

The standard video codecs, such as H.261, H.263, H.264/AVC, MPEG-1, 2 and 4 are based on the hybrid video coding concept, consisting of a combination of video coding techniques, such as motion compensation and transform coding. The generalised block diagram of such a video encoder is shown in Figure 2.2. Initially the video encoder divides the input image into Macro-Blocks (MBs), each comprising all three colour components known as Y, C_b and C_r. The luminance component representing the brightness information is denoted by Y, while Cb and Cr represent the chrominance or colour information. To reduce the number of bits, the chrominance components, which are less accurately resolved by the human eye, are sub-sampled both in the horizontal and vertical direction by a factor of two. This results in a 4:2:0 image format consisting of a macro-block of 16x16 pixels for the Y component and two 8x8 pixel blocks of C_b and C_r components [84].

The macro-blocks are coded either in Intra- or Inter-frame mode. In Inter-frame block coding, the block is encoded using motion compensation by utilising blocks in the previous coded frame. For each block a motion vector representing the best match of the current block in the previous frame is calculated and transmitted.

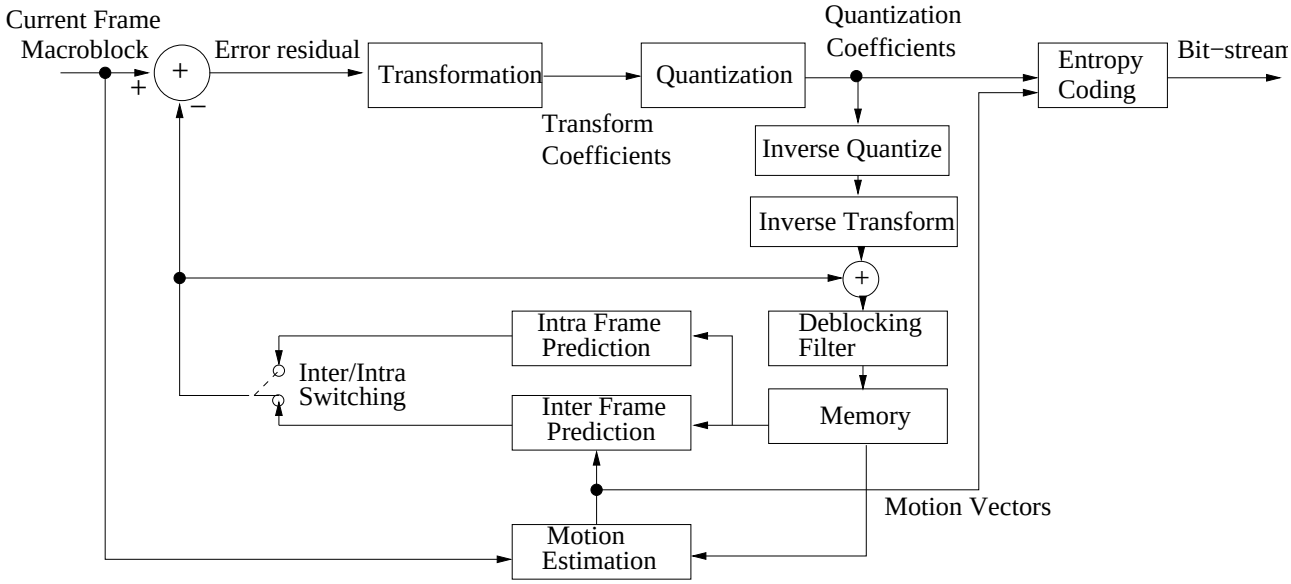

Figure 2.2: Generalised block diagram of H.264/AVC video encoder [74].

The motion prediction error, which is the difference between the original and the predicted matching block is transformed into frequency domain using the DCT. Then the DCT coefficients are quantised, entropy coded and transmitted. At the receiver the quantised (DCT) coefficients are inverse transformed and the MCER is superimposed on the predicted signal, in order to form a reconstructed MB. All the MBs are typically coded and decoded in raster-scan order [85].

The new improvements incorporated in the H.264/AVC codec relative to the previous standards are:

1. A so-called blocking filter is used in the prediction loop in order to smoothen the potentially abrupt luminance and chrominance changes near the block edges and hence to reduce the blocking artifacts. The soothened MBs are stored in memory to be used by other macro-blocks for motion prediction.

2. The H.264 codec allows the storing of multiple video frames in memory, which has the potential of improving the motion compensation accuracy at an increased complexity.

3. In the H.264 codec intra-frame prediction can also be used for improved prediction based on already encoded macro-blocks within the same image.

4. The classic floating-point representation-based DCT is replaced by the integer DCT transform [31].

In H.264 macro blocks are arranged in slices, where a slice consists of a group of macro-blocks arranged in raster scan order. Five different types of slices are supported, which are the Intra-coded (I), Predicted (P), Bidirectional (B), Switching I (SI) and Switching P (SP) slices. In the intra-coded slice all macro-blocks are intra-frame coded, while in the predictive slices all macro blocks are inter-frame coded, using Motion Compensated Prediction (MCP) based on a single reference frame. The B slice (Bi-directionally predictive-coded slice) uses bidirectional inter-frame prediction both from past and future pictures to predict the sample values of each block. Switching I and switching P slices are specific types of slices used for efficient switching between two different bit-streams [31].

H.264 supports two different coding modes for the encoding of interlaced video, namely the frame-mode and field-mode. In the frame-mode the two fields of a frame are encoded as a single progressive frame, while in the field-mode the two fields of a frame are coded separately. The choice of frame/field-mode depends upon the relative motion in the video scene. For a slow-motion video clip used in head-and-shoulders video telephony the frame-mode achieves efficient encoding by exploiting the statistical dependency between the adjacent lines, while the field-mode is preferred for a video clip associated with high motion activity in the scene. In case of high-motion clips the statistical dependency between the adjacent lines is limited and hence it is more efficient to encode the two fields separately [86].

2.4 H.264/AVC Video Coding Features

2.4.1 Network Adaptation Capability

The success of any video codec is dependent on striking an attractive compromise amongst the various design factors seen in Figure 2.3. Two important factors are the compression efficiency and the easy integration of the coded video stream into the network architecture. In H.264 these two factors are taken into account by the so-called VCL, specifying the representation of the coded video and the NAL which provides the interface between the video codec and the outside world. The VCL strikes the required compromise between compression ratio and error resilience, which are essential features required for supporting real time video services, such as video conferencing, multicasting and streaming applications. These requirements are satisfied by the appropriate choice of the encoding, transmission and decoding procedures [85].

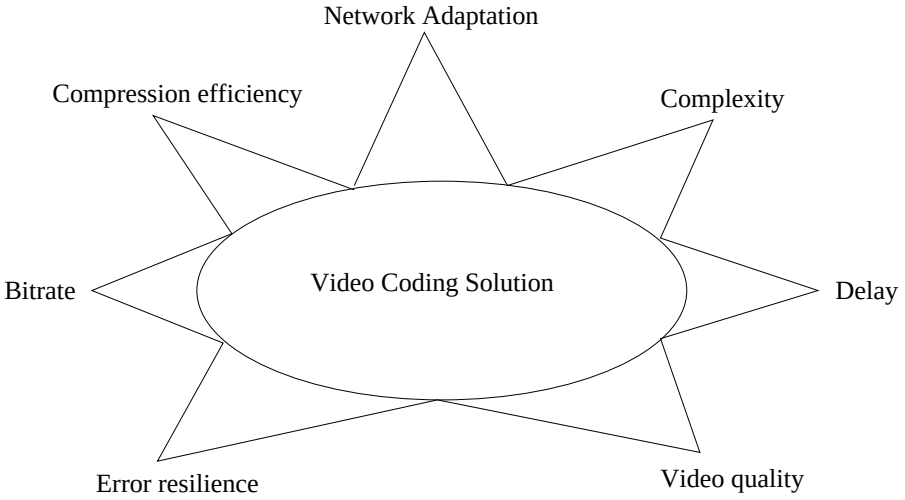

Figure 2.3: Design requirements of video communications system [1].

2.4.2 Flexible Bit Stream Structure

Variable Bit Rate (VBR) channels, such as the internet and wireless links are subjected to packet loss events and hence require channel-adaptive streaming. Channel-adaptive packet scheduling designed for transmission over these channels allows us to react to diverse network conditions, while transmitting encoded video. These capabilities are supported in H.264 with the aid of various techniques, such as dropping of 'non-reference' frames, which results in the so-called temporal scalability feature. The multiple reference frame concept along with the concept of generalised B frames provides a high flexibility in terms of temporal scalability and facilitates the rate-control of the bit stream generated. The switching between two bit streams using SI and SP types pictures provides additional flexibility and error resilience [87].

2.4.3 Error Resilience

Conversational-style real-time interactive applications have stringent delay requirements, which generally imposes additional challenges, because transmission errors due to congestion and link-layer imperfection are difficult to avoid and may lead to grave error propagation. Therefore, error resilient features have to be incorporated in the video coding standards [74]. Some of the error resilient features of H.264/AVC are briefly described below;

1. Flexible Macro-block Ordering :
   Flexible Macro-block Ordering (FMO) allows for the allocation of macro-blocks to slices using macro-block allocation maps, where a slice group may itself be composed of several slices. Therefore, when using FMOs, macro-blocks can be discrete cosine transformed and transmitted in an order out of the natural raster scan sequence. Various macro-block allocation maps, such as slice interleaving, dispersed macro-block allocation using checkerboard-like patterns and using one or more foreground slice groups as shown in Figure 2.4, which allow for different error resilient features to be implemented [88]. Using FMO the macro-blocks are coded into different slices based on the applied allocation map. Therefore, in the event of transmission errors if one of the slice is dropped due to corruption, the lost macro-blocks can still be concealed using the neighbouring macro-blocks transfered using another slice as shown in Figure 2.4.

2. Arbitrary Slice Ordering:
   As suggested by the terminology, Arbitrary Slice Ordering (ASO) allows the arbitrary decoding order of macro-blocks within a slice, in other words, the decoding of consecutive macro-blocks does not have to monotonically follow each other. This assists in achieving a reduced decoding delay in case of out of order delivery of NAL units [74], which may have been imposed by using different routes in the Internet, for example.

3. Data-Partitioning:
   Data-Partitioning (DP) in H.264/AVC allows for the creation of up to three partitions per slice having different subjective importance and hence error sensitivity for transmission of coded information. In contrast to previous standards, where the coded information was partitioned into header and motion

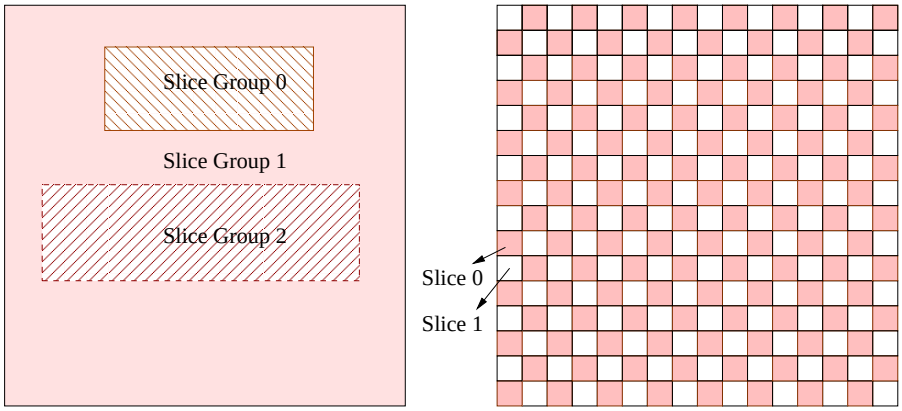

Figure 2.4: Flexible Macro-block Ordering in H.264/AVC [74].

information, in H.264 the last partition is separated into intra-coded and inter-coded information. This allows the separation of the more important intra-coded information from the inter-coded information. Hence a considerable reduction of visual artifacts may be achieved by providing prioritised or unequal error protection to different partitions based on their relative importance [89].

4. Intra macro-blocks update:
   The intra macro-block update allow us to curtail error propagation in coded sequences due to packet loss or bit error events in the bit-stream, since the inclusion of image blocks in intra mode allows the prompt recovery from error propagation. H.264 also allows intra coding of macro-blocks, which cannot be efficiently motion predicted. Furthermore, H.264 also allows us to perform the selection of intra-coded macro-blocks either in random fashion or using channel-adaptive rate distortion optimisation [76] [90]. The employment of intra macro-block updates results in significant performance improvements, when the error rate is high. Generally speaking, channel adaptive intra-update results in more beneficial performance improvement than pure random intra-frame updates.

5. Redundant coded slices:
   Redundant coded slices may be transmitted to assist the decoder in eliminating the effects of errors in the corresponding primary coded picture. Examples of this redundant coding technique include the Video Redundancy Coding (VRC) technique of [91] and the protection of key pictures in multi-cast environments [92, 93].

It is important to note that all these error resilient techniques generally result in an increased data rate for the coded video without improving the error-free video quality. Therefore, their application should always be carefully considered in the light of the compression efficiency.

2.5 The H.264 Video Coding Techniques

In this section we will discuss different video coding techniques used by H.264/AVC for enhanced video coding efficiency. The schematics showing the detailed operations of the H.264 encoder and decoder are shown in

Figure 2.5 and Figure 2.6, respectively.

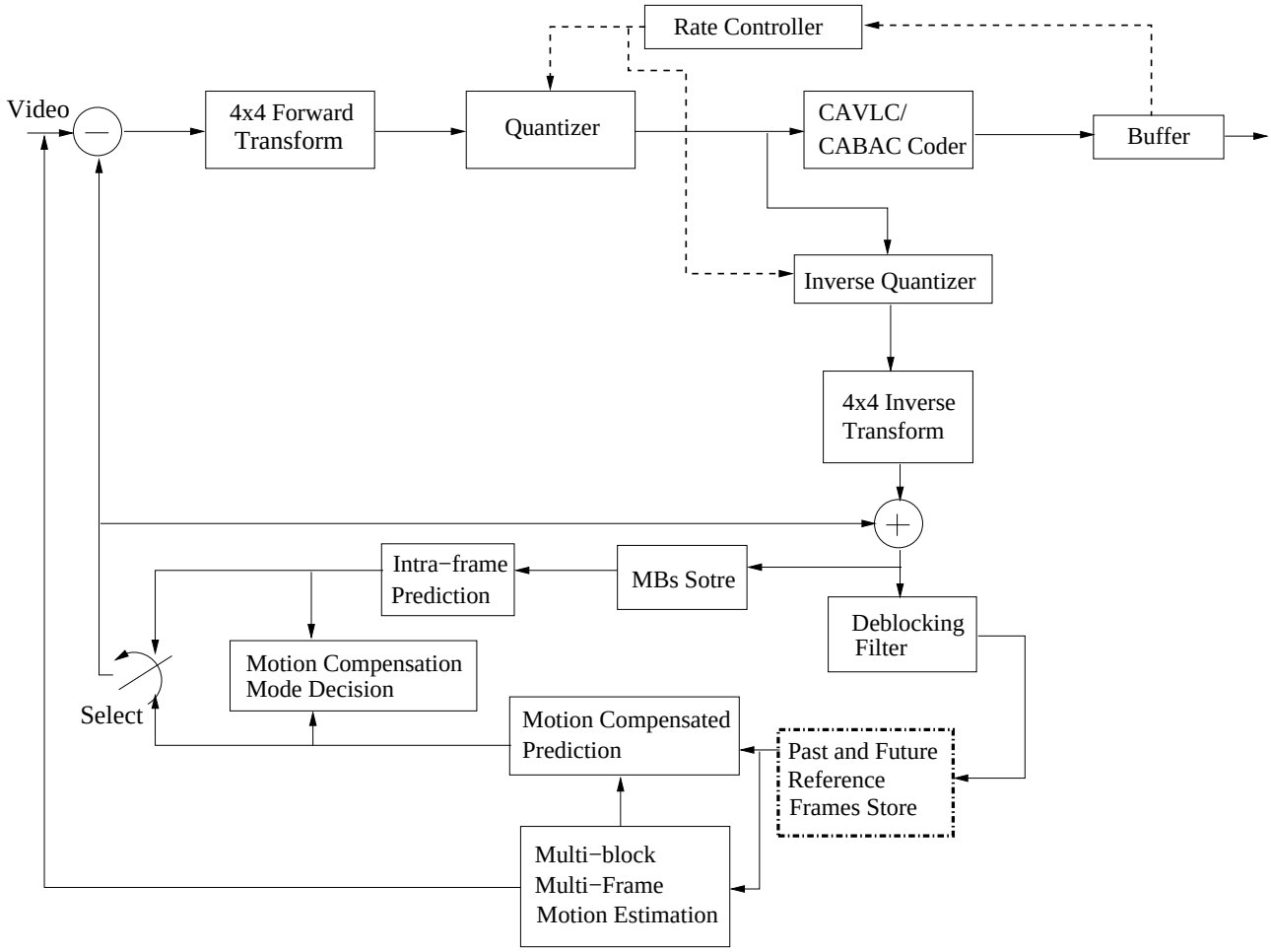

Figure 2.5: H.264/AVC encoder block diagram [94, 95].

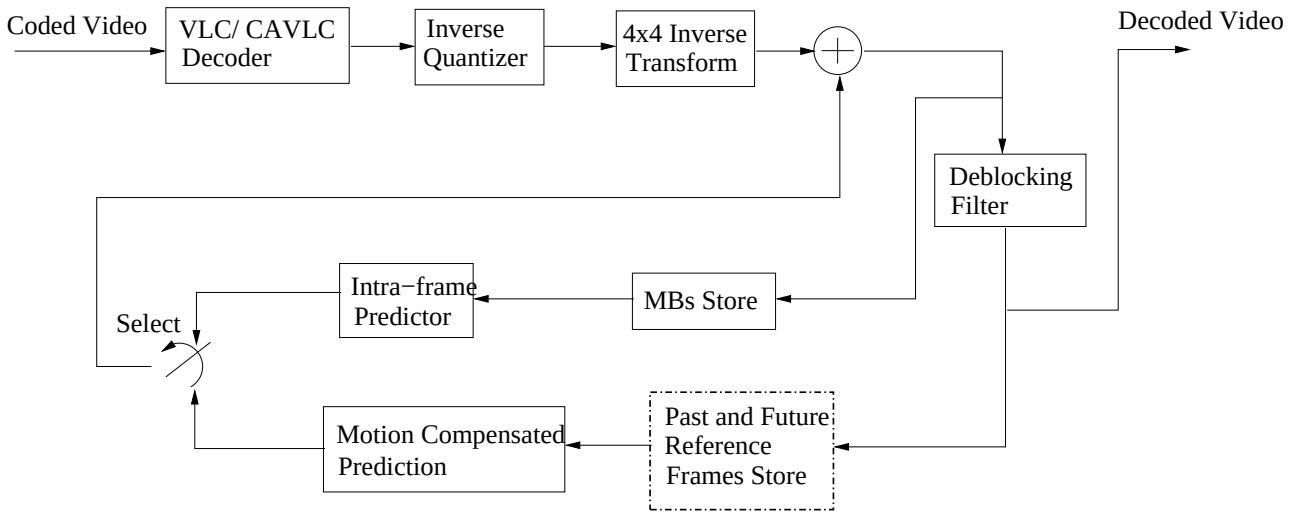

Figure 2.6: H.264/AVC decoder block diagram [94, 95].

2.5.1 Block Based Intra-Coding

In intra-coding already transmitted macro blocks are used for intra-frame prediction. H.264/AVC supports two types of intra-frame prediction for the prediction of the luminance components Y. One of them is referred to as the INTRA_4x4 mode and the second one is the INTRA_16x16 mode. In INTRA_4x4 a macro block of 16x16 picture elements is partitioned into sixteen 4x4 sub-blocks and prediction is applied separately for each 4x4 sub-block. This mode is suitable for encoding the most important parts of the image with fine detail. In H.264/AVC nine different prediction modes are supported. DC-prediction is one of these nine modes, in which all samples to the left and top of the current block, as well as the currently transmitted ones are used for block prediction. Similarly, the remaining eight prediction modes, each having a specific prediction direction are shown in Figure 2.7. For example, in vertical prediction (mode 0) all samples below sample A are predicted by sample A and all samples below B are predicted by sample B and so on. Similarly, in the horizontal prediction (mode 1) horizontal samples are used for prediction, as shown in Figure 2.8 [31].

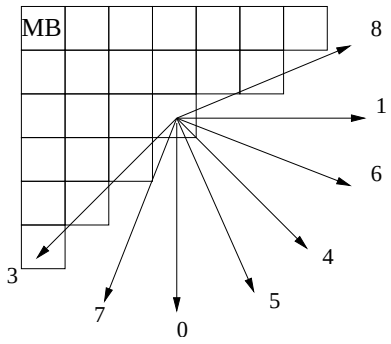

Figure 2.7: Possible prediction directions for the INTRA_4x4 mode [74].

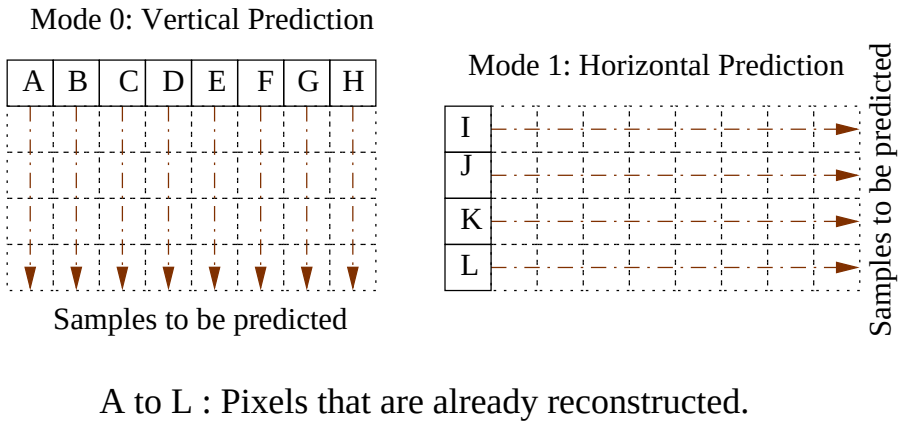

Figure 2.8: Horizontal and vertical prediction for the INTRA_4x4 mode [74].

In the Intra_16x16 prediction mode, prediction of the whole macro block is performed without partitioning, which is suitable for encoding smooth areas of the picture, while using four different prediction modes, which are vertical prediction, horizontal prediction, DC-prediction and plane prediction. The plane prediction uses a linear function to predict the current sample from the neighbouring left and top samples, while the operation of remaining modes is similar to that defined for the 4x4 prediction mode, except that they are applied for the

entire macro-block instead of 4x4 sub-blocks [31]. As the chroma is usually smooth over relatively large areas, the chrominance samples of a MB are always predicted using a technique similar to the Intra_16x16 luma MB prediction. The only difference is that it is applied to 8x8 blocks instead of 16x16 blocks.

2.5.2 Block Based Inter-Frame Coding

This technique is also referred to as Motion Compensated Prediction (MCP). There are two types of inter-frame prediction techniques each associated with a different slice type, which are detailed below.

2.5.2.1 P-Slice Based Inter-Frame Prediction

In inter-frame predictive coding MBs are predicted from the already encoded image MBs, where the macro-block can also be divided into smaller partitions. The luminance partitions having block sizes of 16x16, 16x8, 8x16, and 8x8 samples are supported by the syntax. In case of 8x8 sub-partitions, an additional syntax element has to be transmitted, which specifies, whether the corresponding 8x8 sub-MB has to be further partitioned into block sizes of 8x4, 4x8 or 4x4 samples. The partitioning of macro-blocks is shown in Figure 2.9. Motion compensation is performed separately for each partitioned block. Thus, if a macro block is encoded by partitioning it into four 8x8 sub-partitions and then each 8x8 sub-partition is further decomposed into four 4x4 partitions, then motion compensation of the whole macro block will result in a maximum of sixteen motion vectors, which are transmitted for a single P macro block [31]. In the H.264 video codec using a motion vector accuracy of

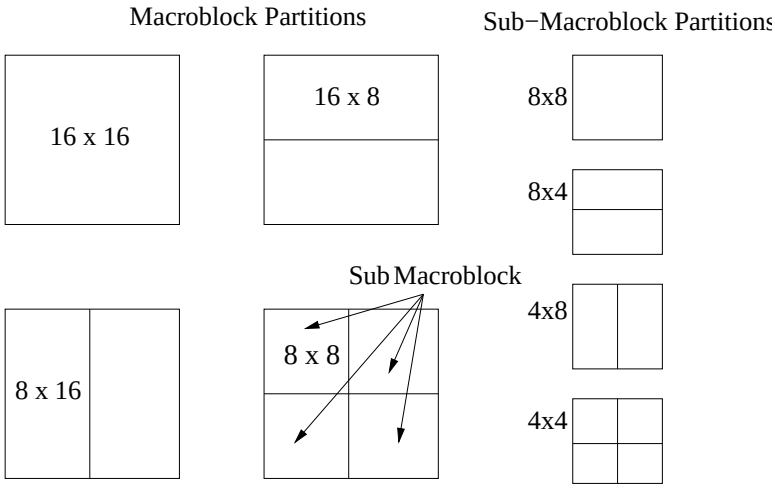

Figure 2.9: Partitioning of a Block and sub-macroblock.

as fine as a quarter-pel accuracy is possible. The motion estimation at a quarter of a picture element's resolution is achieved by generating pixels at non-integer pixel-index positions using interpolation. The luminance samples at half pixel positions are generated by using a 6-tap Finite Impulse Response (FIR) filter, designed to reduce the aliasing artifact. Aliasing artifacts result in reduced interpolation accuracy and hence reduced motion prediction accuracy [96]. The image samples at quarter-pel positions are generated by averaging the pixels at integer and half-pel positions. The samples of the chroma components are predicted using bi-linear

interpolation. Since the sampling resolution of the chrominance samples is half that of the luma resolution, the motion compensation used for the chroma components has 1/8 sample position accuracy.

Interpolated Sample Generation

For the sake of generating samples by interpolation at fractional pixel-positions, first samples are interpolated to create pixels half-way between integer-position samples, as illustrated by the grey pixels in Figure 2.10. For example, to create samples by interpolation at half-pixel-positions, such as those at labels 'b' and 'h', first the intermediate values 'b_1' and 'h_1' are calculated by applying a 6-tap Finite Impulse Response (FIR) filter [97], as shown below [94],

$$b_1=(E-5F+20G+20H-5I+J)$$
$$h_1=(A-5C+20G+20M-5R+T).$$

The predicted pixel values for 'b' and 'h' are obtained from 'b_1' and 'h_1' as follows,

$$b=(b_1+16)/32,$$

$$h=(h_1+16)/32.$$

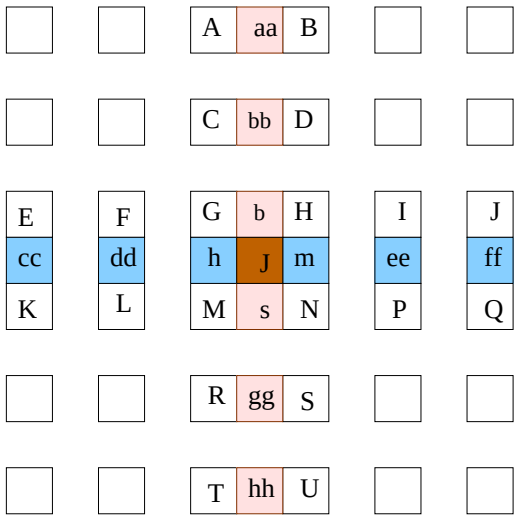

Figure 2.10: Interpolation of the luminance values at half-pixel positions.

The samples at the half-sample central position 'j' of Figure 2.10 are obtained by interpolation from the six horizontal pixels of (cc, dd, h, m, ee, ff) or from the vertical half-pixel samples of (aa, bb, b, s, gg and h) in Figure 2.10, which were calculated in previous steps. Statistically speaking, interpolation based on the horizontal or vertical samples has the same result.

Once all the values at half-pixel positions were calculated, the pixels at quarter-pixel positions are interpolated using a further stage of linear filtering, as shown in Figure 2.11. Explicitly, the samples at quarter-pel positions that are associated with two horizontal or vertical adjacent half- or integer-pixel positions (such as a, c, i, k and d, f, n, q) in Figure 2.11 are calculated using linear interpolation from the adjacent pixels [97]. For example, 'a' can be calculated as,

$$a = round((G + b)/2)$$

The remaining pixels, such as e, g, p, and r of Figure 2.11 are calculated from a pair of diagonally adjacent half-pel position samples, where, for example, 'e' is calculated from 'b' and 'h'.

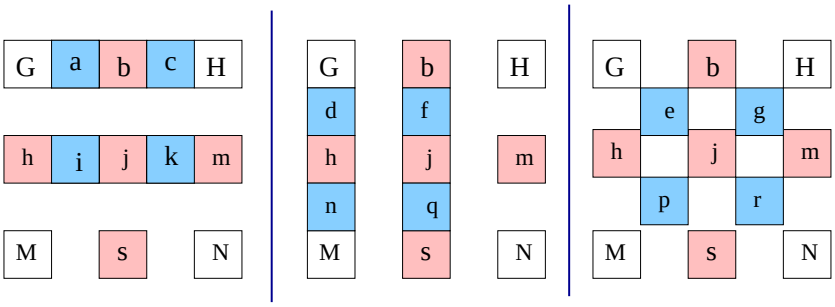

Figure 2.11: Interpolation of luma quarter-pixel position.

This more accurate motion compensation allows us to opt for full-sample, half-sample and quarter-sample prediction, which constitutes one of the major improvements in comparison to earlier standards. More details about fractional-pixel motion compensation and on its accuracy can be gleaned from [96]. This technique also facilitates motion compensation across frame boundaries. For motion vectors that point outside frame boundaries, the reference frame is extrapolated beyond the image boundaries using edge samples, before interpolation takes place. Motion vectors are differentially encoded using neighbouring blocks in the same slice in order to exploit the correlation between them, which results in a further reduced bit-rate after variable length coding.

H.264 also supports multiple reference-frame based motion compensation, where more than one previous encoded pictures can be used for motion compensation, as shown in Figure 2.12. For this purpose multiple pictures have to be stored at both the encoder and the decoder [98]. The reference picture parameter, which specifies the index of the reference picture in the multi-picture reference buffer has to be transmitted for each motion compensated 16x16, 16x8, 8x16 or 8x8 sub-MB [99].

Another efficient feature of H.264/AVC is that large frame areas associated with either a constant change or no motion can be encoded with the aid of a few bits using the so-called Skip P (SP) macro block mode. For this specific coding mode neither the motion vectors nor the reference index parameter or prediction error residual is transmitted. The reconstructed signal is computed in a way similar to the prediction of (16x16)-pixel macro blocks with reference to the picture index 0. In contrast to the previous standards, the motion vectors used for the reconstruction of the skipped macro blocks are obtained using neighbouring macro blocks' motion vectors, instead of considering zero motion for the skipped block.

2.5.2.2 B-Slice Based Inter-Frame Prediction

In the H.264/AVC codec the B picture concept is more general in comparison to previous video coding standards, because for example in the MPEG-2 the B-pictures are encoded using a linear combination of both past and future pictures as a reference, which was an average of two predicted signals. However, the H.264/AVC

allows the weighted combination of several different signals, regardless of the temporal direction in conjunction with arbitrary weights. For example, a linear combination of two forward predicted signals may be used as shown in Figure 2.12. H.264/AVC also allows us to use images containing B slices as the reference image for further predictions, which was not allowed in previous standards. Further details about this generalised B-slice concept, which is also known as multi-hypothesis MCP can be found in [100, 101]. The B slices utilise two

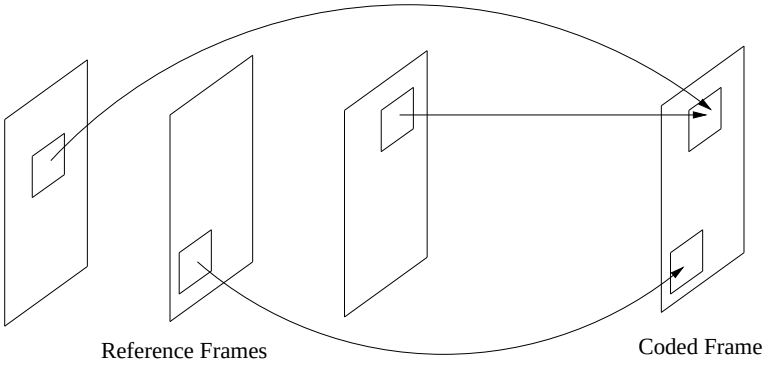

Figure 2.12: Inter-frame prediction using multiple reference frames.

distinct lists of reference pictures, known as list 0 and list 1, which are used for indexing the above-mentioned multi-picture buffer. The procedure of selecting which particular pictures are located in each of these two lists depends on the multi-picture buffer control procedure. The order of pictures displayed and used as a reference can be different, which is achieved with the aid of this procedure. This results in an increased flexibility in terms of choosing reference pictures that are used by other pictures for motion prediction. This flexibility is particularly beneficial in Scalable Video Coding (SVC) and Multi View Coding (MVC) standards [102, 103].

Four different types of inter-picture prediction are supported by B slices, namely the above-mentioned list 0, list 1, bidirectional-predictive and direct prediction. Therefore, Inter-layer prediction can use either list 0, list 1 or both, where the latter uses a combination of list 0 and list 1, which is the main feature of B slice inter-frame prediction. The same partitioning as that devised for P slices is also applied for B slices and for each partitioned block one of the four possible inter-picture prediction type is used. In the direct prediction mode the partitioned blocks are subjected to simple frame-differencing and hence they are encoded without transmitting any motion information, such as motion vectors. If neither motion vectors nor prediction residual information are transmitted for the direct-prediction based macro blocks, then they are referred to as skipped blocks, and skipped macro blocks can be differentiated using the same parameter as in case of the skip mode of P slices [31].

2.5.3 Block Based Transform Coding and Quantisation

Transform coding constitutes a well-established technique of transforming the video frame from the spatial video domain to frequency domain, which allows us to reduce the residual spatial redundancy in the Motion Compensated Error Residual (MCER). Transform coding techniques employed in previous standards, such as the MPEG-1 and MPEG-2 standards are based on two dimensional DCT [104] of size 8x8. In H.264/AVC a different transform known as the 4x4 integer transform having similar properties as the 4x4 DCT is used,

although in some special scenarios the 2x2 integer transform is also invoked.

Some of the advantages of using smaller transform block sizes are;

- The smaller block size of 4x4 instead of 8x8 pixels facilitates better motion estimation for objects of smaller sizes both in the inter- and intra-frame coded mode. As a result, the residual signal has a reduced spatial correlation.
- The smaller transform size results in reduced artifacts around the block edges, such as mosquito noise or ringing artifacts.
- The smaller transform size also results in reduced computational complexity owing to its smaller processing word length.

To elaborate a little further, three different types of transforms are used, which are described by the transform matrices H1, H2 and H3 shown in Figure 2.13. The transform H1 of size 4x4 is applied to all blocks of the luminance component and to two chrominance components, regardless, whether inter- or intra-frame coding is used. If the predicted macro-block is of type Intra_16x16, then the second transform H2 known as the Hadamard transform of size 4x4 is applied after the first one. It is also used to transform all DC coefficients of H1-transformed blocks of the luminance component.

$$H1 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 1 & -1 \end{bmatrix} \quad H2 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \quad H3 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

Figure 2.13: Three different transform matrices in H.264/AVC [74].

The transform H3 of size 2x2 is also a Hadamard transform as shown in Figure 2.13, which is used for the transformation of the four DC components of each chrominance component. Figure 2.14 shows the transmission order of all the transform coefficients. If a macro block is coded using the Inra_16x16 type, then the block labeled -1 in Figure 2.14 contains the DC coefficients of all blocks of the luminance component and this is transmitted first. After that all blocks labeled 0-25 in Figure 2.14 are transmitted, where blocks 0-15 of Figure 2.14 contain all AC coefficients of the blocks of the luminance component. Furthermore, blocks 16 and 17 of Figure 2.14 contain the DC coefficients while blocks 18-25 convey the AC coefficients of the chrominance components, which are transmitted last. In contrast to the DCT, all the integer transforms have integer coefficients in the range of -2 to 2, as shown in the transform matrices of Figure 2.13. This allows us to compute the transform and its inverse using simple shift, add and subtract operations in 16-bit arithmetic. On a similar note, in case of the Hadamard transform only add and subtract operations are required for both the forward transformation and for its inverse. As a benefit of using integer operations, any potential mismatch problem imposed by the inverse transform is completely avoided, which was problematic in former standards [31].

All transform coefficients are quantised using a scalar quantiser known as the Uniform Reconstruction Quantiser (URQ) [38]. The quantisation step size is selected using the Quantisation Parameter (QP), which

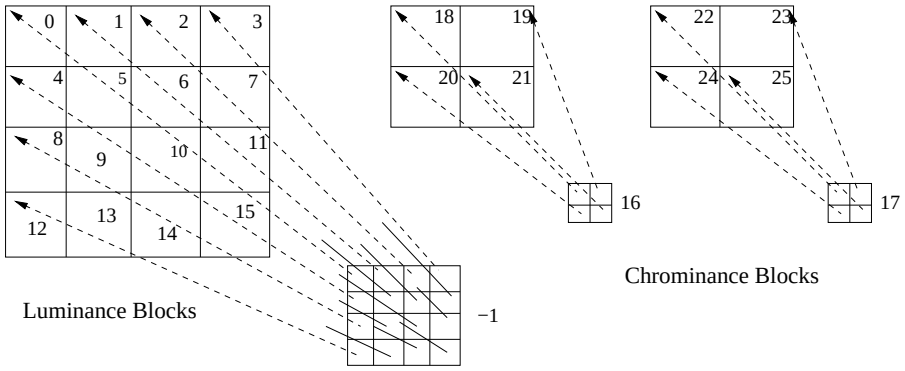

Figure 2.14: Transmission order of macroblock transform coefficients [74].

supports 52 different quantisation schemes, corresponding to 52 different bit-rates. The step size of the quantiser doubles with each increment of 6 in the QP. An increase in the QP by 1 results in an approximately 12.5% increase in the data rate. The transform is explained in more detail in [105].

The quantised transform coefficients are generally scanned in a zigzag fashion, while the 2x2 DC coefficients of the chroma component are scanned in raster-scan order and are transmitted after entropy coding.

2.5.4 De-blocking Filter Within the Prediction Loop

In low bit rate video coding quantisation has to be performed coarsely, which typically results in visually noticeable discontinuities along the transform block boundaries. This phenomenon results in the blocking artifacts exemplified in Figure 2.15. The effect of these artifacts is also propagated to successive frames during the motion compensation process. Therefore, the removal of these artifacts results in a considerable improvement of the perceptual video quality. In previous standards de-blocking was typically carried out as a post-filtering operation, affecting only the picture to be displayed. According to this method the past frame having potential blocking artifacts was then used for motion estimation, which often resulted in a degraded decoded video quality. Therefore, in the H.264/AVC standard deblocking filtering is carried out within the encoding loop. This results in the employment of the filtered version of previous frames for motion compensation and hence produces a high visual quality by removing blockiness without affecting the quality of the video contents, as shown in Figure 2.16. Another reason for using the de-blocking filter within the coding loop is to assist the decoder in delivering the best possible output quality, with no blocking artifacts. The decoder used by the H.264/AVC codec is highly adaptive. Several parameters such as the quantisation threshold and the local spatial as well as temporal characteristics of the picture may be used to control the filtering process [31], where the de-blocking filter is adaptive on three levels:

- The de-blocking filter can be adjusted based on the characteristics of the sequence within a given video slice with the aid of the slice-level de-blocking filter;
- The second level of de-blocking operation is referred to as block-level filtering. In the block-level filtering operation the filter characteristics are dependent on the block's characteristics, such as the activation

Figure 2.15: Video frame decoding without in loop de-blocking filter.

Figure 2.16: Video frame decoding with in loop de-blocking filter.

of the inter/intra-frame motion prediction, the relative motion and the features of the MCER between neighboring blocks. Based on these parameters, a so-called 'filtering-strength' parameter is calculated, which has a value ranging from 0 to 4, controlling the smoothening effect of the filter from light filtering to strong filtering of the block involved.

- The third level of de-blocking filtering is referred to as sample-level filtering. In the sample-level de-blocking the different levels of filtering are based on the video sample characteristics, which may either be part of true edges in the image or those created by the quantisation of transform coefficients, because in contrast to artifact-based 'false' edges, true edges must be left unaffected as best as possible. For this reason the sample values across every edge are analysed. A tangible example is seen in Figure 2.17, where pixels p0, p1, p2 and q0, q1, q2 are part of two 4x4-pixel blocks with the actual block boundary being between p0 and q0. Filtering of pixels p0 and q0 takes place, if their absolute difference is higher than a certain threshold α. On the other hand, the pixel differences within each block should fall below a certain threshold β, which is considerably lower than α, for the sake of disabling the de-blocking

filtering. The thresholds α and β have to be dependent on the specific quantiser activated. Therefore, the bit-rate-dependent choice of a quantiser links the strength of filtering to the general quality of the reconstructed picture. For smaller quantiser values both the α and β thresholds are set to zero, which deactivates filtering [74].

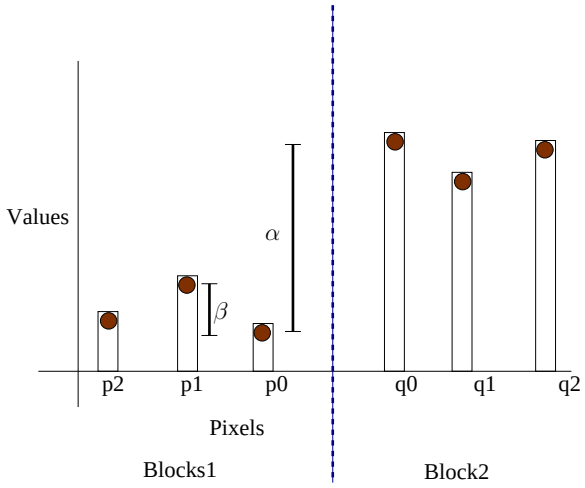

Figure 2.17: Visualisation of one-dimensional sample-level pixel filtering [74].

The employment of the deblocking filter results in a considerable improvements of the subjective quality. It also results in a typically 5-10 % reduction of the bit-rate, while producing the same objective quality as the non-filtered video. A more detailed description of the H.264 deblocking filter can be found in [106].

2.6 Entropy Coding

The H.264/AVC codec uses two alternative methods of variable length coding, which are Context-Adaptive Variable Length Coding (CAVLC) based on a low complexity context adaptive switched sets of variable length codes, and a computationally complex Context-Adaptive Binary Arithmetic Coding (CABAC) scheme. Both methods result in considerable coding efficiency improvement in comparison to traditional coding techniques used in prior video coding standards. In earlier standards specifically designed fixed variable length codes were used for each syntax element or set of syntax elements, whose probability distribution was assumed to be similar. The underlying statistics of the source were always assumed to be stationary, which in practice rarely happens. Actually, the MCER generated by motion compensated predictive coding often exhibits non-stationary statistical characteristics, especially in the presence of high-resolution motion compensation, which removes much of the predictable video content. Therefore, by incorporating any of the above-mentioned methods of variable length coding in the H.264/AVC codec, agile adaptation to the statistical characteristics of the underlying source has to be achieved, which results in a high compression efficiency, while striking an appropriate complexity-compression trade-off [74].

In CAVLC the basic coding tool is constituted by a single Variable Length Coding (VLC) table of Exp-Golomb codes [107], which are applied to all syntax elements, except for the quantised transform coefficients.

For the quantised transform coefficients a more sophisticated coding scheme is designed, where the given block of quantised transform coefficients is re-arranged for a one-dimensional array according to the predefined scanning order of the TCs using the classic zigzag scan. The goal of zig-zag scanning is to ensure that the TCs having similar statistical properties which tend to be in each other's vicinity both vertically and horizontally remain close to each other and employ the same quantisers [1]. After TC quantisation a block typically contains only a few significant nonzero coefficients originating from the top left corner of the block, along with a large number of coefficients having a low magnitude equal to 1, which appear at the end of the coefficient stream. As a first step, the number of significant transform coefficients and the number of trailing 1 values is transmitted using a combined VLC codeword. Then, in the second step the list of significant transform coefficients is scanned in reverse order and their sign and magnitude values are encoded. Again, the VLCs are adopted for encoding each individual TC value based on the previously encoded value by choosing from six VLC tables. Finally, the total number of zero-valued quantised coefficients preceding the last non-zero level is transmitted, and for each significant level the number of consecutive preceding zero-valued coefficients is signalled. For typical video sequences bit rate savings of 2-7% may be achieved relative to conventional run-length coding schemes by Exp-Golomb codes [31].

CABAC [108], which is the alternative technique of entropy coding, has significantly improved the attainable coding efficiency. The advantages of CABAC include, but are not limited to:

- Allocation of non-integer number of bits to each alphabet, which is advantageous for symbols associated with probabilities higher than 0.5, because the frequent occurrence of a non-integer number of bits results in significant compression erosion.
- Adaptation to non-stationary symbol statistics, due to the employment of adaptive codes;
- The employment of context modelling, in which statistics of already coded symbols are used for the conditional probability estimation of symbols.

In the H.264 codec the entire arithmetic coding operations and the associated probability estimation processes are free from multiplications and rely on low-complexity shift and table look- up operations. CABAC results in about 5-15% bit rate reduction relative to CAVLC. More details about the CABAC technique are provided in [106].

2.7 Rate-Distortion Optimised Video Coding

The H.264 standard defines the bit-stream syntax along with different coding 'tools', which may be activated from the 'toolbox' of techniques for achieving a high coding efficiency, but the achievable coding efficiency also depends on the specific network conditions encountered. Figure 2.18 depicts the different stylized rate versus distortion operating points for different encoder strategies. If the minimisation of the rate is of prime concern, then using a specific strategy in favour of achieving a reduced data rate may imply that the distortion would be high. Similarly, when aiming for distortion minimisation, the strategy applied in favour low distortion would result in a high data rate. Both of these operational points may be undesirable. A desirable point is

one, which results in both rate and distortion minimisation. This operational point may be found by using Lagrangian optimisation techniques [109] for the sake of identifying the most appropriate choice of macro-block coding mode and motion vector estimation [110] [109] [111]. This implies that an undesirable decision at the encoder may result in poor coding efficiency or poor error resilience or in fact both. Usually in bi-directional communication, the encoder has some awareness of the NAL unit loss experienced at the decoder owing to packet-loss events, which may be achieved by conveying some feedback information to the encoder from decoder. Naturally, this feedback is also subject to delays. Although the employment of packet-retransmission is not feasible in low-delay lip-synchronised interactive applications, retransmissions are useful at the encoder in terms of limiting error propagation in broadcast-type video-streaming.

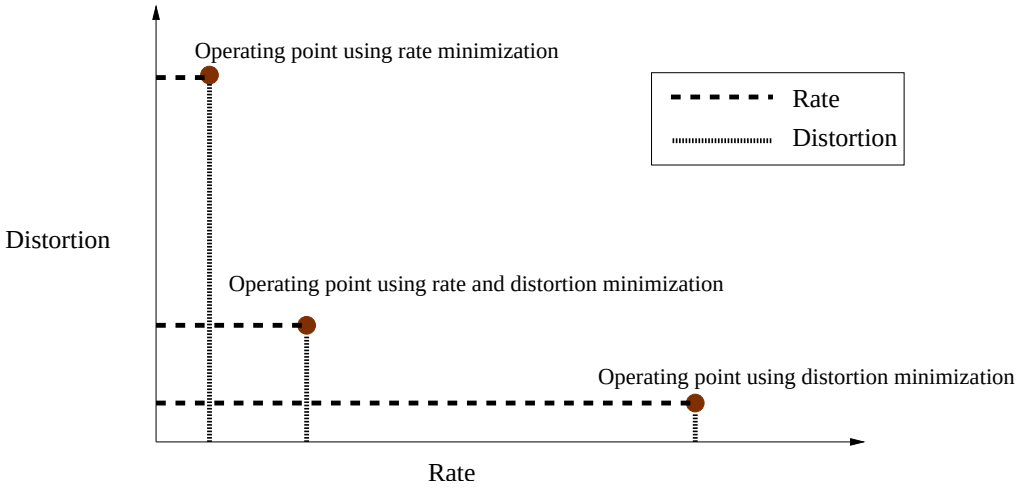

Figure 2.18: Rate distortion operational points based on different encoder strategies.

2.8 H.264/AVC Video Coding Profiles and Levels

The H.264/AVC video codec was developed to address a broad range of applications at different bit rates, video resolutions, video qualities and services characteristics. However, different applications impose different requirements in terms of video quality, error resilience, compression efficiency, delay, and complexity. Therefore, in order to increase the codec's interoperability, while limiting its complexity, the H.264/AVC standard defines various 'Profiles' and 'Levels'. A 'profile' is defined as a subset of standard coding tools. For this reason, side-information parameters and flags are included in the bit-stream, which specify the presence or absence of the corresponding tools in the stream. All decoders that are compliant with certain profiles must support all the tools within that profile. However, there is still a high degree of freedom within the boundaries imposed by the syntax of a specific profile. For example, these variations are dependent on the values assumed by the different parameters, such as the decoded picture size, frame rate etc. For many applications it is neither economical nor practical to implement a decoder, which is capable of processing all possible syntax parameters within a given profile. For this reason, a second profile descriptor known as 'Level' is created for each profile, which specifies a set of constraints imposed on the syntax parameters within each profile. These constraints may either be syntax parameter values or they may be a combination of values, such as the picture width and height

expressed in terms of the number of pixels or the frame rate. In H.264/AVC all profiles employ the same level definitions. Furthermore, if the application considered is capable of supporting more than one profiles, then we have the option of support either the same or different levels for each profile [76]. In H.264/AVC three profiles are defined, which may be invoked for supporting a diverse range of applications. A stylized representation of the capabilities of these profiles is provided in Figure 2.19.

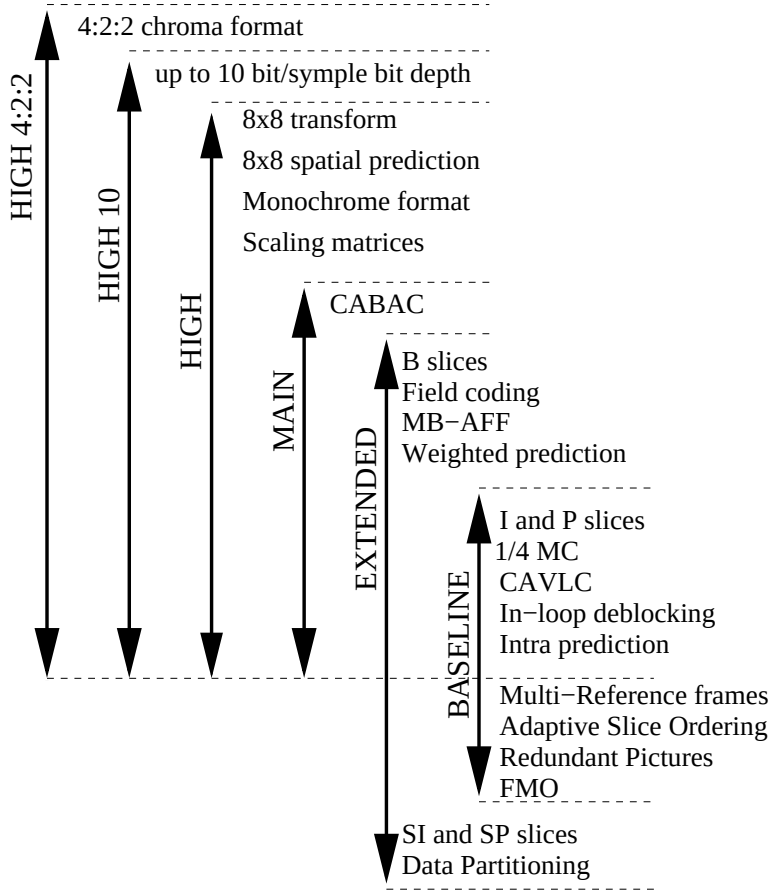

Figure 2.19: Scope of the H.264/AVC profiles [74].

2.8.1 Baseline Profile

The simplest profile of the three is capable of supporting all H.264/AVC tools, except for handling B-slices, interlaced coding, weighted prediction, adaptive switching between frame/ field coding, CABAC, SP/SI slices and data partitioning [84]. This profile typically targets applications with low complexity and low delay requirements [74].

2.8.2 Main Profile

The H.264/AVC coding tools that are not supported in this profile include FMO, ASO, and the transmission of redundant pictures. By contrast, the list of supported tools contains all the above-mentioned Baseline Profile

tools, along with handling B-slices, weighted prediction, interlaced coding, adaptive switching between frame/ field coding and CABAC, again, except for FMO, ASO and redundant pictures. Due to the inclusion of complex tools, such as the support of B-slices and CABAC, this profile provides the best quality at the cost of an increased complexity [84] in comparison to the baseline profile. This profile typically allows the best quality at the cost of higher complexity (essentially due to the B-slices and CABAC) and delay [74].

2.8.3 Extended Profile

This profile contains all the coding tools of H.264, except for CABAC. The SP/SI slices and slice data partitioning are only included in this particular profile, but not in the previously mentioned more simple profiles. It is generally difficult to establish a strong relation between the profiles and their specific applications, but its possible to say that conversational services associated with low delay requirements will typically use the Baseline profile, entertainment services tend to rely on the Main profile, while streaming services using wireless or wired transmission medium may employ the Baseline or Extended profiles [84]. There are 15 levels defined for each profile in H.264/AVC. Each level defines an upper bound for the encoded bit-stream or a lower bound for the decoder's capabilities. These different parameter specifications may include the picture size, ranging from QCIF to (4K x 2K)-pixel high-definition video, the decoder's processing rate of say 1485 to 983 040 macro-blocks per second, the affordable memory size for employment in multi-picture references, the video bit rate ranging from 64 Kbps to 240 Mbps and the motion vector range of say -64 to +64 or -512 to +512 [73].

2.8.4 FRExt Amendment for High-Quality Profiles

In addition to the above-mentioned three profiles, the H.264/AVC FRExt amendment specifies three additional nested sets of profiles relying on the main profile, namely the so-called High, High 10 and High 4:2:2 profiles of Figure 2.19. The High profile contains coding tools for the further improvement of the coding efficiency relative to the main profile and results in a moderate increase of the compression ratio at a modest implementation and computational cost. The High 10 profile further extends the capabilities of the standard. A high pixel-resolution ranging up to 10 bits/pel is supported by this profile. Similarly, High 4:2:2 is used to extend the video format to 4:2:2, which is associated with a high chroma resolution. These profiles extend the capabilities of the standard in order to provide enhanced-quality applications, such as HD consumer applications, including HDTV and computer monitors having a high quality [84].

The Peak Signal-to-Noise Ratio (PSNR) versus bit-rate performance of the various H.264 profiles is presented in Figure 2.22. The *Foreman* video test sequence consisted of 30 frames represented in the (176×144)-pixel Quarter Common Intermediate Format (QCIF) resolution was encoded at 15 frames-per-second (fps) using the JM 13.2 encoder of the H.264/AVC standard. The remaining system parameters of our encoding setup are listed in Table 2.1. It is observed from Figure 2.22 that bearing in mind the high computational complexity of the H.264 FREXT profiles, described in Section 2.8.4 associated with High 4:2:0 and High 4:2:2, they may not constitute realistic design options for the encoding of low-definition, low bit-rate wireless applications. By contrast, the *Extended* profile of Section 2.8.3 combined with the additional error-resilience feature of data partitioning may be viewed as a realistic method of choice for wireless multimedia communications. It is worth

noting that although the PSNR versus bit-rate curves of the different-complexity profiles appear to be similar, their ability to handle diverse video scenes varies. This may therefore result in a noticeably different subjective video quality. Furthermore, the substantially lower PSNR of the FREXT High 4:2:2 profile is due to its higher spatial resolution, which hence results in a reduced-accuracy pixel-representation and a reduced PSNR at a given bit-rate.

Parameters	Baseline	Main	Extended	FREXT: High 4:2:0	FREXT: High 4:2:2
No of encoded frames	45	45	45	45	45
YUV Format	4:2:0	4:2:0	4:2:0	4:2:0	4:2:2
Frame rate	15	15	15	15	15
Transform8x8Mode	OFF	OFF	OFF	ON	ON
Sequence type	IPPP	IBPBP	IBPBP	IBPBP	IBPBP
Entropy coding method	CAVLC	CABAC	CAVLC	CABAC	CABAC

Table 2.1: Parameters used in the profiles of Figure 2.22 for various QCIF-resolution sequences

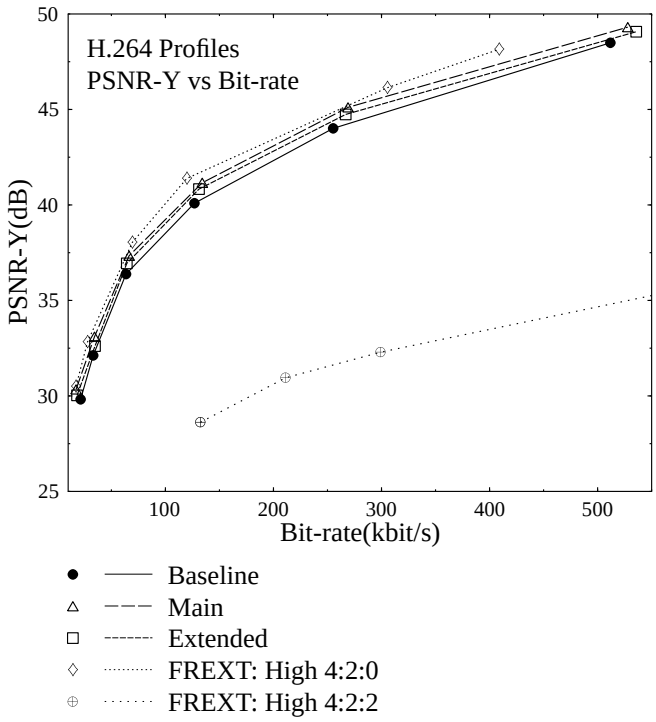

Figure 2.20: PSNR-Y versus bit rate performance of the various H.264 coding profiles using the parameters of Table 2.1 and various QCIF sequences.

Additionally, the subjective video quality of the 45^{th} frame of the *"Foreman"* video sequence using the H.264 profiles of type baseline, main, extended, FREXT: high 4:2:0 and FREXT: high 4:2:2 summarised in Table 2.1 is presented in Figure 2.20. From Figure 2.20 it can be observed that the FREXT: high 4:2:2 profile of the H.264/AVC, which is designed for high definition video applications with high bandwidth requirement has the worst performance while considering low bit-rate video coding scenario. Therefore the FREXT: high 4:2:2 profile can not be considered for our low bit-rate video communication systems.

Furthermore, the error resilience of the different H.264 profiles of Table 2.1 is demonstrated by systematically corrupting the 5^{th} slice in the 3^{rd} frame of the *"Foreman"* video sequence. The resultant artifacts imposed as a result of corruption in the 5^{th} slice of the 3^{rd} frame *"Foreman"* video sequence is shown in Figure 2.21.

Figure 2.21: Subjective video quality of the 45^{th} *"Foreman"* video sequence frame using the H.264 profiles of type (from left) baseline, main, extended, FREXT: high 4:2:0 and FREXT: high 4:2:2 summarised in Table 2.1.

Figure 2.22: Subjective video quality of the 3^{rd} frame of the *"Foreman"* video sequence using the H.264 profiles of type (from left) baseline, main, extended, FREXT: high 4:2:0 and FREXT: high 4:2:2 summarised in Table 2.1 with corruption in the 5^{th} slice of 3^{rd} frame.

2.9 Chapter Conclusions

This chapter has the following findings,

- The focus of the H.264 AVC standardisation activity was to design an efficient and network-friendly video codec capable of supporting a variety of applications, as stated in Section 2.1.
- In Section 2.2 it was stated that the design of H.264/AVC is defined in form of two layers known as the VCL and NAL [76]. The core compression scheme of H.264/AVC is based on the VCL, while the NAL formates the stream generated by VCL to a specific format suitable for various transport layers in multiplex environments, in order to be as network-independent as possible.
- It may be concluded from Section 2.3, that just like the standard video codecs, such as H.261, H.263, H.264/AVC, MPEG-1, 2 and 4 the H.264 codec is also based on the hybrid video coding concept, relying on a combination of video coding techniques, such as motion compensation and transform coding.
- The H.264-coded stream has a flexible network adaptation capability, flexible bit stream structure and high error resilience, as stated in Section 2.4.

- In order to improve the attainable motion compensation capability and to mitigate the blocking artifacts in highly motion active areas of the frame, the H.264/AVC codec supports additional INTRA_4x4 type intra-frame prediction for the prediction of the luminance components Y. In the INTRA_4x4 mode a macro block of 16x16 picture elements is partitioned into sixteen 4x4 sub-blocks and prediction is applied separately for each 4x4 sub-block, as described in Section 2.5.

- The advantages of using the smaller transform block sizes described in Section 2.5.3 are;

  – The smaller block size of 4x4 instead of 8x8 pixels facilitates better motion estimation for objects of smaller sizes both in the inter- and intra-frame coded mode. As a result, the residual signal has a reduced spatial correlation.

  – The smaller transform size results in reduced artifacts around the block edges, such as mosquito noise or ringing artifacts.

  – The smaller transform size also results in reduced computational complexity owing to its smaller processing word-length.

- In comparison to the previous video coding standards the H.264 codec provides significant improvements, including;

  – Enhancement of motion compensation by using high-accuracy motion vectors and multiple reference frames, as described in Section 2.5.2.

  – Employment of an integer transform, instead of the classic DCT, as presented in Section 2.5.3.

  – Adaptive in-loop de-blocking filtering, as detailed in Section 2.5.4.

  – Enhanced adaptive entropy coding, as highlighted in Section 2.6.

2.10 Chapter Summary

In Section 2.1 we noted that the H.264 AVC video coding standard was jointly developed by the ISO/IEC MPEG group and ITU-T VCEG group. In comparison to previous standards, H.264 AVC provides an improved coding efficiency and flexibility for employment over a wide range of network infrastructures. The evolution of the H.264/AVC standard's development from the 1980s combined inter-frame DPCM and DCT based compression designed for video-conferencing type of applications to today's advanced H.264 AVC standard was outlined in Section 2.2. The H.264 hybrid video coding concept was presented in Section 2.3. The list of the main H.264 video coding features, including its network adaptation capability, flexible bit-stream structure and error resilience was summarised in Section 2.4. Section 2.5 provided details about the different video coding techniques used by H.264/AVC for achieving an enhanced video coding efficiency. In the block-based intra-frame coding technique of Section 2.5.1 already transmitted macro blocks were used for intra-frame prediction. It was observed that the H.264/AVC codec supports two types of intra-frame prediction for the prediction of the luminance components, which are referred to as the INTRA_4x4 mode and the INTRA_16x16 mode. The concept of block based inter-frame coding which is also referred to as MCP coding was detailed in Section 2.5.2. Furthermore, the block based transform coding concept used for transforming the video frame

from the spatial video domain to the frequency domain, was detailed in Section 2.5.3 along with the concept of the scalar quantiser known as the URQ used to quantise the transform coefficients. Moreover, the concept of deblocking filtering used for removing the blocking artifacts, which result from the coarse quanization of the encoded coefficients and results in visually noticeable discontinuities along the transform block boundaries was detailed in Section 2.5.4. The two alternative entropy coding methods were presented in Section 2.6, which are CAVLC scheme based on a low complexity context adaptive switched sets of variable length codes, and the computationally more complex CABAC scheme, which result in considerable coding efficiency improvements in comparison to traditional coding techniques employed in prior video coding standards. Section 2.7 describes the rate-distortion based optimisation of the H.264 codec, which is capable of striking an appealing trade-off between rate and distortion minimisation of the coded video based on the specific coding and transmission conditions. Finally, Section 2.8 provided details about the H.264/AVC video coding profiles and levels, designed for a range of applications having different bit rates, resolutions, qualities and services characteristics. The detail provided in this chapter about the H.264/AVC specific coding techniques will be used in the next chapter in order to consider the optimal coding parameters for the video test sequence, while keeping in view the lower complexity and low bit-rate vido telephony scenario.

The advanced coding tools of the H.264/AVC codec described in this chapter may result in about 50% bit-rate savings, relative to previous video coding standards, such as MPEG-4 and MPEG-2 for a wide range of resolutions and bit-rates. However, the price paid for this video-quality improvement is its increased complexity. The H.264/AVC encoder is about 10 times and the decoder is about twice as complex as the corresponding MPEG-4 encoder and decoder in the context of a simple profile. The H.264/AVC Main profile decoder, which is suitable for entertainment applications, is about four times as complex as MPEG-2. The main contributor to the increased encoder complexity is the improved motion compensation and rate-constrained encoder control.

Chapter 3

EXIT Chart Aided Unequal Error Protection Based Video Transmission

3.1 Introduction

The H.264/AVC codec [31] constitutes an attractive candidate for wireless applications, since it was designed to be a high compression standard for network-friendly video transmission. In wireless systems, the charges payable by the end-users are proportional to the amount of transmitted data, where the data rate is limited by both the available bandwidth and transmission power. Therefore, maintaining a high compression efficiency as well as a high integrity, while communicating over heterogeneous communication networks are of paramount importance for wireless multimedia applications [1] [76]. The H.264/AVC standard is conceptually divided into the Video Coding Layer (VCL) and Network Abstraction Layer (NAL) [31], as described in Section 2.4. The hybrid video coding functions are part of the VCL, while the NAL is responsible for facilitating the resultant bit-stream's transport over a wide variety of transport layer protocols. It is stated in Sections 2.6 and 2.3 that the H.264/AVC codec employs various Variable Length Coding (VLC) and predictive coding techniques in order to achieve a high compression efficiency, which however makes the compressed bit-stream vulnerable to transmission errors [1]. This is because a single bit error in the video stream may render the correct decoding of future codewords impossible. Furthermore, owing to predictive coding, the effects of channel errors are likely to be propagated to the neighbouring video blocks. Therefore, the limited bandwidth and error-prone nature of wireless transmission systems makes the transmission of compressed video a challenging task. Various error resilient features [81] such as Data-Partitioning (DP) [112] have been incorporated in the H.264/AVC codec to mitigate the effects of channel errors. The H.264/AVC coded bit-stream using DP results in three different types of streams, each containing specific sets of coding elements having different levels of importance. The design of Unequal Error Protection (UEP) schemes based on the relative importance of the video bits is carried out by carefully considering the error sensitivity of the bit-streams generated. Various error resilient schemes have been proposed in [1] in order to mitigate this error sensitivity problem at the cost of increasing the computational complexity and potentially reducing the achievable compression efficiency. Several robust transmission

techniques [113], such as layered video coding using UEP were shown to be advantageous for H.264 video transmissions in [114]. Error resilient H.264/AVC schemes were created for example employing UEP aided macro-block slices using Reed Solomon codes in [115]. On the other hand, the authors of [116, 117] used the DP mode of H.264/AVC, where UEP was implemented using adaptive hierarchical Quadrature Amplitude Modulation (QAM). Additionally, UEP aided H.264/AVC video transmission was also provided using turbo codes [89]. A robust cross-layer architecture that exploits the advantages of the H.264/AVC error resilience features combined with the Medium Access Control (MAC) capabilities of 802.11 Wireless Local Area Networks (WLANs) was presented in [118]. while a feedback channel aided videophone scheme was presented in [119]. A refined error concealment algorithm based on an adaptive spatio-temporal video estimation method designed for improving the representation of object edges was presented in [120]. An iterative source and channel decoding aided IrRegular Convolutional Coded (IRCC) videophone scheme using Reversible Variable-Length Codes (RVLC) and the Maximum A-posteriori (MAP) probability detection algorithm was proposed in [121]. Furthermore, the performance characteristics of different video transceivers assisted by decision feedback equaliser-aided wide-band burst-by-burst adaptive trellis-coded modulation, Turbo Trellis-Coded Modulation (TTCM) and bit-interleaved-coded modulation designed for H.263 assisted video telephony were presented in [122]. Furthermore, a joint source-channel decoding method based on the MAP [69] algorithm was proposed in [123]. An overview of recent advances in the field of efficient video coding and transmission are presented in Tables 3.1 and 3.2.

Against this background, in this chapter we analyse the effects of diverse H.264/AVC coding techniques on the transmission of an encoded video bit-stream over wireless channels. We propose a novel coding arrangement for the transmission of an H.264/AVC coded video bit-stream. This scheme is utilised for the UEP of the H.264 coded bit-stream using Recursive Systematic Convolutional (RSC) codes [69], while exploiting the different relative importance of different H.264/AVC bit-stream partitions. Furthermore, the resultant H.264 coded bit-stream is also protected using IRCC, which results in an improved performance, when iterative decoding is employed. We cosidered a temporally correlated narrowband Rayleigh fading channel for the coded bit-stream's transmission. Furthermore, the achievable performance of the error protection schemes employed was analysed with the aid of Extrinsic Information Transfer (EXIT) charts.

The novelty and rationale of this chapter can be summarised as follows [7]:

1. *We studied the error sensitivity of the H.264 bit-stream in order to understand the importance of the different segments of the coded stream before applying UEP.*

2. *We applied UEP to the DP-aided and H.264/AVC coded stream using EXIT chart-optimised IRCCs. More explicitly, we proposed a serially concatenated turbo transceiver, consisting of an iterative combination of an outer IRCC and an inner rate-1 precoder. A low bit-rate, head-and-shoulders' type video sequence was encoded using the state-of-the-art H.264 codec and was employed for the performance evaluation of the proposed error protection scheme.*

3. *Additionally, the performance of the error protection schemes employed was analysed with the aid of EXIT charts.*

The rest of the chapter is organised as follows. The concept of iterative detection schemes is presented in Section 3.2 along with an overview of the state-of-the-art research contributions in this field. The theory behind, EXIT charts used for iterative detection convergence testing is described in Section 3.3. The concept of different component codes used in the iterative detection schemes is described in Section 3.4. The video source signal input to the proposed systems is characterized in Section 3.5, while Section 3.6 presents the concepts behind the hierarchical structure of the H.264 coded video stream. The bit-stream syntax of the H.264/AVC coded video is detailed in Section 3.7. In Section 3.8 we describe the H.264 DP concept, followed by our H.264 error sensitivity study in Section 3.9. In Section 3.10 we characterise our UEP based system using RSC codes, based on the system model presented in Section 3.10.1. A novel iterative detection aided H.264 wireless video telephony scheme using IRCCs is proposed in Section 3.11. Our conclusions are presented in Section 3.12, followed by the chapter's summary in Section 3.13.

3.2 Iterative Detection

Iterative detection aided schemes [69] consist of a combination of two or more constituent encoders and interleavers. The principle of iterative detection using concatenated codes was first introduced in [133]. However, due to the lack of sophisticated hardware at that time, it failed to stimulate further research owing to its excessive computational complexity. After the discovery of turbo codes [134], which employed low-complexity component codes, the implementation of low-complexity iterative decoding aided concatenated codes became a practical reality. The innovative iterative decoding of concatenated codes inspired researchers to extend this concept to numerous communication schemes in order to achieve high-integrity transmission of information [135–151]. In [143] the Soft-Input Soft-Output (SISO) *A Posteriori* Probability (APP) module corresponding to the input and output bits of the encoder was described, which exploited the benefits of iterative decoding. Similarly, an iterative extrinsic information exchange was performed between the detector and channel decoder in [142], in order to mitigate the effects of intersymbol interference in digital transmission. Furthermore, in [144, 145] the authors presented the theory behind bit-interleaved coded modulation, complemented by its design guidelines and performance evaluation. As a further advance, the principle of iterative demapping was proposed in [145] for communication systems applying multi-level modulation combined with low-complexity channel coding, in order to reduce the BER. On the other hand, iterative multi-user detection and channel decoding designed for Code Division Multiple Access (CDMA) schemes was proposed in [150]. Moreover, a turbo coding scheme constituted by a serially concatenated block code and an orthogonal Space-Time Block Codes (STBC) was designed for Rayleigh fading channels in [151]. In [137] a serially concatenated system composed of a cascaded outer encoder, an interleaver fed with the outer coded bits and an inner encoder was presented. It was demonstrated that in order to maximise the interleaver gain and to avoid having a BER floor in case of iterative decoding, the employment of a recursive inner code was found to be important. This principle was proposed in [137], which has been adopted by several authors in [152–156], in order to design serially concatenated schemes constituted by the unity-rate precoder as an inner code, when designing low-complexity iterative detection aided schemes suitable for power-limited systems having demanding BER requirements.

Considerable research efforts have been dedicated to the design of semi-analytical tools [152, 157–164]

Year	Author(s) name and contribution
2003	*Author(s):* M. Grangetto, E. Magli and G. Olmo [124] *Contribution:* Arithmetic Codes (ACs) are proposed as a joint entropy coding and error protection tool for robust video transmission over error-prone channels.
2004	*Author(s):* L. Xu and H. Zhou [120] *Contribution:* a set of error concealment techniques are advocated to provide error resilience based on the coding and network characteristics of H.264. An adaptive spatial/temporal estimation method having a low complexity is designed for transmission over mobile IP channels.
	Author(s): H. B. Yu, C. Wang, SY Yu [119] *Contribution:* an algorithm is presented which relies on a feedback channel to achieve prompt error recovery for H.264 video. Its application in conversational services, such as Internet videophones is discussed.
2005	*Author(s):* B. Barmada, M. M. Ghandi, E. V. Jones, and M. Ghanbari [125] *Contribution:* a Hierarchical QAM (HQAM) is used to provide UEP for layered DP H.264 coded video. A multilevel HQAM arrangement based on adaptive constellation point distances provides a graceful degradation in the quality of the decoded video without requiring feedback from the receiver.
2006	*Author(s):* Q. Qu, Y. Pei and J. W. Modestino [126] *Contribution:* a low-complexity motion-adaptive UEP video coding and transmission scheme is proposed, which efficiently combines the existing error-resilience techniques by exploiting knowledge of the source material as well as the near-instantaneous channel quality conditions.
	Author(s): A. Ksentini, M. Naimi and A. Gueroui [118] *Contribution:* characterises H.264 wireless video transmissions over IEEE 802.11 WLANs by proposing a robust cross-layer architecture that activates the inherent H.264 error resilience tools relying on DP and characterises the achievable performance of the proposed architecture.
2007	*Author(s):* YJ. Lee, SW. Lee, Yong H Kim, SI. Lee, ZK. Yim, BH. Choi, SJ. Kim and JS. Seo [127] *Contribution:* the field trial results of the Terrestrial Digital Multimedia Broadcasting (T-DMB) service in Korea are presented. The services provided by the tested T-DMB system included streaming video, CD-quality audio and data services using the H.264 video codec along with additional channel coding and interleaving schemes.

Table 3.1: Video Transmission Advances (Part 1).

Year	Author(s) name and contribution
2008	*Author(s):* P. Ferre, A. Doufexi, J. Chung-How, A. R. Nix and D. R. Bull [128] *Contribution:* a packetisation method is designed for robust H.264 video transmission over the IEEE 802.11 WLAN configured as a wireless home network.
2009	*Author(s):* V. Sgardoni, M. Sarafianou, P. Ferre, A. Nix and D. Bull [129] *Contribution:* robust H.264 coded video delivery for broadcast transmission over the 802.11a/g standard in combination with an accurate time-correlated fading channel is proposed.
	Author(s): S. Ahmad, R. Hamzaoui and M. Al-Akaidi [130] *Contribution:* transmission strategies for H.264 coded video sequences, that aim at minimising the expected bandwidth usage while ensuring successful decoding, subject to an upper bound on the packet loss ratio. The framework invokes a generic strategy, where the source keeps on transmitting the encoded symbols until it receives an acknowledgement from the receiver, indicating that the block was successfully decoded.
2010	*Author(s):* R. A. Farrugia and C. J. Debono [131] *Contribution:* a Hybrid Error Control and Artifact Detection (HECAD) mechanism is proposed to enhance the error resilience of the standard H.264/AVC codec. The proposed HECAD technique relies on residual source redundancy based bit-stream recovery and on a pixel-level artifact detection mechanism invoked for detecting the visually impaired MBs and for concealing them.
	Author(s): S. Gao and K-K. Ma [132] *Contribution:* a new VLC coding scheme is proposed, which is referred to as the two-way decodable variable length data block (TDVLDB) based technique, which renders the compressed bitstream to be bidirectionally decodable and has the capability of effectively recovering more uncorrupted data from the corrupted bit-stream.

Table 3.2: Video Transmission Advances (Part 2).

conceived for analysing the convergence behaviour of iteratively decoded systems. The employment of so-called EXIT charts was proposed by ten Brink in [158], in order to analyse the flow of extrinsic information between the SISO constituent decoders. Furthermore, in [160] the computation of EXIT charts was further simplified by exploiting that the PDFs of the information communicated between the input and output of the constituent decoders are symmetric. A tutorial introduction to the powerful technique of EXIT charts, along with simple examples, typical applications and preliminary analytical results was presented in [161]. Finally, the concept of iterative detection using three stage concatenated systems and their convergence analysis using EXIT charts was provided in [164–166]. The major contributions to the field of iterative detection and their EXIT chart analysis are summarised in Table 3.3, 3.4 and Table 3.5.

3.3 Binary EXIT Chart Analysis

The concept of EXIT charts was proposed by Stephan ten Brink [158] in order to predict the convergence behaviour of iterative decoding without exhaustive bit-by-bit Monte-Carlo simulations by analysing the exchange of mutual information between the inner and outer decoder during iterative decoding. The EXIT charts are based on two assumptions, normely that

1. the *a priori* Log-Likelihood Ratio (LLR) values are fairly uncorrelated, upon assuming large interleaver length:
2. the *a priori* LLR values obey a Gaussian probability density function.

3.3.1 EXIT Characteristics of the Inner Decoder

The input of the inner decoder consists of the demodulated soft-information obtained using the noise-contaminated channel observations and the *a priori* information $L_{M,a}$ fed back to the inner decoder by the outer channel decoder. The inner decoder outputs the *a posteriori* information $L_{M,p}$, which is converted into extrinsic information $L_{M,e}$ by subtracting the *a priori* information $L_{M,a}$ fed back by the outer decoders. With reference to the above-mentioned two assumptions, the *a priori* information $L_{M,a}$ can be modelled using independent zero-mean Gaussian random variables n_A having a variance of σ_A^2 [176]. Therefore, in relation to the outer channel coded and interleaved bits x, the *a priori* input $L_{M,a}$ can be written as [158]

$$L_{M,a} = \mu_A \cdot x + n_A, \tag{3.1}$$

where we have $\mu_A = \sigma_A^2/2$.
Similarly, the conditional probability density function of the *a priori* input $L_{M,a}$ is

$$p_A(\zeta|X=x) = \frac{1}{\sqrt{2\pi\sigma_A}} \exp\left(-\frac{(\zeta - \frac{\sigma_A^2}{2} \cdot x)^2}{2\sigma_A^2}\right). \tag{3.2}$$

where where x is the transmitted symbol and ζ is the received AWGN channel output.

YEAR	AUTHOR(s)	CONTRIBUTION
1966	Forney [133]	introduced concatenated codes.
1974	Bahl *et al*. [167]	introduced the Maximum A-Posteriori (MAP) algorithm.
1993	Berrou *et al*. [134]	invented turbo codes and showed that iterative decoding constitutes an efficient way of improving the attainable performance.
1995	Robertson *et al*. [168]	proposed the log-MAP algorithm that results in a performance similar to that of the MAP algorithm at a significantly lower complexity.
	Divsalar *et al*. [135]	extended the turbo principle to multiple parallel concatenated codes.
1996	Benedetto *et al*. [136]	extended the turbo principle to serially concatenated block and convolutional codes.
1997	Benedetto *et al*. [143]	proposed an iterative detection scheme, where iterations were carried out between the outer convolutional code and inner TCM decoders.
	Caire *et al*. [144,145]	presented the BICM concept and its design rules.
	Li *et al*. [147–149]	combined the BICM with an iterative detection scheme.
1998	Benedetto *et al*. [137, 169]	studied the design of multiple serially concatenated codes combined with interleavers.
	Brink *et al*. [146]	introduced a soft demapper between the multilevel demodulator and the channel decoder in an iteratively detected coded system.
1999	Wang *et al*. [150]	proposed iterative multiuser detection and channel decoding for coded CDMA systems.
2000	Divsalar *et al*. [152, 153]	employed unity-rate inner codes for designing low-complexity iterative detection schemes suitable for bandwidth- and power-limited systems having stringent BER requirements.
	ten Brink [158]	proposed the employment of EXIT charts for analysing the convergence behaviour of iteratively detected systems.
2001	Lee [155]	studied the effect of precoding on serially concatenated systems for transmission over ISI channels.
	ten Brink [159, 165]	extended the employment of EXIT charts to three-stage parallel concatenated codes.
	El Gamal *et al*. [157]	used SNR measures for studying the convergence behaviour of iterative decoding.

Table 3.3: Iterative Detection Contributions (Part 1).

Year	Author(s)	Contribution
2002	Tüchler *et al*. [160]	reduced the computational complexity of EXIT charts.
	Tüchler *et al*. [162]	compared several algorithms predicting the decoding convergence of iterative decoding schemes.
	Tüchler *et al*. [166]	extended the EXIT chart analysis to three-stage serially concatenated systems.
2003	Sezgin *et al*. [151]	proposed an iterative detection scheme, where a block code was used as an outer code and STBC as an inner code.
2004	Tüchler *et al*. [170]	proposed a design procedure for creating systems exhibiting beneficial decoding convergence, depending on the block length.
2005	Divsalar *et al*. [156]	showed that non-square QAM can be decomposed into a parity-check block encoder having a recursive nature and a memoryless modulator. Iterative decoding was implemented by exchanging extrinsic information with an outer code for improving the system performance.
	Brännström *et al*. [164]	considered EXIT chart analysis for multiple concatenated codes using 3-dimensional charts and proposed a way for finding the optimal decoder-component activation order.
2006	Adrat *et al*. [171]	presented a novel approach to quantify the minimum amount of residual redundancy required for successful ISCD.
2007	Alamri *et al*. [172]	demonstrated that the performance of space-time block-coding combined with sphere packing (STBC-SP) modulation can be substantially improved with the aid of iterative information exchange between the SP demapper and the channel decoder. The convergence behaviour of the proposed concatenated scheme was investigated with the aid of EXIT charts.
2008	Maunder *et al*. [173]	designed irregular variable length codes for the near-capacity operation of joint source and channel coding aided systems.

Table 3.4: Iterative Detection Contributions (Part 2).

Year	Author(s)	Contribution
2009	Alamri *et al*. [174]	proposed a three-stage turbo-detected Sphere Packing (SP) aided Space-Time Block Coding (STBC-SP) scheme, where a rate-1 recursive inner precoder was employed to avoid having a BER floor. The convergence behaviour of the proposed serially concatenated scheme was investigated with the aid of 3D Extrinsic Information Transfer (EXIT) Charts.
2010	Kong *et al*. [175]	presented an irregular distributed space-time (Ir-DST) coding scheme designed for near-capacity cooperative communications. At the source node, a serially concatenated scheme comprising an IR-regular Convolutional Code (IRCC), a recursive unity-rate code (URC), and a space-time block code (STBC) was designed. At the relay node, another IRCC was serially concatenated with an identical STBC. The IRCC used both at the source and the relay was optimized with the aid of EXtrinsic Information Transfer (EXIT) charts for the sake of designing a scheme approaching the channel's capacity.

Table 3.5: Iterative Detection Contributions (Part 3).

The mutual information $I_{A_M} = I(X; L_{M,a}), 0 \leq I_{A_M} \leq 1$, between the outer coded as well as interleaved bits x and the LLR values $L_{M,a}$ is used to quantify the information content of the *a priori* knowledge [159]:

$$I_{A_M} = \frac{1}{2} \cdot \sum_{x=-1,+1} \int_{-\infty}^{+\infty} p_A(\zeta|X=x) \cdot log_2 \frac{2 \cdot p_A(\zeta|X=x)}{p_A(\zeta|X=-1) + p_A(\zeta|X=+1)} d\zeta. \tag{3.3}$$

The information content of the extrinsic LLR values $L_{M,e}$ at the output of the inner decoder can be quantified using Equation (3.3), which is then used to compute the mutual information $I_{E_M} = I(X; L_{M,e})$, while employing the PDF p_E of the extrinsic output formulated as

$$I_{E_M} = \frac{1}{2} \cdot \sum_{x=-1,+1} \int_{-\infty}^{+\infty} p_E(\zeta|X=x) \cdot log_2 \frac{2 \cdot p_E(\zeta|X=x)}{p_E(\zeta|X=-1) + p_E(\zeta|X=+1)} d\zeta. \tag{3.4}$$

Using Equation (3.2), Equation (3.4) can be expressed as

$$I_{A_M}(\sigma_A) = 1 - \frac{1}{\sqrt{2\pi\sigma_A}} \int_{-\infty}^{+\infty} \exp\left(-\frac{(\zeta - \frac{\sigma_A^2}{2})^2}{2\sigma_A^2} \right) \cdot log_2[1 + e^{-\zeta}] d\zeta. \tag{3.5}$$

For notational simplicity the following abbrevation is introduced [159]

$$\mathrm{J}(\sigma) = I_{A_M}(\sigma_A) \tag{3.6}$$

$$\lim_{\sigma \to 0} \mathrm{J}(\sigma) = 0, \lim_{\sigma \to \infty} \mathrm{J}(\sigma) = 1, \sigma > 0.$$

Considering I_{E_M} as a function of both I_{A_M} and the E_b/N_0 value encountered, the inner decoder's EXIT characteristics is defined as [159, 177]

$$I_{E_M} = T_M(I_{A_M}, E_b/N_0). \tag{3.7}$$

Figure 3.1 presents the parameters required for the calculation of the EXIT characteristics $I_{E_M} = T_M(I_{A_M}, E_b/N_0)$ for a specific $(I_{A_M}, E_b/N_0)$-input combination. For this purpose the noise variance σ_n is set according to the E_b/N_0 value considered and σ_A is calculated based on the specific I_{A_M} value using $\sigma_A = \mathrm{J}^{-1}(I_{A_M})$ expressed from Equation (3.6). Next $L_{M,a}$ is calculated using Equation (3.1), which is applied as the *a priori* input of the inner decoder. Finally, the mutual information $I_{E_M} = I(X; L_{M,e}), 0 \geq I_{E_M} \leq 1$, between the outer coded and interleaved bits x as well as the LLR values $L_{M,e}$ is calculated using Equation (3.4). The PDF p_E required in Equation (3.4) is calculated by means of Monte-Carlo simulations. Figure 3.2 shows the EXIT characteristics

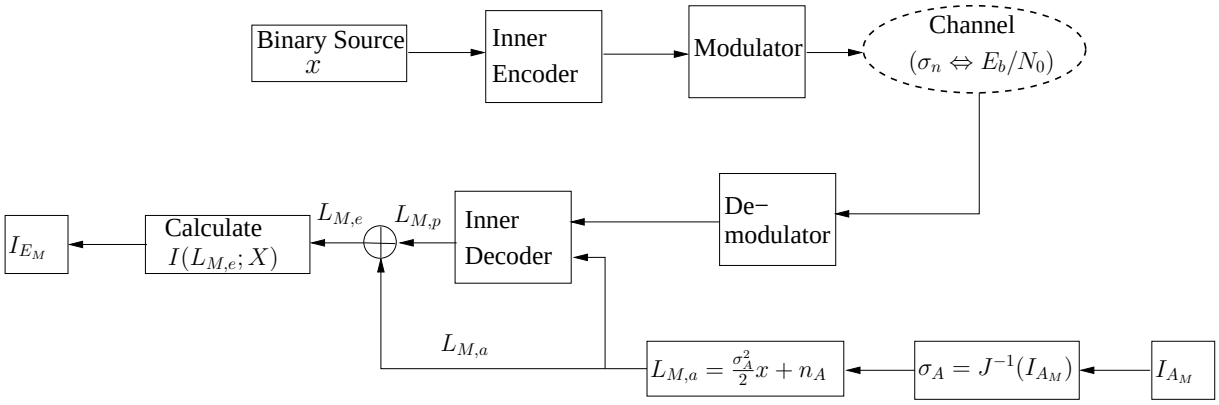

Figure 3.1: Evaluation of the Inner decoder's EXIT characteristics.

of the rate-$\frac{3}{4}$ RSC inner decoder obtained by puncturing the rate-$\frac{1}{2}$ RSC code having the generator polynomials $g_0 = [101]$ and $g_1 = [111]$.

3.3.2 EXIT Characteristics of the Outer Decoder

The extrinsic transfer characteristics of the outer decoder describes the relation between the outer channel coded bits c and the extrinsic information $L_{D,e}$ generated after outer decoding. The input provided for the outer channel decoder consists of the *a priori* input $L_{D,a}$, which is independent of the E_b/N_0-value. Therefore, it can be written as

$$I_{E_D} = T_D(I_{A_D}), \tag{3.8}$$

where $I_{A_D} = I(c; L_{D,a}), 0 \leq I_{A_D} \leq 1$, is the mutual information between the outer channel coded bits c and the LLR values $L_{D,a}$. Similarly, the mutual information between the outer channel coded bits c and the LLR values $L_{D,e}$ is represented by $I_{E_D} = I(c; L_{D,e}), 0 \leq I_{E_D} \leq 1$. The schematic of the EXIT characteristic evaluation for the outer channel decoder is presented in Figure 3.3. Comparing the corresponding schematic of the inner decoder presented in Figure 3.1 to that of Figure 3.3, it can be observed that they employ similar procedures, except that the outer decoder's curve is independent of the signal-to-noise ratio.

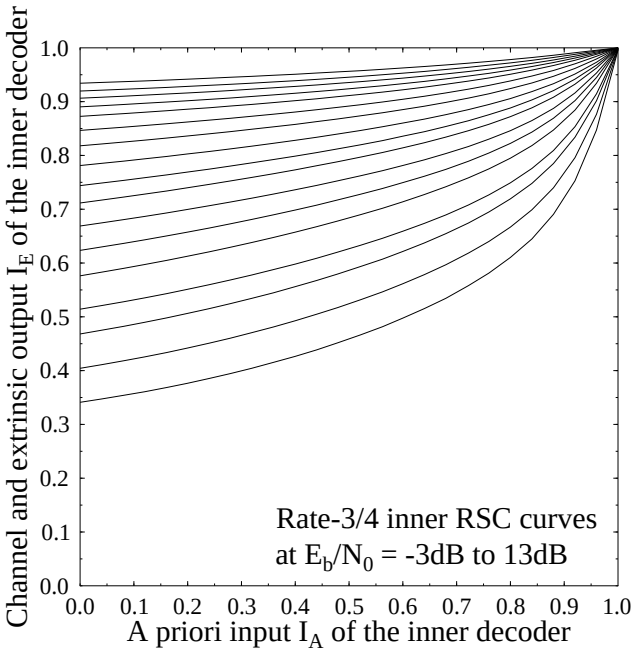

Figure 3.2: Inner decoder's EXIT characteristics at E_b/N_0=-3dB to 13dB in steps of 1dB.

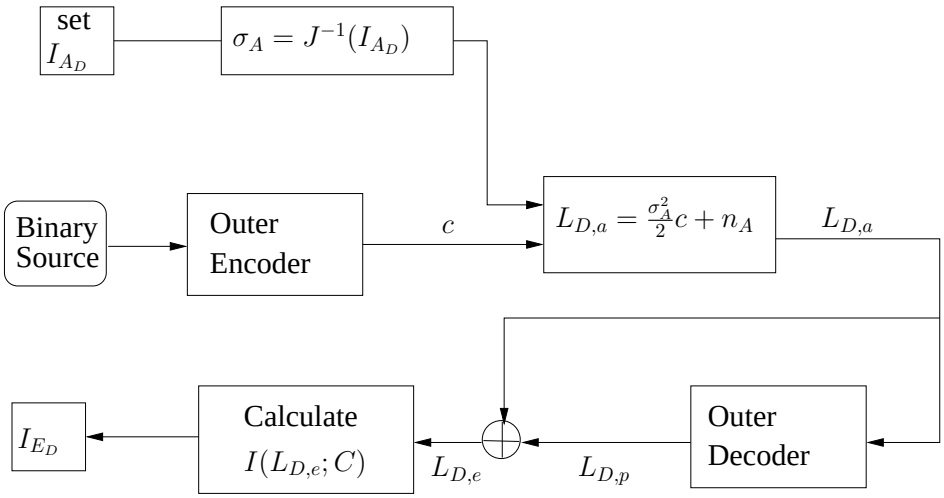

Figure 3.3: Evaluation of outer decoder's EXIT transfer characteristics.

The extrinsic transfer characteristic of the $\frac{1}{2}$-rate outer RSC code having generator polynomials $g_0 = [1\ 0\ 1]$ and $g_0 = [1\ 1\ 1]$ and that of the $\frac{1}{3}$-rate outer RSC code having generator polynomials of $g_0 = [1\ 0\ 1\ 1]$, $g_1 = [1\ 1\ 0\ 1]$ and $g_0 = [1\ 1\ 1\ 1]$ is presented in Figure 3.4. Additionally, the $\frac{2}{3}$-rate and $\frac{3}{5}$-rate RSC outer code EXIT curves provided in Figure 3.4 were obtained for the code generated by employing puncturing to the $\frac{1}{2}$-rate RSC mother code. Similarly, the $\frac{2}{5}$-rate and $\frac{4}{9}$-rate RSC codes are the result of applying puncturing to the $\frac{1}{3}$-rate RSC mother code.

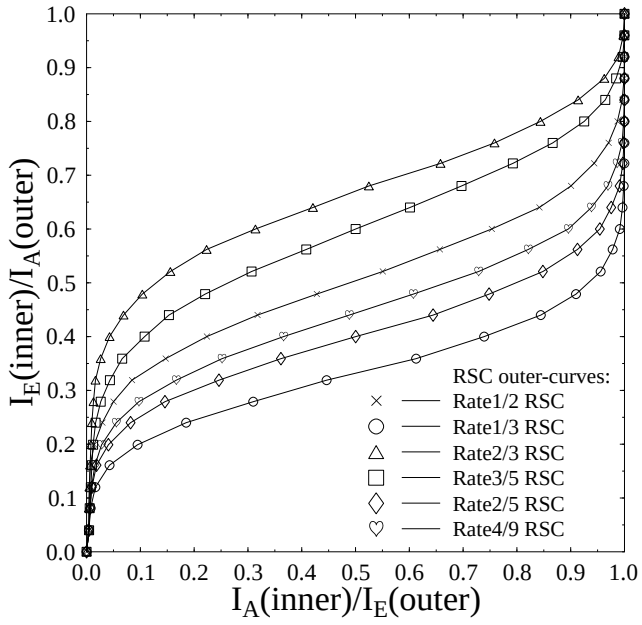

Figure 3.4: EXIT characteristics of various outer RSC codes having different code rates.

3.3.3 Extrinsic Information Transfer Charts

The extrinsic information exchange between the two constituent inner and outer decoders is visualised by plotting the EXIT characteristics of both constituent decoders in a joint diagram, known as EXIT chart [159, 177]. The outer channel decoder's extrinsic output $I_E(outer)$ or I_{E_D} presented on the x-axis becomes the inner decoder's *a priori* input $I_A(inner)$ or I_{A_M}. Similarly, the inner decoder's extrinsic output $I_E(inner)$ or I_{E_M} represented on the y-axis, becomes the outer channel decoder's *a priori* input $I_A(outer)$ or I_{A_D}. Furthermore, the axes of the outer decoders are swapped in the joint diagram for the sake of consistency with the EXIT chart concept, as shown in Figure 3.5.

Figure 3.5 shows the EXIT chart of a turbo-detection aided rate-$\frac{3}{4}$ inner RSC channel-coded scheme employing QPSK modulation in combination with various different outer RSC codes described in Section 3.3.2, when communicating over a correlated Rayleigh fading channel having $f_D = 0.01$. In an ideal EXIT chart convergence situation, the extrinsic transfer characteristic curves of the inner and outer decoders should only intersect at the $(1,1)$ point of the EXIT chart, at the E_b/N_0 value considered. If this condition is satisfied, then a so-called *convergence tunnel* [159,177] appears between the inner and outer EXIT curves. The narrower the tunnel, the more iterations are required to reach the $(1,1)$ point, which results in an increased iterative decoding complexity. However, instead of convergence to the $(1,1)$ point, if the two curves intersect at a point close to $(1,1)$, then a reasonably low BER could still be achieved. The resultant EXIT tunnel may be referred to as a *semi-convergent* tunnel. Observe from Figure 3.5 that for the RSC outer codes of rate-$\frac{2}{3}$, $\frac{3}{5}$, $\frac{1}{2}$, $\frac{4}{9}$, $\frac{2}{5}$ and $\frac{1}{3}$ a *semi-convergent* tunnel exists at E_b/N_0=1dB, 0dB, $-2dB$, $-3dB$, $-3dB$ and $-4dB$, respectively. Therefore, the prediction of the EXIT chart seen in Figure 3.5 implies that the iterative decoding process may be expected to converge to a low BER at the specific E_b/N_0 value at which an open-tunnel is provided for the inner and outer

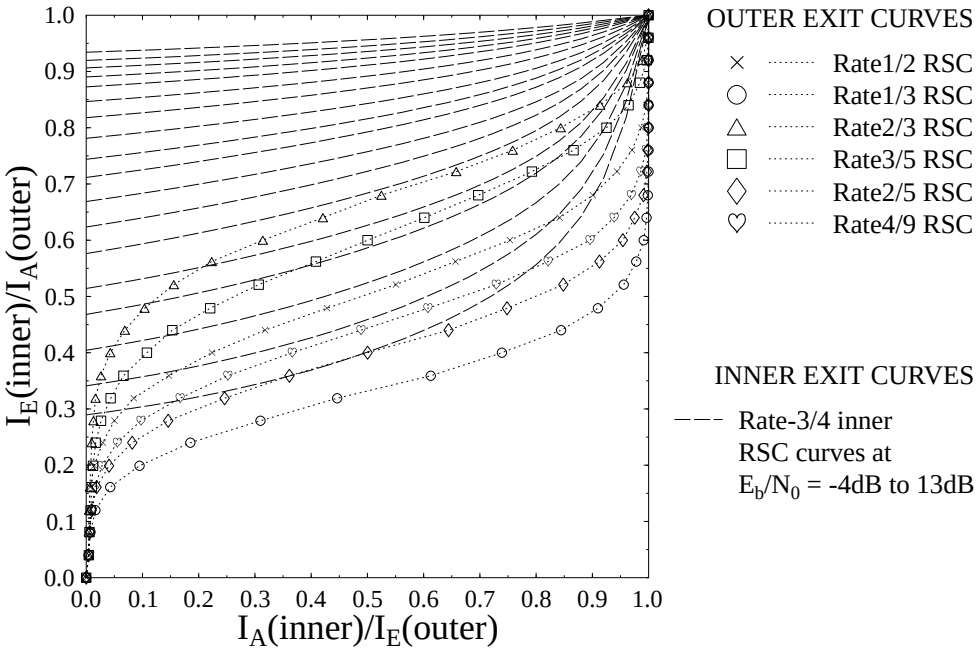

Figure 3.5: EXIT chart of a turbo-detected inner RSC rate-$\frac{3}{4}$ channel-coded scheme employing QPSK modulation in combination with various outer RSC codes described in Section 3.3.2, when communicating with over a correlated Rayleigh fading channel associated with $f_D = 0.01$.

RSC code considered. However, the accuracy of this prediction is dependent on how accurately the assumptions outlined in Section 3.3 are satisfied. In order to verify the EXIT chart based convergence prediction, the actual Monte-carlo simulation-based decoding trajectory of the inner and outer constituent decoders is used, which will be discussed in Section 3.11.4. Below we provide a brief summary of the EXIT-chart properties [161]:

- The area under the outer code's EXIT curve is given by the code-rate;
- The area in the open EXIT-tunnel is proportional to the E_b/N_0 distance from the system's capacity. Hence having an infinitesimally small tunnel-area corresponded to near-capacity operation provided that the $(1, 1)$ point of the EXIT-chart is reach without having an intercept point between the two EXIT curves.
- If there is an intercept point, a residual BER floor is expected.

3.4 Component Codes for Iterative Detection

Different regular and irregular codes can be considered as component codes for employment in iterative detection systems. The concept of regular and irregular convolutional codes is briefly highlighted below.

3.4.1 Recursive Systematic Convolutional Codes

Convolutional codes [178] encode each m-bit input source symbol to an n-bit output source symbol using a mapping, which is a function of the most recent k input symbols. The variable k is referred to as the constraint length and the resultant fraction of m/n is the code rate.

The convolutional encoder associated with a constraint length k consists of k 1-bit shift-register stages. These register stages are initially set to 0 value and their outputs are feeding number of n mod-2 adders, representing the n generator polynomials. As an example, the pictorial representation of a convolutional encoder is shown in Figure 3.6. This convolutional encoder is represented by the generator sequences $g_1 = [111]$ and $g_2 = [101]$, which may equivalently be represented as $G = [g_1, g_2]$. The input bits are fed into the register

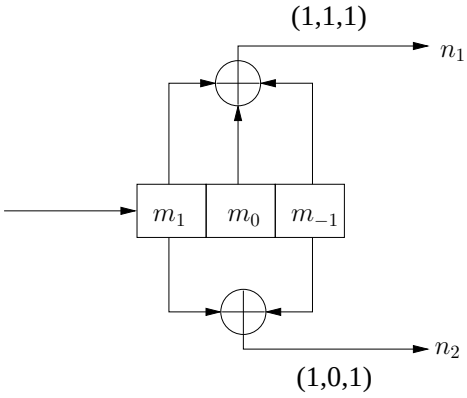

Figure 3.6: Rate-$\frac{1}{2}$ convolutional encoder with constraint length 3.

m_1 and the encoder calculates the output n_1 as well as n_2 based on the binary values in m_0 and m_{-1} using the generator polynomials. Then the binary values in the registers are shifted right to create space for the next input bit in m_1. At the end of input bit stream the encoder will continue to output the encoded bits, until the last input bit passes through the register stage m_{-1} and all registers return to the zero state, as detailed for example in [69].

3.4.1.1 Recursive versus Non-Recursive Codes

If one of the output of the output bits created with the aid of the generator polynomial is fed back to the convolutional encoder's input, then the encoder is referred to as recursive code, otherwise it is a non-recursive code. Figure 3.7 presents a RSC encoder example.

3.4.1.2 Systematic versus Non-Systematic Codes

If the original input bits of the encoder are directly copied to the output bits, then the code is referred to as a systematic code, otherwise it is non-systematic. It can be observed from Figure 3.7 that before the distinct parity bits are concatenated, the input bits that are being encoded are also included in the output sequence, therefore this is a systematic code. This RSC encoder is described by the generator polynomials $g_1 = [111]$

and $g_2 = [101]$, which may also be represented as $G = [1, g_2/g_1]$, where the first output g_1 is fed back to the input. In the above representation 1 is used to denote the systematic output, while g_2 denotes the feedforward output, and g_1 denotes the feedback to the input of the encoder.

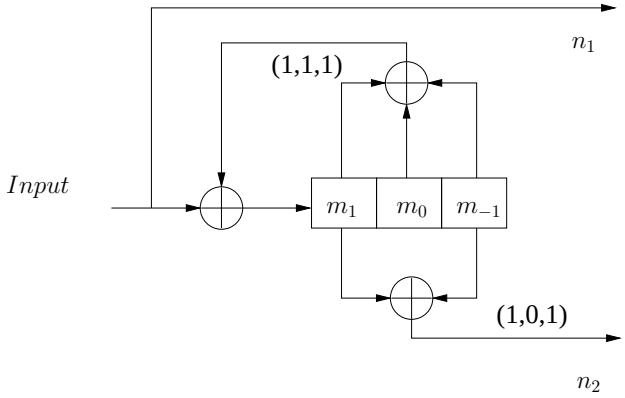

Figure 3.7: Rate-$\frac{1}{2}$ RSC encoder having a constraint length of k=3.

3.4.1.3 Trellis Termination

The state of the encoder after the end of the input sequence is determined by a process referred to as trellis-termination. The trellis is normally terminated by inserting $(K-1)$ additional zero bits at the encoder's input in order to drive it to the all-zero state. However, due to the use of feedback, this strategy is not directly applicable to RSC encoders owing to their recursive structure. A simple strategy to overcome this problem is shown in Figure 3.8, where the switch is turned to position ON during the encoding process, while for terminating the trellis the switch is turned to position OFF [178].

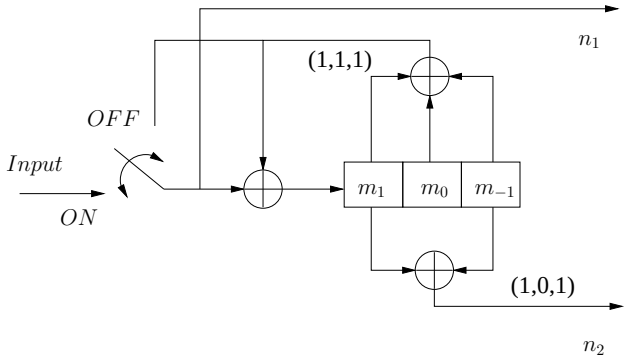

Figure 3.8: Rate-$\frac{1}{2}$ RSC encoder trellis termination strategy.

3.4.2 Irregular Coding

The idea of irregular coding is to provide reliable transmission of information at a channel SNR that is close to the channel capacity bound without imposing an excessive computational complexity and latency. Tuchler and

Hagenauer [179] proposed the serial concatenation and iteratively decoding of an irregular outer code combined with a regular inner code. The outer irregular code comprised N number of component codes, associated with a variety of inverted EXIT functions $\mathrm{I}_A^n(outer)[\mathrm{I}_E(outer]_{n=1}^N$, as shown in Figure 3.9. These component codes were invoked for generating specific fractions of the $\alpha^{nN}_{n=1}$ of the encoded bit-stream, which were appropriately chosen in order to shape the outer EXIT curve $\mathrm{I}_A(outer)[\mathrm{I}_E(outer)]$ of the irregular code by taking into account the EXIT-curve shape of the inner code, according to

$$I_A^o(I_E^o) = \sum_{n=1}^{N} \alpha^n I_A^{o,n}(I_E^o), \tag{3.9}$$

where I_A^o and I_E^o represent the *a priori* and *extrinsic* information of the outer IRCC code, while satisfying

$$\sum_{n=1}^{N} \alpha^n = 1. \tag{3.10}$$

Again, IRCCs were proposed in [170, 179], where a single convolutional code referred to as the mother code was employed to derive a set of component convolutional codes $CC^{nN}_{n=1}$, having a variety of coding rates $R(CC^n)_{n=1}^N$. These different component convolutional codes associated with various coding rates were obtained either by using puncturing [170] or by employing various generator polynomials in addition to the mother convolutional code. Therefore, the overall IRCC coding rate R_{IRCC} may be obtained according to

$$R_{IRCC} = \sum_{n=1}^{N} \alpha^n \cdot R(CC^n). \tag{3.11}$$

The EXIT chart matching algorithm of [179] can be employed to shape the inverted EXIT function of the

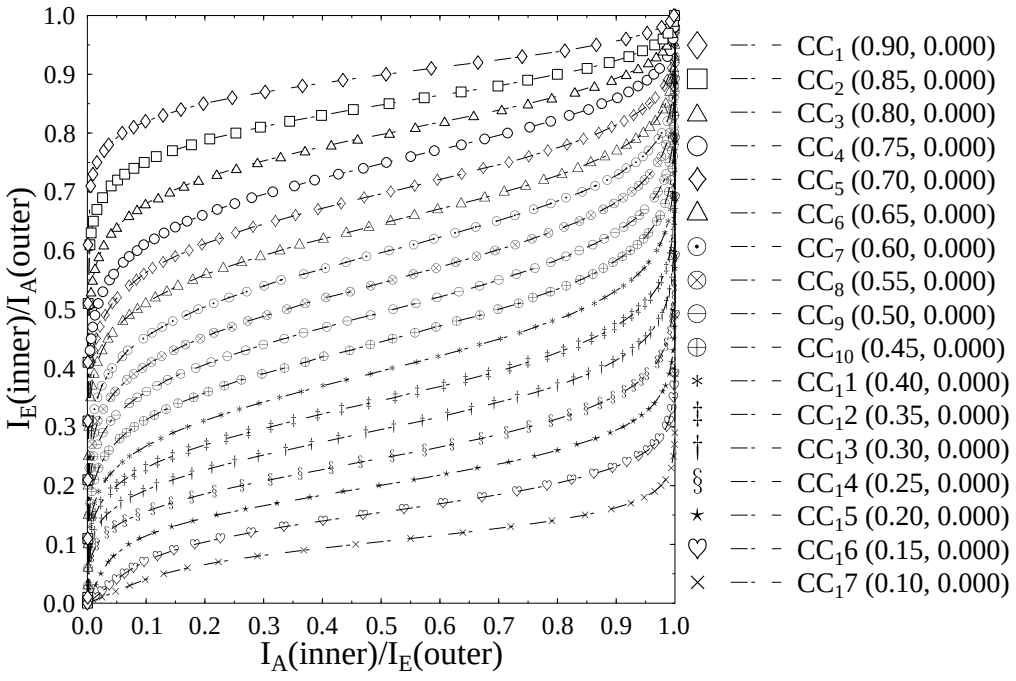

Figure 3.9: The inverted EXIT functions of 17 subcodes used in [160].

irregular outer code by appropriately selecting the fraction $\alpha_{n=1}^{nN}$ of the input stream to be encoded by the different component codes, in order to match its EXIT function to that of the serially concatenated regular inner code. More specifically, the design procedure seeks a particular set of $\alpha_{n=1}^{nN}$ values, which minimises the squared error between the inverted EXIT function of the irregular outer code and of the regular inner code, without intersecting each other.

As an example, the IRCC EXIT chart matching procedure is illustrated in Figure 3.10, where the matching of the inverted EXIT curve of the outer IRCC is fitted to the shape of the inner URC EXIT curve, when considering transmission of BPSK modulated signals over Rayleigh fading channels, at the channel SNR value of 5 dB. According to Equation (3.9), the EXIT function of the IRCC is formed as a weighted average of 17 component codes, provided by the inverted EXIT functions of Figure 3.10. Additionally, this procedure also provides the weights $\alpha_{n=1}^{n17}$ in addition to the coding rates $CC_{n=1}^{n17}$, which can be combined using Equation (3.11) in order to determine the overall IRCC coding rate. The ability of IRCCs to create a narrow but still open EXIT tunnel depends on the availability of a sufficiently diverse range of component codes having a wide variety of EXIT function shapes.

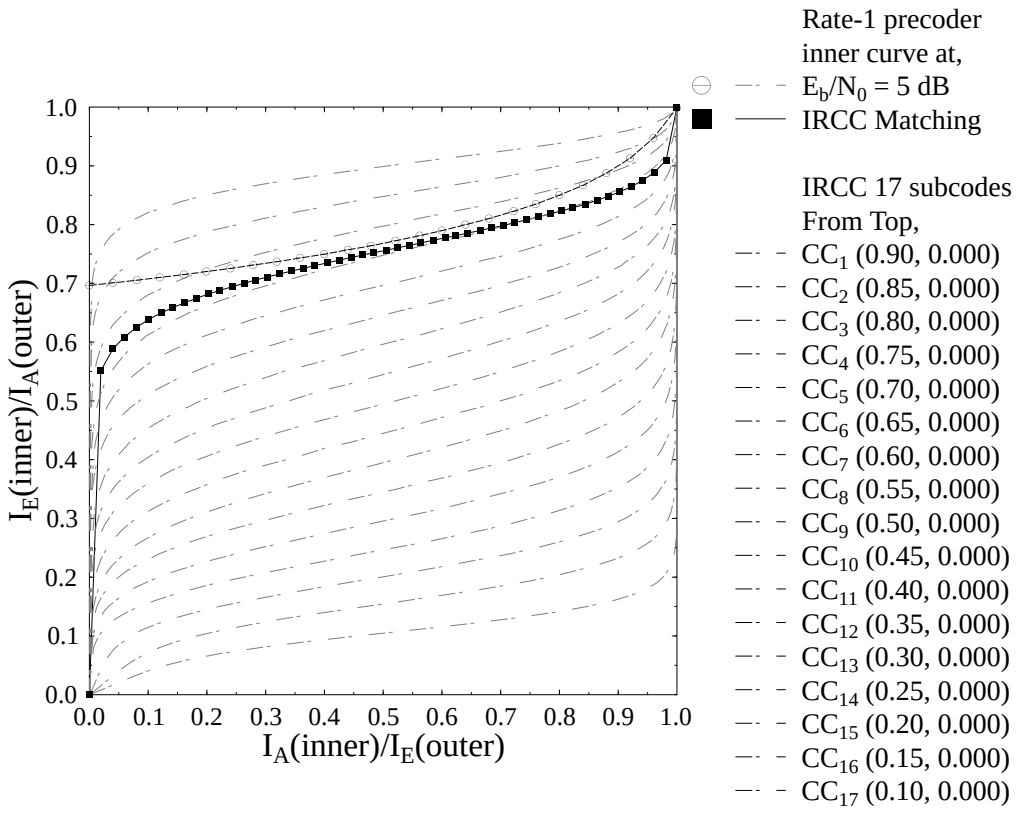

Figure 3.10: The inverted EXIT functions of 17 subcodes used in [160], along with IRCC matched EXIT curve to a BPSK-modulated Rayleigh fading channel at E_b/N_0 of 5dB.

3.4.3 Applications of Irregular Coding

IRCCs may be widely used in digital multimedia communications in order to exploit their beneficial features, including those detailed in the following sections.

3.4.3.1 Near-Capacity Operation

Near-capacity operation is of particular interest in digital wireless multimedia applications. This is because wireless channels tend to have a low channel capacity owing to their low Signal to Noise Ratio (SNR).

Shannon [180] demonstrated that if the source signal is conveyed over a noisy channel at a rate (in bits per second) that does not exceed the channel capacity, then it is theoretically possible to reconstruct it with an infinitesimally low error probability. This motivated the design of channel coding, which imposes redundancy on the source coded signal that can be utilised by the receiver to mitigate the effects of channel-induced errors. However, the computational complexity and latency of the channel codec tends to escalate, when aiming for approaching the channel capacity [181]. The discovery of irregular coding techniques [179, 182] facilitated 'near-capacity' operation at a moderate complexity.

3.4.3.2 Joint Source and Channel Coding

According to Shannon's source and channel separation theorem [180], source coding and channel coding may be performed separately without jeopardising our ability to maintain a low bit error ratio, while operating near the channel's capacity. However, Shannon's findings are only valid under his idealistic assumptions, namely that the information is transmitted over an uncorrelated non-dispersive narrowband Additive White Gaussian Noise (AWGN) channel, while tolerating a potentially infinite decoding complexity and buffering latency. Furthermore, Shannon considered lossless entropy coding, while realistic multimedia source codecs typically exploit the psycho-visual and psycho-acoustic properties of the human eye and ear, respectively. Hence, under realistic fading wireless channel conditions joint source and channel coding has substantial benefits [183].

The joint source-channel coding may be achieved using diverse methods, such as 'channel-optimised' source coding, the exploitation of residual source redundancy for joint source-channel decoding, and the employment of IRCCs.

3.4.3.3 Unequal Error Protection

The multimedia source coded bit-stream consists of various coding elements, with different degree of importance. Therefore, these different coding elements can be protected by applying various error protection schemes, depending on their level of importance. This results in more secure transmission of important information, relative to less important coding elements, and hence improves the overall performance in terms of the perceived multimedia signal quality at the receiver in comparison to EEP schemes. The IRCCs have an innate UEP capability, due to the employment of various channel code rates for the different portions of the coded bit-stream.

As described in Section 3.8, the H.264 AVC coded stream consist of three different partitions which are associated with different levels of importance. In the H.264 video stream, partition A contains the most important information. Similarly, the information contained in partition B and C has its own significance. If partition B of a slice is present along with A, the intra-frame MB update is activated for the reconstructed frame, while if partition C is present, the MCER of a slice is reconstructed and added to the motion compensated slice [77]. Recovery from inter-frame error propagation can only be achieved by encoding certain image regions in intra-frame mode by switching off inter-frame prediction for certain MBs, when no feedback channel is available. Typically a limited fraction of the MBs is encoded in the intra-frame coding mode, so that partition B typically hosts the lowest number of bits in an encoded slice.

By concatenating the three resultant partitions of each video frame a specific order depending on the error protection scheme applied, we can control their relative error protection with the aid of IRCC coding, which provides a stronger error protection for the concatenated bit-stream portion coded by lower-rate subcodes relative to the bit-stream portion coded by the higher-rate subcodes. This methodology will be utilised in order to improve the performance of H.264/AVC coded video, when communicating over correlated narrowband Rayleigh fadding channels.

3.5 Input Video Source

The video source signal provided as input for our proposed systems is encoded using the JM 13.2 encoder of the H.264/AVC standard. The *Akiyo* video test sequence of 45 frames represented in the Quarter Common Intermediate Format (QCIF) consisting of (176×144) pixel resolution was encoded at 15 frames-per-second (fps) at a target bit-rate of 64 *Kbps*. In order to exploit frame slicing based partitioning, each frame was partitioned into 11-Macro-Block (MB) slices, resulting in 9 slices per QCIF frame as shown in Figure 3.11. Two consecutive intra coded 'I' frames followed each other after 44 predicted or 'P' frames, which curtailed error propagation beyond the 45-frame boundary, i.e. beyond 3 seconds. Additional source codec parameters include the employment of quarter-pixel motion estimation, intra-frame MB update and the use of Universal Variable Length Coding (UVLC) [184] based entropy coding. Furthermore, the motion search was restricted to the immediately preceding QCIF frame in order to reduce the computational complexity of the video decoder. Similarly, the employment of the Flexible Macro-block Ordering (FMO) scheme of Section 2.4 was turned off, because despite its substantial increase in computational complexity it typically resulted in modest video performance improvements in low-motion video-telephony, using the "Akiyo" video test sequence. This allowed us to keep the encoder complexity realistic for real-time implementations. Similarly, only one previous frame was used for inter-frame motion search, which results in a reduced computational complexity as compared to using five previous reference frames.

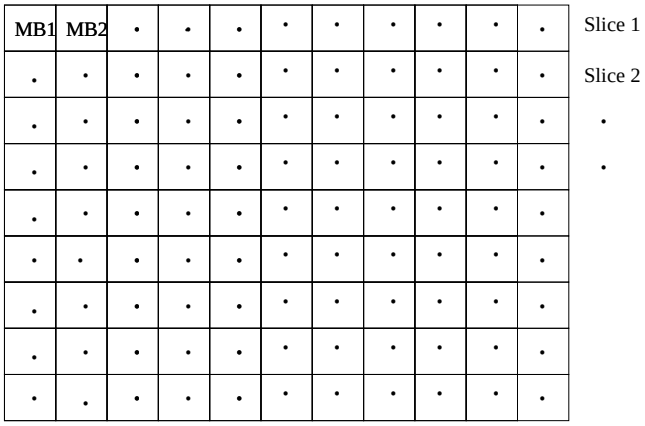

Figure 3.11: MB and Slice structure of a frame.

3.6 Hierarchical Structure of the H.264 Coded Video Stream

The hierarchical structure of different coded elements in the H.264 coded video stream is shown in Figure 3.12. In the H.264 standard the video signal is coded as a repetitive sequence of different types of frames with intra-coded 'I' frames at the boundaries of these encoded streams, which are referred to as Group of Pictures (GOP). Each frame within a GOP is further divided into a sequence of slices, each of which consists of an integer number of Macro-Blocks (MBs). The MB itself has 16×16 pixels', constituted by 4 blocks of 8×8 pixels, as shown in Figure 3.12.

3.7 The H.264/AVC Bit-Stream Syntax

The bit-stream syntax of the H.264 video codec is pictorally presented in Figure 3.13. It is shown in Figure 3.13 that the bit-stream of the H.264 video codec consists of various coding elements, such as the Start Code Prefix, the Information Byte, the Sequence Parameter Set and the Picture Parameter Set. These different coding elements are briefly summarized below:

♢ Start Code Prefix:
It is a unique sequence of three or four bytes, which is used as a prefix to indicate the start of a NAL unit as shown in Figure 3.13. It is used by the decoder to identify the start of a new NAL unit and the end of the previous NAL unit. The first NAL unit of a frame is represented by the frame start code of four-byte 00 00 00 01, while the NALU start code of 00 00 01 is used to identify the remaining NAL units within the frame.

♢ Information Byte:
It is an 8-bit sequence containing important information about the NAL unit. It contains three fields, which are the *forbidden_zero_bit*, the *nal_ref_idc* and the *nal_unit_type* as presented in Figure 3.13. The *forbidden_zero_bit* shall be equal to zero in the H.264 byte stream. The *nal_ref_idc* is identified by the bit 2-4 segment of the *info_byte*. The *nal_ref_idc* field being not equal to zero indicates that the NAL unit

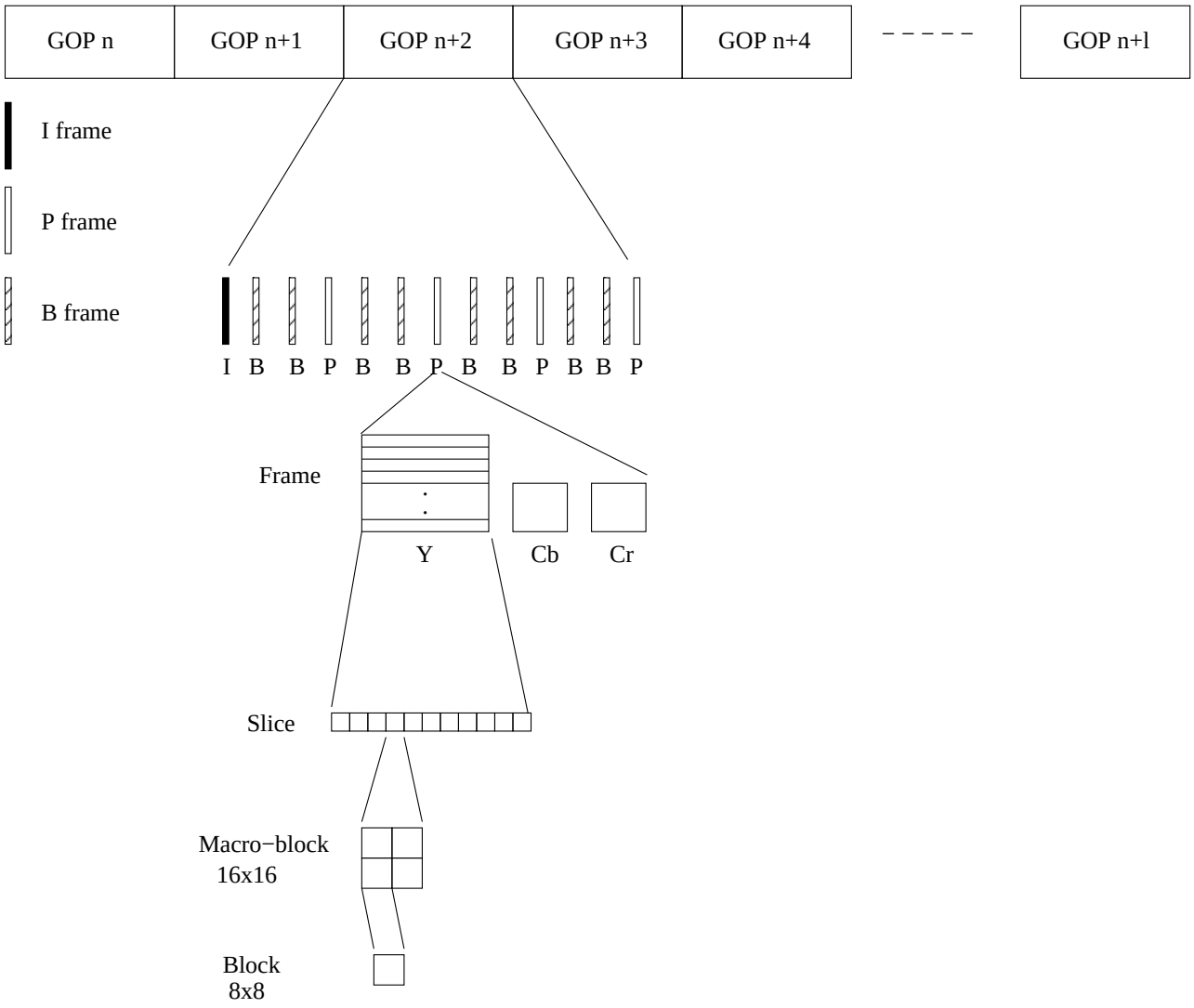

Figure 3.12: The H.264/AVC coded elements, hierarchical structure.

contains a sequence parameter set or a picture parameter set or a slice of the reference frame. By contrast, the *nal_ref_idc* field being equal to zero indicates that the NAL unit contains a slice of a non-reference picture. Furthermore, the *nal_ref_idc* field shall not be equal to zero for the Instantaneous Decoding Refresh (IDR) NAL unit. The *nal_unit_type* is represented by the bit-segment 5-8 of the *info_byte* field. The coded slice of a frame is represented by *nal_unit_type*=1, while 2, 3 and 4 is used to identify Partition A, B and C of the DP H.264 byte stream. Additionally, the coded slice of the IDR picture is represented by the field *nal_unit_type*=5, while 7 and 8 represent sequence and picture parameter sets, respectively.

♢ Sequence Parameter Set:
Contains syntax elements, which apply to an entire GOP. Each sequence parameter set is identified with the aid of a specific *sequence_parameter_set_ID* syntax element, found in the picture parameter set shown in Figure 3.13.

♢ Picture Parameter Set:
Contains syntax elements, which apply to one or more coded pictures. Each picture parameter set is

identified by a specific *picture_parameter_set_ID* shown in Figure 3.13 and is referred to by this syntax element in the slice header.

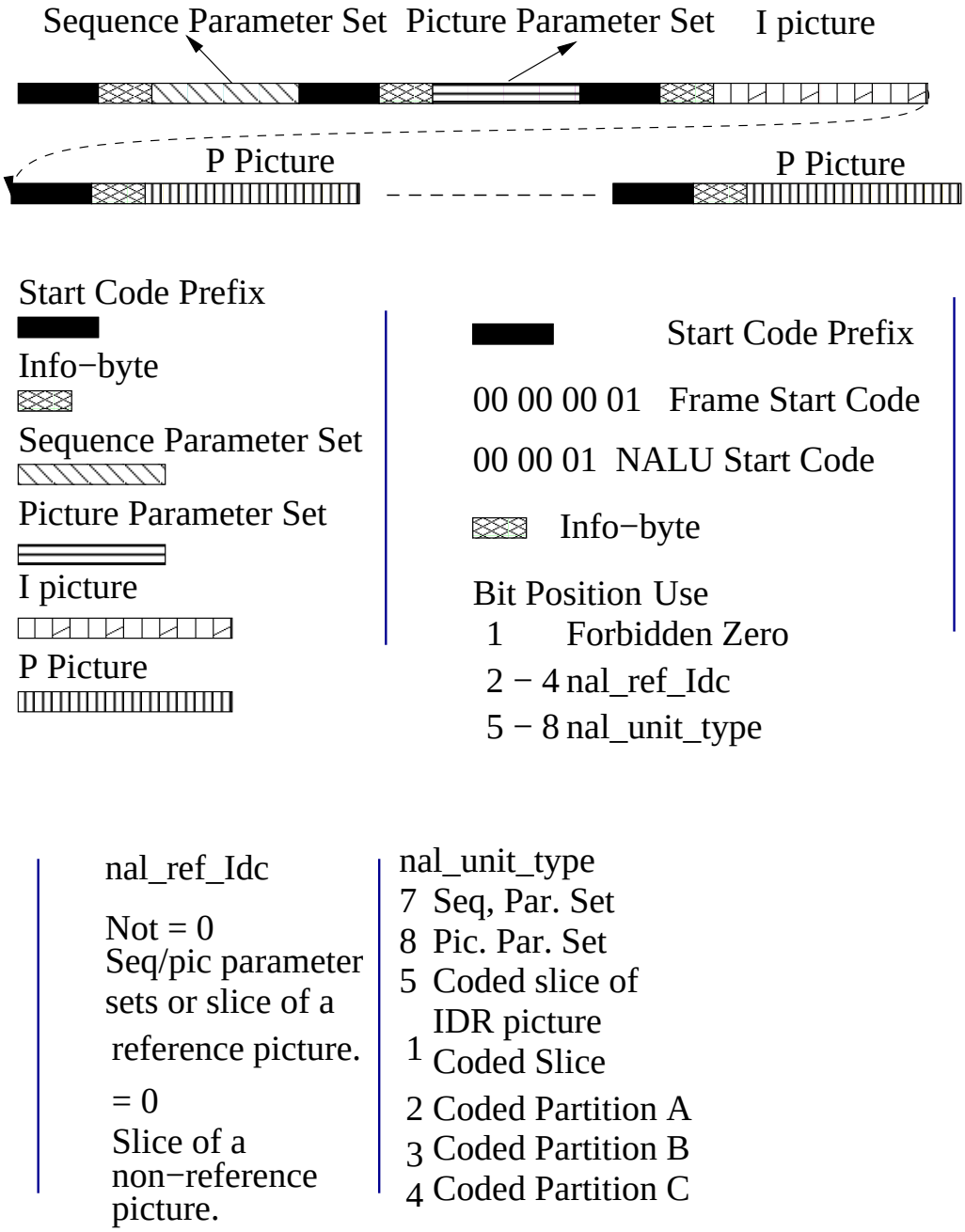

Figure 3.13: Bit-stream syntax of H.264/AVC.

3.8 H.264 Data-Partitioning

In addition to providing a high video compression, the H.264/AVC video codec employs a range of techniques designed for achieving an enhanced error resilience [81]. One of these features is DP [112]. Instead of the classic approach of coding all video parameters of a Macro-Block (MB) into a single bit-string represent-

ing a video-slice, the H.264 DP may generate three bit-strings per slice referred to as partitions, and hosting different-sensitivity classes, as presented in Figure 3.14. The advantage of this method is that it separates the bits representing for example the Motion-Vectors (MVs) and other subjectively important high-sensitivity video parameters potentially inflicting prolonged error propagation from the less sensitive motion compensated residual. This provides us with the ability to protect the different sensitivity classes based on their relative importance. In the H.264/AVC scheme three different partions are used, namely types A, B, and C, as shown in the Figure 3.14. The detailed description of these three partitions is described below;

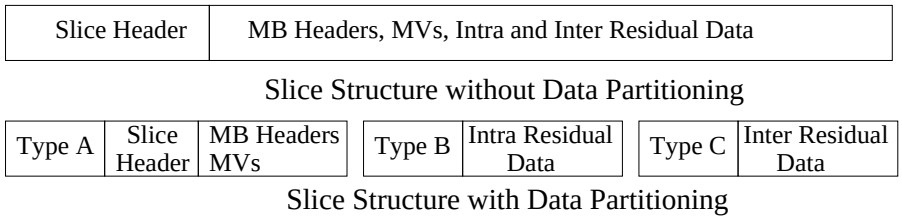

Figure 3.14: Block Diagram of H.264 Slice with and without DP [89]

- Type A contains the most vulnerable bits representing the slice header information, MB types, quantisation parameters and MVs, as shown in the Figure 3.14. The bits of an entire video slice including partitions B and C are dropped, when partition A is deemed to be corrupted. In this scenario the entire slice is marked as corrupted and the decoder will apply error concealment techniques using the corresponding video segment of the previously decoded frame [76]. In case of low-motion video clips, such as the "Akiyo" sequence, the error concealment techniques are typically quite effective.

- Type B partitions typically contain intra-frame coded MB coefficients and intra-MB Coded Block Patterns (CBP) bits, indicating which blocks within a MB contain non-zero transform coded coefficients, as shown in the Figure 3.14. Recovery from error propagation can only be achieved by encoding image regions in intra-frame mode, i.e., without reference to previously coded frames, by switching off intra-frame prediction for certain MBs, when no feedback channel is available. The error propagation would then be curtailed, because the MBs are independent of past frames. Therefore, the intra-frame coded MB updates and hence their encoded bits tend to be more important than inter-frame coded MBs. Corruption of partition B implies that no intra-frame coded MB update will take place during the current reconstructed frame. The absence of intra-coded MB updates potentially also results in further error propagation to the remaining P frames in a GOP. However, the price paid for the enhanced robustness achieved by intra-frame coding is the increased bit-rate compared to inter-frame prediction. Using a high number of intra-frame coded MBs results in a significant degradation of the achievable compression. Therefore, using channel-quality-aided switching between intra-frame coding and inter-frame coding to achieve the right balance between compression efficiency and robustness is important. Typically a limited fraction of the MBs is encoded in the intra-frame coding mode, so that partition B typically hosts the lowest number of bits in an encoded slice.

- Type C partitions carry inter-frame CBP and inter-frame Motion Compensated Error Residual (MCER) bits for MBs encoded using motion compensated prediction, as well as intra-frame CBP and intra-frame

MCER bits for the MBs coded using the H.264-specific intra-frame prediction mode, as shown in the Figure 3.14. The inter-frame MCER and intra-frame MCER bits tend to be the least important. In case of low-motion videophone sequences, typically there is significant correlation among the consecutive video frames and hence the loss of partition C usually does not result in a significant Peak Signal-to-Noise Ratio (PSNR) degradation, even if the lost video-tile is replaced by the coresponding tile of the previously decoded frame.

Therefore, in the H.264 video stream partition A is the most important one, MVs and header information. The specific significance of a partition B and C is dependent on that of partition A, but they are not decodable in the absence of partition A. Again, if partition A is dropped, the entire slice including partitions B and C will be dropped and the corresponding slice will be marked as corrupted. Otherwise, if partition A of a slice is correct, then the header and MV information bits of partition A are decoded and are used in the motion compensation process. If partition B of a slice is present along with A, the intra-frame MB update is added to the reconstructed frame, while if partition C is present, the MCER of a slice is reconstructed and added to the motion compensated slice [77].

3.9 H.264 Error Sensitivity Study

In addition to providing a high video compression efficiency, the H.264/AVC video codec employs a range of techniques designed for achieving an enhanced error resilience [81]. One of these features is DP [112]. The H.264/AVC video frame format consists of macroblocks, which are encoded by a number of sub-bit-strings referred to as slices. Accordingly, when using DP, these slices are sub-divided into more than one bit-string per slice, which are called partitions. In H.264/AVC three different types of partions are used, namely type A, type B, and type C, each containing a specific set of slice symbols [81], as described in Section 3.8. The Slice Error Sensitivity (SES) of the H.264/AVC bit-stream has to be evaluated, before employing UEP. For this reason we used the 45-frame ”Akiyo” sequence [1] consisting of the first intra-coded ’I’ Frame followed by 44 predicted ’P’ Frames. The target bit-rate was 64 Kbps and we systematically dropped partition A, B and C, one at a time in the first four ”P” frames of the 45-frame (176x144)-pixel QCIF resolution ”Akiyo” video sequence. The resultant PSNR degradation was recorded for each decoded frame under the following two conditions. Firstly, while quantifying the sensitivity of the codec dropping partitions A, B or C, the corresponding partition was dropped from the frame and the resultant PSNR degradation was recorded for the full sequence. Secondly, the above process is performed separately for each partition of type A, B and C, in the first four ’P’ frames of the sequence. The number of partitions and their sizes expressed in terms of the number of bits for the first four ’P’ frames is shown in Table 3.6

3.9.1 Error Sensitivity of Partition A

The objective PSNR degradation caused by the systematic corruption of partition A in the first four frames is shown in Figure 3.15. It is observed from the figure that 2 out of 36 partitions result in considerable PSNR degradation, while one of the partitions has a PSNR degradation, which is only slightly above 1 dB. All the

Table 3.6: Number of partitions and their Sizes in bits

Frame #	Partition type: A		Partition type: B		Partition type: C	
	# of partitions	# of Bits	# of partitions	Size	# of partitions	# of Bits
1	9	1152	3	856	9	48
2	9	1624	2	608	9	2128
3	9	1904	3	1152	9	3664
4	9	1336	3	952	9	720
Total	36	6016	11	3568	36	6560

remaining partitions result in negligible PSNR degradations. The whole slice, including partitions B and C is dropped as a result of corrupting Partition A. The slice is marked as corrupted and the decoder will apply regular error concealment using the previous frames. The error concealment techniques are normally effective in case of low-motion video scenes like ”Akiyo” and result in a lower PSNR degradation in the decoded video sequence.

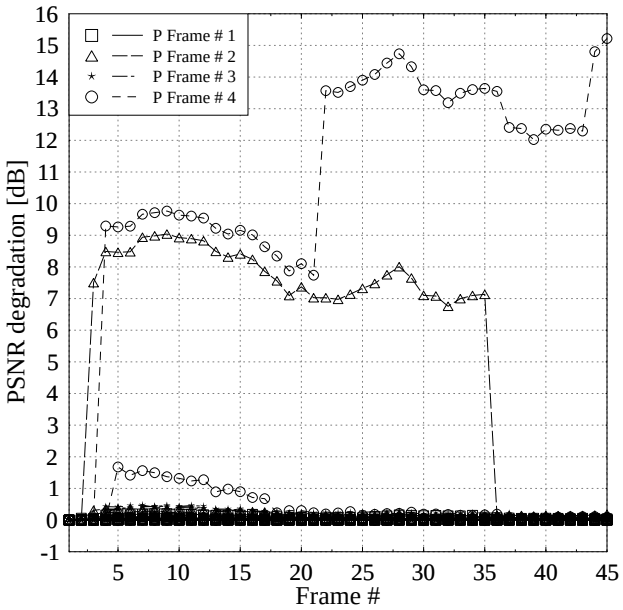

Figure 3.15: H.264 Partition A Error Sensitivity

Figure 3.15 presents PSNR degradation caused due to the systematic one by one corruption of 36 different type A partitions, presented in Table 3.6. Out of 36 partitions 33 partitions have PSNR degradation lower than 1 dB, as shown in Figure 3.15.

The resultant artifacts imposed by the corruption of the most important slice (6th) in the first P frame is circled in Figure 3.16.

Figure 3.16: Artifacts due to Slice A Corruption

3.9.2 Error Sensitivity of Partition B

The objective PSNR degradation caused by the systematic corruption of partition B is shown in Figure 3.17. It becomes clear from the Figure 3.17 that 6 out of 11 partitions result in considerable PSNR degradations while the remaining have PSNR degradations below 1 dB. Corruption of Partition B means that no intra MB update will be incorporated in the reconstructed frame. The absence of intra MB update results in the inability to stop error propagation to the remaining P frames in a GOP, as imposed bit errors in the received coded bit-stream. The resultant artifacts due to the corruption in the 3rd B partition of the 1st P frame is circled, in Figure 3.18

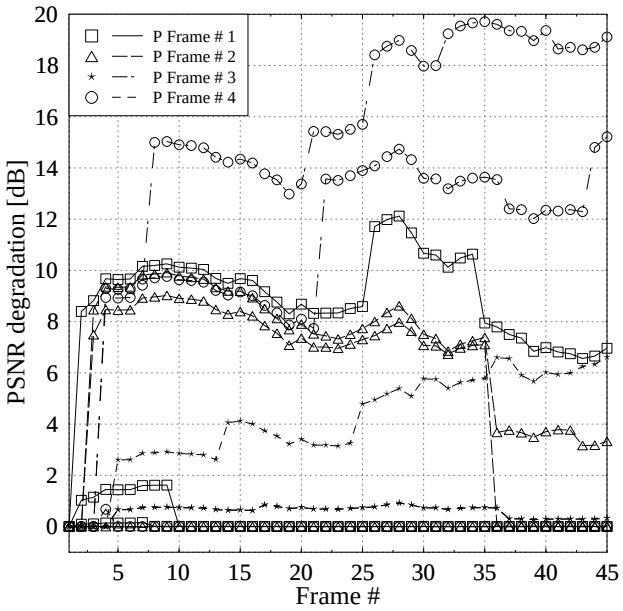

Figure 3.17: H.264 Partition B Error Sensitivity

and Figure 3.19.

Figure 3.18: Artifacts due to Slice B Corruption

Figure 3.19: Artifacts due to Slice B Corruption

3.9.3 Error Sensitivity of Partition C

Observe from Figure 3.20 that only 1 out of 36 type C partitions has a noticeable PSNR degradation, while the remaining partitions have a low PSNR degradation. Partition C contains inter-MB difference information. In case of low-motion video videophone sequence, typically there is significant correlation among consecutive frames and the loss of partition C usually does not result in significant PSNR degradation. The artifacts generated due to the most important 5th C partition in the 3rd P frame is circled in Figure 3.21 and Figure 3.22 below,

Therefore, in the H.264 video stream, partition A is the important type of partition containing MVs and header information. Partition B and C are dependent on partition A and are not decodable in the absence of partition A. If partition A is dropped, the whole slice including B and C will be dropped and the corresponding block will be marked as corrupted. Otherwise if partition A is correct, then the header and motion vector information in Partition A is decoded and is used in the motion compensation process. If partition B is present along with A, the intra MB update is incorporated in the reconstructed frame. By contrast, if Partition C is present, the displaced frame MB difference is reconstructed and added to the motion compensated frame [77].

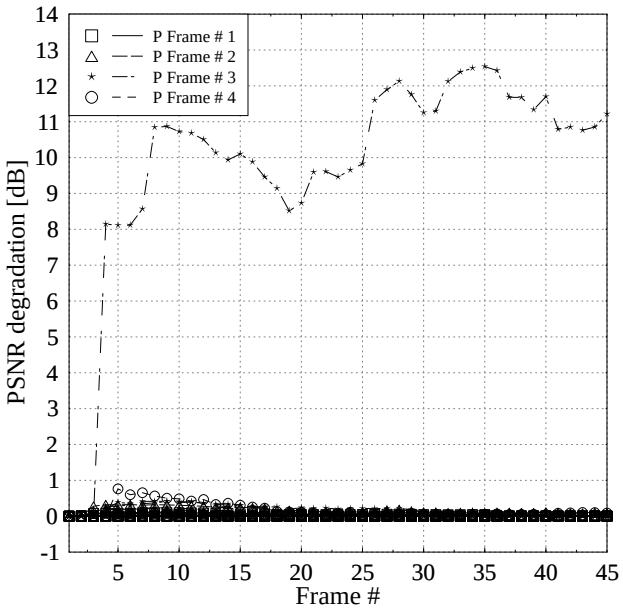

Figure 3.20: H.264 Partition C Error Sensitivity

Figure 3.21: Artifacts due to Slice C Corruption

3.10 Unequal Error Protection Using RSC Codes

In Section 3.7 we explained that the H.264 coded bit-stream contains different coding elements having different levels of importance. Similarly, the DP concept of H.264 is presented in Section 3.8. Additionally, it was shown in Section 3.9 that the H.264 coded bit-partitions have different sensitivity to channel-induced errors. Therefore, by appropriately allocating the available bit-rate budget to the different portions of the bit-stream, UEP can be provided, which may result in a performance improvement for the system relative to the conventional equal protection scheme. In this system configuration UEP is provided for H.264 coded video stream by applying different-rate RSC codes to the three different types of H.264 bit-stream partitions.

Figure 3.22: Artifacts due to Slice C Corruption

3.10.1 System Overview

The schematic of the proposed video coding and transmission arrangement is shown in Figure 3.23. At the transmitter side, the video sequence is compressed using the H.264/AVC video codec employing the specific codec parameters described in Section 3.5. Then, the output bit-stream of Figure 3.23 is de-multiplexed into three different bit-streams, namely Steam A, Stream B and Stream C, containing the sequentially concatenated partitions of type A, B and C of all the slices per frame, respectively. The de-multiplexer's binary output sequences x_a, x_b, and x_c, where we have a = 1,2, ..., b_a, b = 1,2, ..., b_b, c = 1,2, ..., b_c, and $B = b_a+b_b+b_c$, are then interleaved using the bit-interleavers Π of Figure 3.23, yielding the interleaved sequences $\bar{x}_a$, $\bar{x}_b$ and $\bar{x}_c$ and are then encoded by RSC codes having different code-rates. Since the degree of the statistical independence guaranteed by an interleaver is always related to its length [185], concatenation of the bits generated by the MBs of a slice within a given partition results in a longer interleaver without extending the video delay and hence improves the achievable performance of iterative decoding. The unequal error protected and partitioned H.264/AVC bit-stream y_i of Figure 3.23 is QPSK modulated and transmitted over a temporally correlated narrowband Rayleigh fading channel, associated with a normalised Doppler frequency of $f_D = f_d T_s = 0.01$, where f_d is Doppler frequency and T_s is the symbol duration. The Doppler frequency of $f_D = 0.01$ represents a relatively slowly fading channel and is commonly used in the literature to characterise the performance of multimedia transmission systems. Increasing the Doppler frequency, i.e. the vehicular speed may degrade the achievable modem performance but typically improves the channel codec's performance, since uncorrelated fading would only be achieved by an infinite interleaver length, which would near-perfectly randomise the occurance of bit errors. For example, for a pedestrian speed of 3 miles per hour and a carrier frequency of 1GHz the Doppler frequency is about 4Hz. At the video encoding rate of 64Kbps and assuming $1/3$-rate FEC coding the total bit rates become 192Kbps, and when assuming 2-bits per symbol QPSK signalling this results in a symbol rate of 96 Kilo-symbols per second (K symbols/s) or a symbol duration of 0.0104 seconds. Hence $f_D = f_d T_s = 4 \times 0.0104 = 0.0416$ and therefore the assumption of $f_d = 0.01$ is a conservative assumption for video transmission system performance evaluation.

The received signal is QPSK demodulated and the soft information is passed from the QPSK demapper to the respective RSC decoder after de-multiplexing the LLRs into three streams corresponding to the partitions

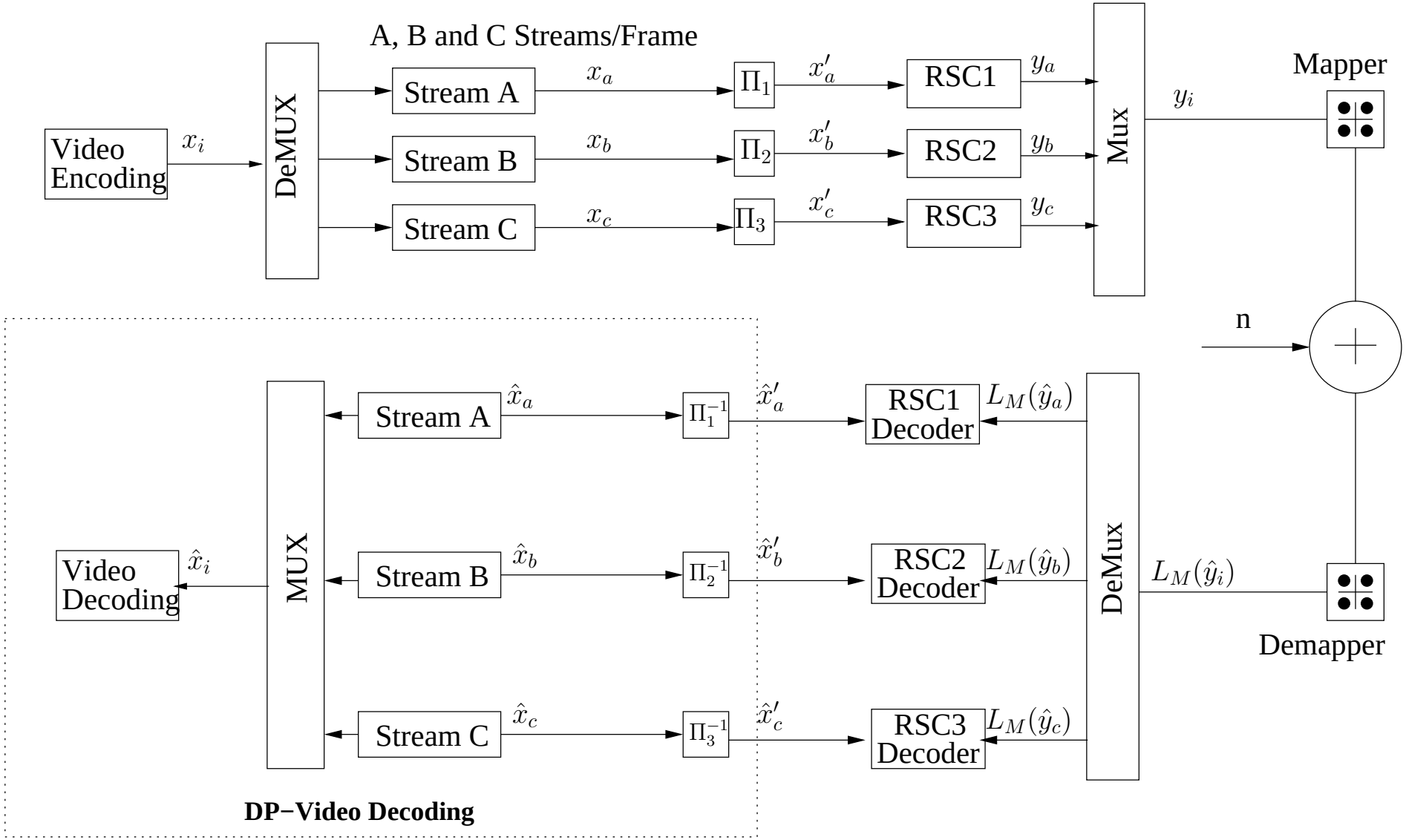

Figure 3.23: System-1: DP H.264 UEP with RSC codes using Rayleigh fading transmission channel. All system parameters were summarised in Table 3.7.

A, B and C. After RSC decoding the resultant bit-streams are de-interleaved using the respective de-interleavers Figure 3.23. Subsequently, the de-interleaved bit-streams are multiplexed into their original input order and are decoded by the H.264 decoder to produce the decoded video. The decoded video sequence is then used to calculate the $PSNR$ performance relative to the input video sequence. The different RSC coding rates applied using Equal Error Protection (EEP) and UEP schemes are shown below in Table 3.7(a). The performance of the

Error Protection Scheme	Code Rate			
	A	B	C	Overall
EEP	1/2	1/2	1/2	1/2
UEP1	1/3	2/3	3/4	1/2
UEP2	1/2	1/3	2/3	1/2

(a) Code Rates for Different Error Protection schemes

System Parameters	**Value**	**System Parameters**	**Value**
Source Coding	H.264/AVC	Video Bit Rate (Kbps)	64
Source Codec	JM 12.1	Video Frame Rate (fps)	15
Video Test Sequence	*Akiyo*	Channel Coded Rate (Kbps)	128
No of Frames	45	Baud-rate (Kbps)	64
Frame Format	QCIF	Channel Coding	RSC
No of 'P' frames between	44	Over-all Code Rate	1/2
two 'I' frames		Code Memory	3
Time Period of 'I' frame (sec)	3	Generator Polynomials	
Use of 1/4-Pixel Motion Estimation	Yes	RSC 1/2,2/3 & 3/4 (G_1, G_2)	$(5,7)_8$
Intra-frame MB update/frame	3	RSC 1/3 (G_1, G_2, G_3)	$(3,7,5)_8$
per QCIF frame		Modulation Scheme	QPSK
Use of 'B' Pictures	No	Number of Transmitters, N_t	1
Use of FMO	No	Number of Receivers, N_r	1
No of Frames Used for	1-Frame	Channel	Correlated Rayleigh Fading
Inter-Frame Motion Search		Normalised Doppler Frequency	0.01
No of Video Slices/frame	9	Interleaver Length	$\approx (64000/15)$
No of Video MBs/Slice	11	No System Iterations I_t	10

(b) Systems parameters used in the schematic of Figure 3.23

Table 3.7: Code Rates and systems parameters used in the schematic of Figure 3.23

system is evaluated by keeping the same overall system code rate for the EEP and UEP coding schemes. In the UEP1 scheme of Table 3.7(a) the highest grade of error protection is provided for Partition A, while in UEP2 the highest code rate convolutional coding is applied to Partition B in order to protect it more strongly against channel errors compared to Partition A and C, because based on our SES results of Section 3.9, its observed that Partition B is also important and in case of its corruption it results in significant PSNR degradation because of error propagation to neighbouring blocks in both the spatial and temporal domains. Furthermore, Partition B has the lowest overall size, therefore its strong protection does not results in a significant increase in the over all system bit-rate. As a result, a reasonable protection can still be provided for Partitions A and C within a

given channel-coded bit rate budget.

3.10.2 Performance Results

The overall BER versus E_b/N_0 performance of the system is presented in Figure 3.24. It can be observed from Figure 3.24 that the three different error protection schemes result in a similar BER performance owing to their identical overall system code-rate. Additionally, the BER vs E_b/N_0 curves recorded for the individual partitions of A, B and C are portrayed in Figure 3.25. It is clear from Figure 3.25, that the EEP scheme has the same BER for the A, B and C partitions, while the UEP1 and UEP2 schemes of Table 3.7(a) have a BER performance, which is dependent on the code rate given in Table 3.7(a) for each type of partition. Furthermore, the objective video quality of the three error protection schemes is characterised in Figure 3.26, which shows that UEP2 results in a better performance than the EEP scheme, which is in turn better than that of the UEP1 scheme.

In contrast to the UEP1 scheme, where partition A had the strongest protection, the rationale of this UEP2 arrangement is, that in low-motion video clips the video slices corrupted owing to the loss of partition A are concealed using the corresponding video-tile of the previously decoded frame, since this method results in less annoying artefacts owing to their high correlation with the adjacent frames. Additionally, reference to these concealed slices by future frames using motion compensated prediction results in adequate video reconstruction results. However, the loss of partition B along with a correctly received partition A of a particular slice implies that the MV and header information received in partition A will be used to decode the MBs of this slice encoded either in intra- or inter-frame prediction mode, while the all-important MBs of partition B represented as intra-frame MBs were completely lost. Furthermore, the future frames, and MBs referring to the lost intra-coded MBs of partition B in predictive coding will also result in a low video quality. Hence, providing high protection for partition B increases its probability to be correctly received, hence facilitating the reconstruction of every correctly received partition A. Therefore, similarly to partition A, partition B is also important and hence its corruption results in significant PSNR degradation of the decoded video sequence owing to the resultant avalanche-like error propagation, when the intra-frame coded MB updates are corrupted as can be observed from the PSNR versus E_b/N_0 curves of Figure 3.26.

It is also important to note that for the sake of maintaining a high confidence level we ran each experiment 160 times and recorded the average results. Explicitly, an E_b/N_0 gain of 4 dB is attained using UEP1 with reference to UEP2, while an E_b/N_0 gain of 3 dB is achieved with reference to EEP scheme at the PSNR degradation point of 2 dB. The major findings related to the different error protection schemes recorded at the PSNR degradation points of 1dB and 2dB are summarized in Table 3.8.

Table 3.8: Performance Summary Table

Schematic	Parameters	Iterations	Video rate	FEC Code rate	Highest PSNR value	E_b/N_0 at 1dB PSNR degradation	E_b/N_0 at 2dB PSNR degradation
Figure 3.23	Table 3.7(a)	10	64 $kbps$	0.5	41.7582	EEP = 36 dB UEP1 = 38 dB UEP2 = 34 dB	EEP = 34 dB UEP1 = 35 dB UEP2 = 31 dB

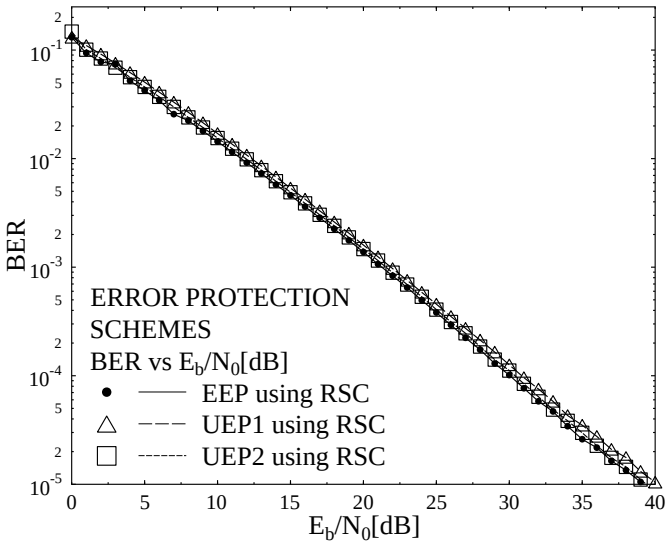

Figure 3.24: BER versus E_b/N_0 for the EEP and UEP schemes of Figure 3.23 using the parameters of Table 3.7.

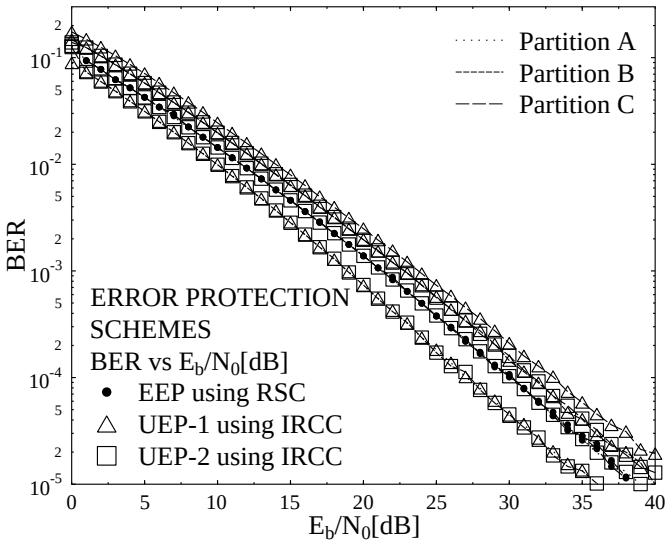

Figure 3.25: BER (A, B and C) versus E_b/N_0 for the EEP and UEP schemes of Figure 3.23 using the parameters of Table 3.7.

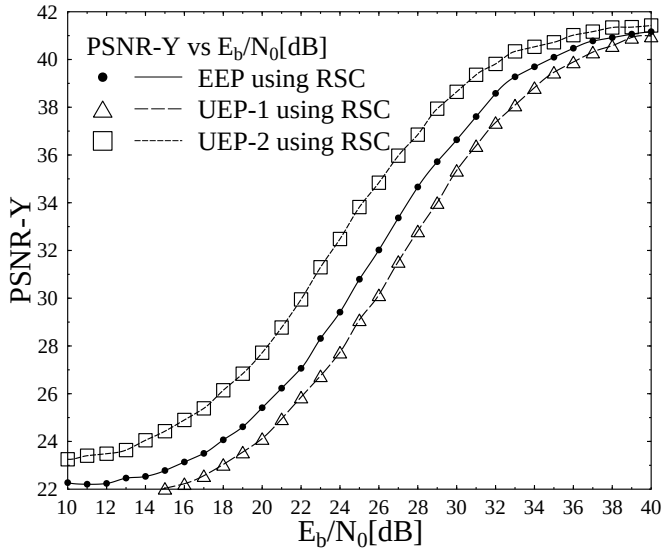

Figure 3.26: PSNR versus E_b/N_0 for the EEP and UEP schemes of Figure 3.23 using the parameters of Table 3.7.

3.11 Iterative Detection Aided H.264 Wireless Video Telephony Using IRCC

Multimedia source coded information, such as speech, audio and video typically exhibit unequal sensitivity to channel errors [160, 186, 187]. Therefore, UEP is used by exploiting the innate UEP capability of IRCCs due to the employment of various constituent codes having different coding rates to ensure that the perceptually more important bits suffer from a lower number of channel errors. The construction of IRCCs was studied by Tuchler and and Hagenauer [170], where a family of different convolutional codes having various code-rates was designed with the aid of EXIT charts for achieving an improved iterative decoding behaviour for a serially concatenated system.

In this section, we will characterise the benefits of employing IRCCs in the iterative source-channel decoding process. In contrast to the regular convolutional codes, which encode the entire segment of the source signal using the same codes, the IRCCs introduced in Section 3.4.2 encode the source signal by splitting it into segments having specifically designed lengths, each of which is encoded by a code having an appropriately designed code-rate.

In order to demonstrate the ability of IRCCs schemes to operate near the capacity, while facilitating UEP, in the next subsection we provide a design example.

3.11.1 System Overview

The schematic of the proposed videophone arrangement is shown in Figure 3.27. The video source signal detailed in Section 3.5 is provided as input to the proposed system. At the transmitter side, the video sequence

is compressed using the H.264/AVC video codec. Then, the output bit-stream representing a video frame consisting of K source coded bits x_k, $k = 1, 2, \ldots K$, is de-multiplexed into three different bit-streams, namely Stream A, Stream B and Stream C, containing the sequentially concatenated partitions of type A, B and C of all the slices per frame, respectively. The resultant binary output sequences x_a, x_b, and x_c, where we have $a = 1, 2, \ldots A$, $b = 1, 2, \ldots B$ and $c = 1, 2, \ldots C$, and $K = A+B+C$, are then concatenated according to a specific order, depending on the choice of the error protection scheme applied to the bit-string x_i. Subsequently, the concatenated bit-string is channel coded using an IRCC having a specific code rate, hence generating the coded string x'_i. The coded string x'_i is then interleaved using the bit-interleaver Π of Figure 3.27, into the interleaved sequence $\bar{x}_i$, which is then encoded by the rate-1 precoder of Figure 3.27 before transmission. In the iterative decoder of Figure 3.27, the extent of the statistical independence of the extrinsic information provided by an interleaver is always related to its length [185]. Therefore, instead of independently performing the iterative decoding operation on the various frame slices, we integrated all the bits generated by the MBs of the slices of each partition within a given video frame. The integrated bit-strings of each partition are then subsequently concatenated into a single string, which results in a longer interleaver and hence improves the attainable iterative decoding performance without extending the video delay. The precoded bit-stream y_i is QPSK modulated and transmitted over a temporally correlated narrowband Rayleigh fading channel, associated with the normalised Doppler frequency of $f_d = f_D T_s = 0.01$, where f_D is the Doppler frequency and T_s is the symbol duration.

At the receiver the soft-information obtained after QPSK demodulation is forwarded to the inner decoder. The extracted extrinsic information is then exchanged between the inner and outer decoder of Figure 3.27, in order to attain the lowest possible BER [188]. At the receiver the soft-information obtained after QPSK demodulation in the form of its LLR representation $D_{M,a}$ is forwarded to the rate-1 precoder. The rate-1 precoder's decoder processes this input information and the *a priori* information $L_{M,a}$ is fed back from the outer decoder of Figure 3.27 in order to generate the *a posteriori* LLR values $L_{M,p}$. The rate-1 precoder's *a priori* LLR values $L_{M,a}$ are subtracted from its *a posteriori* LLR values $L_{M,p}$ to generate the extrinsic LLR values $L_{M,e}$, which are subsequently deinterleaved by the soft-bit interleaver of Figure 3.27, yielding $L_{D,a}$. Then, the soft bits $L_{D,a}$ are passed to the convolutional decoder, which uses the Log MAP algorithm of [168] to compute the *a posteriori* LLR values $L_{D,p}$. Observe in Figure 3.27 that $L_{M,a}$ is the interleaved version of $L_{D,e}$, which in turn is generated by subtracting the *a priori* information $L_{D,a}$ from the *a posteriori* information $L_{D,p}$. During iterative decoding the precoder's decoder exploits the *a priori* information for the sake of providing improved *a posteriori* LLR values for the outer channel decoder of Figure 3.27, which in turn exploits the input LLR values for the sake of providing improved *a priori* information for the precoder in the subsequent iteration. Further details of iterative decoding are provided in [137].

3.11.2 IrRegular Convolutional Code Design

The details of IRCC design were provided in Section 3.4.2 [170]. More explicitly an IRCC consisting of $'I'_n$ constituent subcodes is constructed by first selecting a convolutional mother code C_1 with rate-r_1. The remaining $[I_n - 1]$ subcodes C_i of rate-$r_i > r_1$, where we have $i = 2 \cdots I_n$, are obtained by puncturing, while subcodes C_i with rate-$r_i < r_1$ are created by adding more generators and by puncturing. In IRCCs K

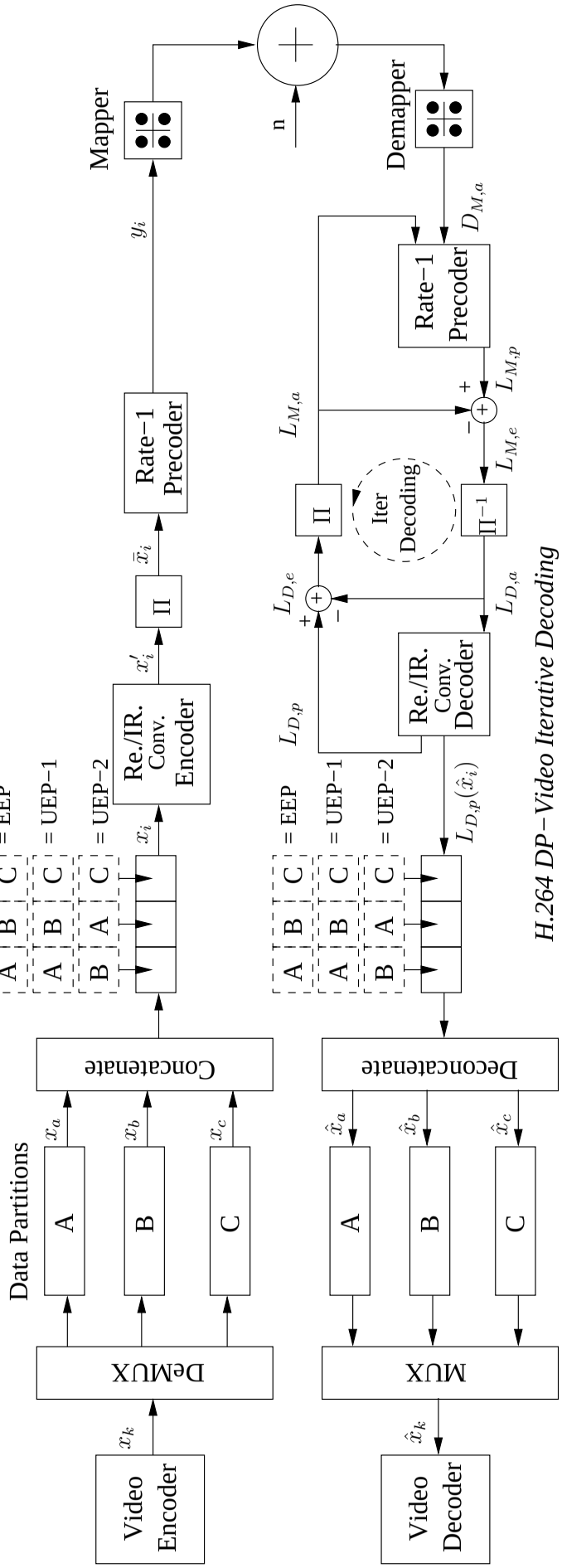

Figure 3.27: The Proposed Iterative Source-Channel Decoding System Model. All system parameters were summarised in Table 3.9. The difference in comparison to Figure 3.23 include the concatenation of stream A, B and C to form a single long stream. Additionally, iterative combination Rate-1 precoder and Re./IR. convolutional codes were incorporated as inner and outer component codes for iterative decoding which resulted in 0.7 overall code rate.

input bits are encoded into N coded bits, where each rate-r_i subcode encodes only a fraction $(\alpha_i r_i N)$ of the information bits and generates $(\alpha_i N)$ encoded bits, with α_i representing the relative size of a fraction. The weighting coefficients α_i, $i = 1 \ldots I_n$ and the target code rate of $R \in [0\ 1]$ should satisfy,

$$\sum_{i=1}^{I_n} \alpha_i = 1, \text{ and } \sum_{i=1}^{I_n} \alpha_i r_i = R, \text{ where } \alpha_i \in [0\ 1].$$

3.11.3 The Proposed IrRegular Convolutional Code

In our design example we used a fixed overall coding rate of $R = 0.7$, which was assigned to the outer code, because we used a unit-rate code as our inner code. As a benchmarker we used a half-rate memory-4 RSC scheme as our outer code defined by the generator polynomial $(1, g_1/g_0)$, where $g_0 = 1 + D + D^4$ and $g_1 = 1 + D^2 + D^3 + D^4$ are the feedback and feedforward polynomials, respectively. Then the overall RSC code rate of 0.7 was obtained by puncturing. We used the $I = 17$ constituent subcodes of [170], having code rates r_i = [0.10 0.15 0.20 0.25 0.30 0.35 0.40 0.45 0.50 0.55 0.60 0.65 0.70 0.75 0.80 0.85 0.90], respectively. We employed the EXIT chart matching algorithm of [160] and selected the target inner code transfer function recorded at $E_b/N_0 = 3\ dB$, which resulted in the $I = 17$-element weighting vector of α_i = [0.003112 0.029389 0 0 0 0 0 0 0 0 0.291014 0.657845 0 0 0.0477998 0] for the outer IRCC code. The '0' weights indicate that the corresponding codes were not activated. By concatenating the three resultant partitions per video frame into a specific order depending on the error protection scheme applied, we can control their relative error protection with the aid of IRCC coding, which provides a stronger error protection for the concatenated bit-stream portion coded by lower-rate subcodes relative to the bit-stream portion coded by the higher-rate subcodes.

3.11.4 EXIT Chart Analysis

The EXIT [159] function of both the IRCC and benchmarker RSC are shown in Figure 3.28. In addition to the beneficial UEP capability of the IRCC, it can be observed from the EXIT curves that there is a wider EXIT chart tunnel between the rate-1 inner precoder's EXIT curve and that of the outer irregular channel code's curve relative to the benchmarker scheme dispensing with the regular outer IRCC channel code. Therefore, given a limited number of affordable iterations, a relatively short interleaver and a wider EXIT tunnel, our 0.7-rate IRCC may be expected to perform better than the benchmarker scheme using the identical-rate RSC code. The actual decoding trajectories of the various error protection schemes employing the regular/irregular outer channel codes as well as using the rate-1 inner precoder detailed in Table 3.9(a) was recorded at $E_b/N_0 = 3\ dB$, $4\ dB$ and $5\ dB$, as portrayed in Figures 3.29, 3.30 and 3.31, respectively. These trajectories were plotted by acquiring the mutual information between the hypothesized and soft-bit values at the input and output of both the inner and outer decoder during the bit-by-bit Monte-Carlo simulation of the iterative decoding algorithm. It may be inferred from the EXIT trajectories of Figures 3.29, 3.30 and 3.31 that as expected, the convergence behaviour of the iterative decoding scheme was improved by the irregular outer channel codes, which provide a wider EXIT tunnel relative to the regular outer channel code, although they have the same code rates.

Protection Scheme	Concatenation order	Code Rate		
		Outer Code	Inner Code	Overall
EEP	[A-B-C]	RSC	Rate-1 Precoder	0.7
UEP1	[A-B-C]	IRCC	Rate-1 Precoder	0.7
UEP2	[B-A-C]	IRCC	Rate-1 Precoder	0.7

(a) Code Rates for Different Error Protection schemes

System Parameters	**Value**	**System Parameters**	**Value**
Source Coding	H.264/AVC	Video Bit Rate (Kbps)	64
Source Codec	JM 12.1	Video Frame Rate (fps)	15
Video Test Sequence	*Akiyo*	Channel Coded Rate (Kbps)	91.42
No of Frames	45	Baud-rate (Kbps)	45.71
Frame Format	QCIF	Channel Coding	Re./IR. conv. code
No of 'P' frames between	44	Over-all Code Rate	0.7
two 'I' frames		Code Memory	4
Time Period of 'I' frame (sec)	3	Generator Polynomials	
Use of 1/4-Pixel Motion Estimation	Yes	RSC & IRCC (G_0, G_1)	$(23,35)_8$
Intra-frame MB update/frame	3		
per QCIF frame		Modulation Scheme	QPSK
Use of 'B' Pictures	No	Number of Transmitters, N_t	1
Use of FMO	No	Number of Receivers, N_r	1
No of Frames Used for	1-Frame	Channel	Correlated Rayleigh Fading
Inter-Frame Motion Search		Normalised Doppler Frequency	0.01
No of Video Slices/frame	9	Interleaver Length	$\approx (64000/15)$
No of Video MBs/Slice	11	No System Iterations I_t	10

(b) Systems parameters used in the schematic of Figure 3.27

Table 3.9: Code Rates and systems parameters used in the schematic of Figure 3.27

3.11.5 System Performance Results

The performance of the proposed system is characterised in this section. The H.264/AVC coded *"Akiyo"* video sequence presented in Section 3.5 was used as our source bit-stream to analyse the performance of our proposed system model. The remaining system parameters of our experimental setup are listed in Table 3.9(b).

For the sake of reducing the computational complexity imposed, we limited the number of iterations between the outer code and rate-1 inner precoder to $I_t = 10$. For the sake of increasing the confidence in our results, we repeated each 45-frame experiment 160 times and averaged the generated results. The performance of the system was evaluated by keeping the same overall code rate as well as video rate for both the EEP and UEP coding schemes. In the UEP1 scheme of Table 3.9(a) based on the specific concatenation order [A-B-C] of partition A, followed by partition B and then C, the highest grade of error protection was provided for partition A, while partition B was more strongly protected than C. By contrast, the UEP2 partitions arrangement [B-A-C] of Table 3.9(a) represents another error protection scheme, in which the highest level of error protection is

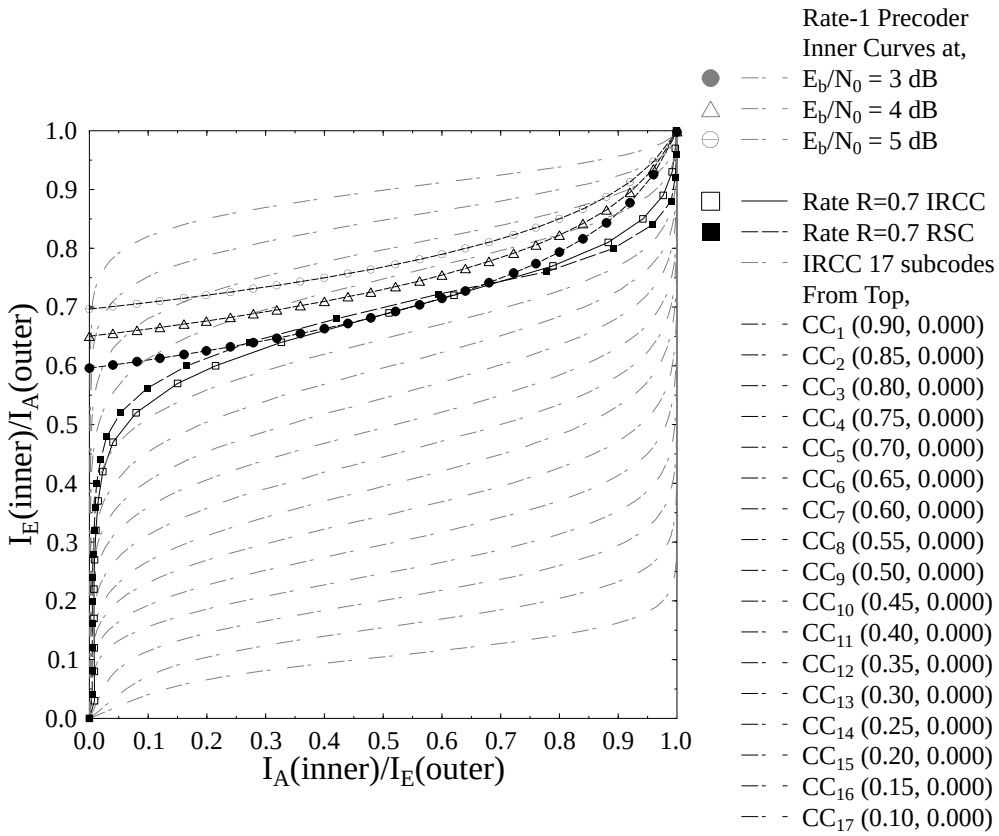

Figure 3.28: The EXIT functions of the $I = 17$ subcodes used in [160] along with the rate-1 inner precoder curves and the regular/irregular outer code curves employed.

provided for partition B, followed by partitions A and C.

The rationale of the UEP2 arrangement in comparison to the UEP1 scheme, is as follows. In the UEP1 scheme partition A had the strongest protection. It is worth noting that in case of the UEP1 scheme applied to low-motion video sequences the slices corrupted due to the loss of partition A are concealed using the motion vector recovery algorithm of [76], which results in less annoying artifacts due to their high correlation across the adjacent frames. However, the loss of partition B along with a correctly received better-protected partition A of UEP1 in a particular slice implies that the MV and header information received in partition A will be used to decode the MBs of this slice, while the similarly important MBs of partition B containing intra-frame coded MB coefficients and intra-MB CBP bits may be completely lost. Additionally, when the intra-frame coded MB updates of partition B are corrupted, this results in avalanche-like error propagation to the future video frames. Therefore, in low-motion video sequences partition B is also very important and hence its corruption results in noticeable PSNR degradation of the decoded video sequence. Hence, by providing a higher grade of error protection to partition B, the correct reception of partition B may be guaranteed for every correctly received partition A, which avoids error propagation to future video frames due to predictive video coding. Furthermore, partition B typically has the lowest number of bits, therefore the employment of a strong IRCC subcode does not result in a significant increase of the overall rate. As a result, using the remaining fraction of the bit-rate budget, a reasonable protection can still be provided for partitions A and C, which completes our justification of investigating the UEP2 scheme.

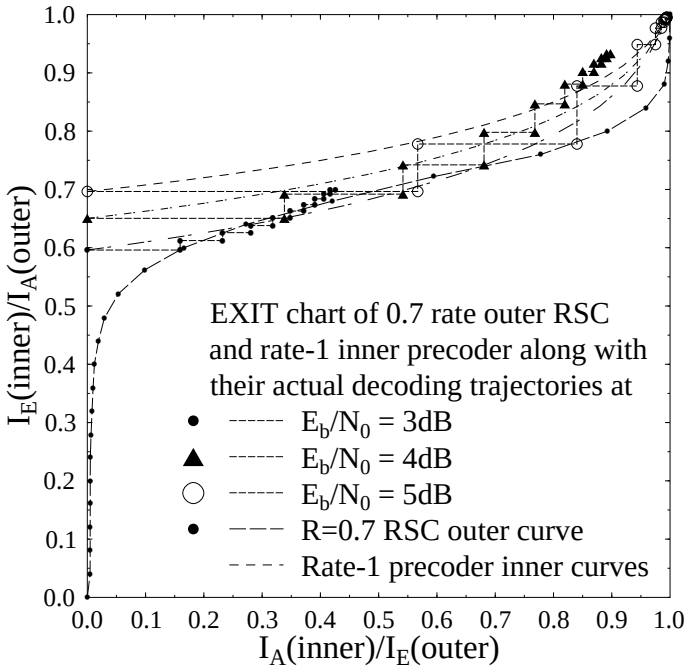

Figure 3.29: The EXIT chart and decoding trajectories of the EEP scheme of Figure 3.27 using the parameters of Table 3.28.

Figure 3.32 presents the attainable BER performance of partitions A, B and C when using the EEP, UEP1 and UEP2 schemes of Table 3.9(a), while employing regular/irregular outer channel codes. Naturally, the three partitions of the EEP scheme experience a similar BER, since they are protected equally with the same code rate. In the UEP1 scheme the partitions B and C have a higher BER than partition A, which is due to their higher code rate relative to partiton A. Likewise, partition A of the UEP2 scheme of Table 3.9(a) has a higher BER relative to partition B, which in turn has a lower BER than partition C. The major findings of the consider error protection schemes at PSNR degradation points of $1dB$ and $2dB$ are summarised in Table 3.10.

Table 3.10: Performance Summary Table

Schematic	Parameters	Iterations	Video rate	FEC Code rate	Highest PSNR value	E_b/N_0 at $1dB$ PSNR degradation	E_b/N_0 at $2dB$ PSNR degradation
Figure 3.27	Table 3.9(a)	10	64 *kbps*	0.7	41.7582	EEP_{RSC} = 5 dB $UEP1_{IRCC}$ = 5.5 dB $UEP2_{IRCC}$ = 4.875 dB	EEP_{RSC} = 4.875 dB $UEP1_{IRCC}$ = 5.125 dB $UEP2_{IRCC}$ = 4.625 dB

The performance trends expressed in terms of the $PSNR$ versus E_b/N_0 curves are portrayed in Figure 3.33. It may be observed in Figure 3.33 that the UEP2 scheme provides the best $PSNR$ performance among the three different error protection schemes of Table 3.9(a) across the entire E_b/N_0 region considered. From the simulation results of Figure 3.33 it may be observed that the UEP1 scheme of Table 3.9(a) results in the worst

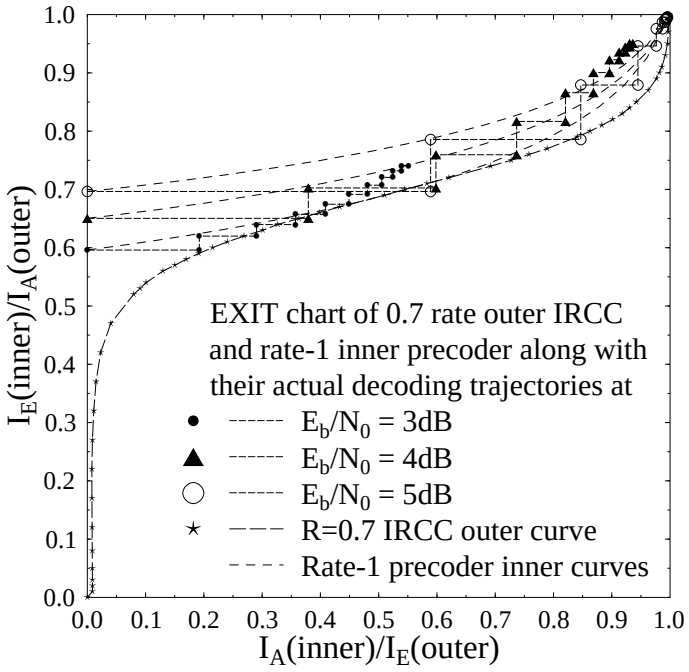

Figure 3.30: The EXIT chart and decoding trajectories of the UEP1 scheme of Figure 3.27 using the parameters of Table 3.28.

$PSNR$ performace, when employing our 0.7-rate IRCC in conjunction with the rate-1 inner precoder. By contrast, an E_b/N_0 gain of 1 dB is attained using UEP2 relative to UEP1 at the PSNR degradation point of 1 dB. Addtitionally, using the UEP2 scheme of Table 3.9(a), an E_b/N_0 gain of about 0.5 dB may be achieved over the identical-rate EEP benchmarker scheme using the regular RSC channel code of rate 0.7.

Finally, the subjective video quality achieved by the proposed EEP employing a regular convolutional code along with those of the UEP1 and UEP2 error protection schemes employing IRCC as outer code was recorded in Figure 3.34 at a channel E_b/N_0 value of 4 dB. In order to provide a fair subjective video quality comparison, we present both the average and cumulative-error results defined in Section 1.7.1 for both the luminance and chrominance components of the 30 frame *"Akiyo"* video test sequence described in Section 3.11.5, decoded using the H.264 video codec after transmission through the proposed system for each type of setup.

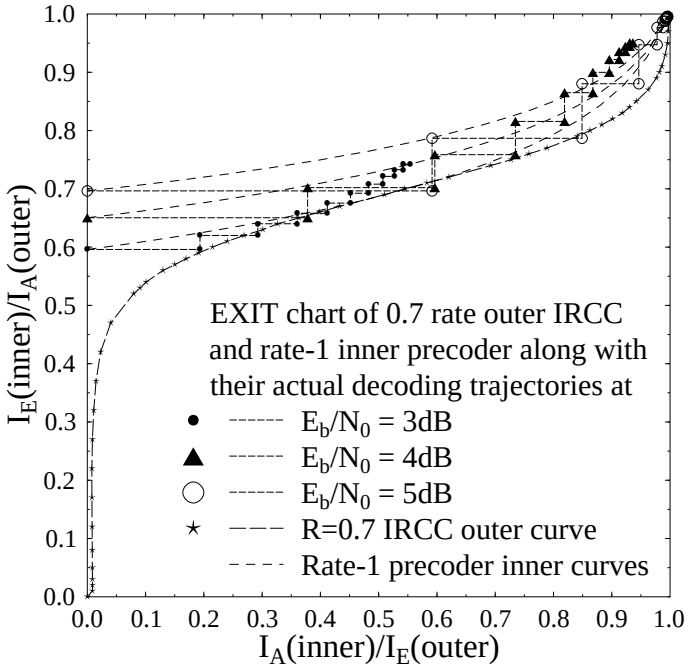

Figure 3.31: The EXIT chart and decoding trajectories of the UEP2 scheme of Figure 3.27 using the parameters of Table 3.28.

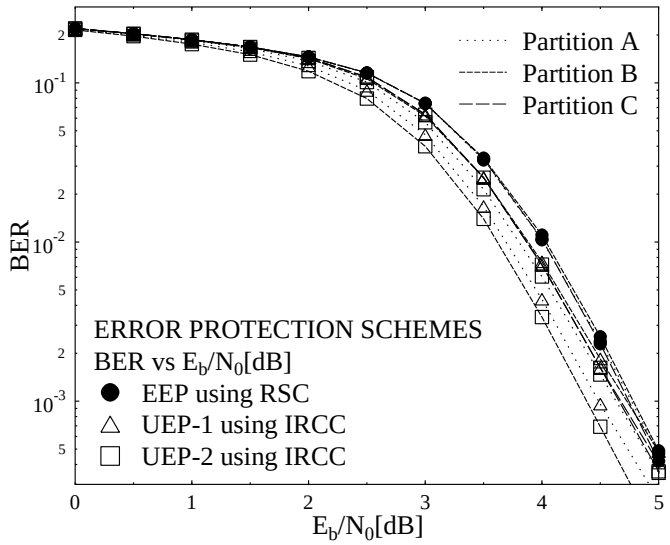

Figure 3.32: BER versus E_b/N_0 performance recorded for partitions A, B and C of the various error protection schemes of Figure 3.27 using the parameters of Table 3.9.

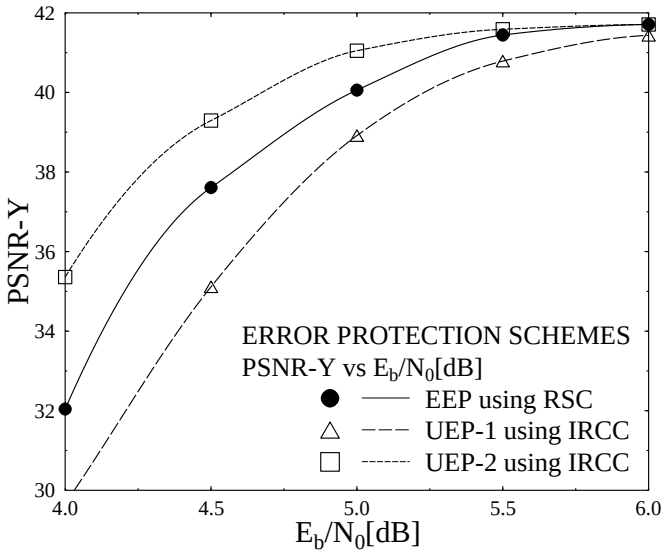

Figure 3.33: PSNR-Y versus E_b/N_0 performance of the various error protection schemes of Figure 3.27 using the parameters of Table 3.9.

Figure 3.34: Subjective video quality of the 45^{th} *"Akiyo"* video sequence frame in terms of (from top) average and cumulative-error video quality using (from left) EEP, UEP1 and UEP2 error protection schemes of Figure 3.27 using the parameters of Table 3.9 at $E_b/N_0 = 4\ dB$.

3.12 Chapter Conclusions

This chapter has the following findings,

- In Section 3.6 we elaborated on the hierarchical structure of the H.264 coded video stream and its bit-stream syntax. As depicted in Figure 3.12, the hierarchical structure of the H.264 coded video stream consists GOPs, which are composed of a number of video frames, while the frames are sub-divided into Slices. The slices are composed macroblocks, which in turn are composed of four (8x8)-pixel blocks.

- It was observed from bit-stream syntax of the H.264 video codec pictorally presented in Figure 3.13 of Section 3.7 that numerous coding parameters, such as Start Code Prefix, Information Byte, Sequence Parameter Set and Picture Parameter Set were encapsulated in the NAL unit.

- In Section 3.8 we observed that the H.264 DP generates different-sensitivity classes and provides us with the ability to protect the different sensitivity classes based on their relative importance.

- In Figures 3.16, 3.18, 3.19, 3.21 and 3.22 of Section 3.9 we performed a detailed error sensitivity study for partitions A, B and C of the H.264/AVC coded video stream. It was observed that in the H.264 video stream partition A is the most important type of partition, containing the most sensitive bits, followed by Partition B and then Partition C.

- Furthermore, in Section 3.10 we proposed the UEP H.264/AVC coded video transmission scheme of Figure 3.23 using RSC codes. Based on the error-sensitivity study of Section 3.9, the performance of the RSC coded system was analysed for the transmission of the DP-aided H.264/AVC coded bit-stream. We have demonstrated in Section 3.10 that by using UEP employing an appropriate channel coded bit-rate budget allocation to the different partitions of the H.264/AVC coded video based on their relative importance resulted in useful PSNR improvements in Figure 3.26. Explicitly, an E_b/N_0 gain of 4 dB was attained using UEP1 of Table 3.7(a) over UEP2, while an E_b/N_0 gain of 3 dB was achieved with reference to the EEP scheme, at the PSNR degradation point of 2 dB.

- The performance of a H.264 coded video transmission system using UEP aided IRCCs was evaluated in Section 3.11. We considered the various error protection schemes of Table 3.9(a) designed for the transmission of H.264/AVC coded video using the serially concatenated turbo transceiver of Figure 3.27, consisting of an EXIT-chart-optimised outer IRCC and an inner rate-1 precoder. Using UEP, perceptually more important bits were provided with more strong protection relative to less important bits. As seen in Figure 3.33, the iterative detection aided combination of the IRCC and the rate-1 precoder improved the overall BER performance and enhanced the objective video quality. The effect of the different error protection schemes of Table 3.9(a) on the attainable system performance was demonstrated in Figure 3.33, while keeping the overall bit-rate budget constant for the transmission of DP H.264 source coded video over correlated narrowband Rayleigh fading channels.

- In Section 3.11.3 we exploited the high design flexibility of IRCCs, which constitutes a family of different-rate subcodes, while maintaining an excellent iterative decoding convergence performance. As a benifit of using different-rate subcodes, IRCCs have the capability of providing UEP for the H.264

coded video stream, as detailed in Section 3.11.3. The EXIT chart matching procedure of Section 3.11.3 was used for the design of our specific IRCC.

- In Figures 3.29, 3.30 and 3.31 of Section 3.11.4 EXIT charts were used for analysing the attainable system performance of various error protection schemes using IRCCs employed. Explicitly, our experimental results of Figure 3.33 show that the proposed UEP scheme using IRCCs outperforms its EEP counterpart employing regular convolutional codes by about 0.5 dB E_b/N_0 at the PSNR degradation point of 1 dB. Moreover, the convergence behaviour of the proposed system of Figure 3.27 was analysed using EXIT charts in Figure 3.28.

3.13 Chapter Summary

An iterative detection aided scheme was proposed in Section 3.2 along with a state-of-the-art review of recent research contribution in this field. Furthermore, in Section 3.3, we introduced the concept of binary EXIT chart analysis. The procedure used to calculated the transfer characteristics of the inner and outer decoder is briefly summerise in Section 3.3.1 and Section 3.3.2. The concept of different component codes used in the iterative detection schemes is described in Section 3.4, followed by an overview of the RSC Codes in Section 3.4.1. The detailed description of the irregular coding and its various features used in digital multimedia communications in Section 3.4.2 and Section 3.4.3, followed by the details about the video source signal provided as input to our proposed systems in Section 3.5.

Section 3.6 presented the concepts behind the hierarchical structure of H.264 coded video consisted of GOP, frames, slices, Macro-blocks and blocks. The bit-stream syntax of the H.264/AVC coded video was detailed in Section 3.7. In Section 3.8 we described the H.264 DP concept. The H.264 provides us with the ability to protect the different sensitivity classes based on their relative importance. Furthermore, our H.264 error sensitivity study was presented in Section 3.9. In Section 3.10 we investigated an UEP aided video system using RSC codes, which relied on the architecture presented in Section 3.10.1. The source coding parameters of the video source signal provided as input to our proposed system is described in Section 3.5. It is demonstrated that the bit-error correction capability of the H.264 video codec can be considerably improved by properly allocating the available bit-rate budget between the source and channel code and results in considerable objective video quality improvement. Explicitly an E_b/N_0 gain of 4 dB is attained using UEP1 of Section 3.10 with reference to UEP2, while an E_b/N_0 gain of 3 dB is achieved with reference to EEP scheme, at the PSNR degradation point of 2 dB.

A novel iterative detection aided H.264 wireless video telephony scheme using IRCC was proposed in Section 3.11, followed by the proposed IRCC design in Section 3.11.3. Finally, the performance of the proposed system was analysed using EXIT charts in Section 3.11.4 and the calculated results along with discussion is provided in Section 3.11.5. Explicitly, our experimental results in Figure 3.33 showed that the proposed UEP scheme outperforms the identical-rate EEP benchmarker scheme using the regular RSC channel code of rate 0.7 by about 0.5 dB E_b/N_0 at the PSNR degradation point of 1 dB. Our conclusions were presented in Section 3.12.

Chapter 4

Robust Video Transmission Using Short Block Codes

4.1 Introduction

In Section 3.10, the performance of a video communication system was analysed using Unequal Error Protection (UEP) based Recursive Systematic Convolutional (RSC) codes described in Section 3.4.1, by appropriately allocating the total available bit rate budget to the different portions of the bit-stream, which resulted in an improved performance relative to the equal protection based single-class benchmarker scheme. We then further improved the achievable performance with the aid of IrRegular Convolutional Codes (IRCC) incorporated in an iterative H.264 joint source-channel decoding process, as detailed in Section 3.11. In contrast to Chapter 2, in Chapter 3 we focused our attention on characterising the Iterative Source-Channel Decoding (ISCD) process, while exploiting artificial channel coding based redundancy. As a further improvement, in this chapter we introduce a novel EXIT-chart based Short Block Coding (SBC) technique, which introduces further *artificial* redundancy into the source-coded bit-stream with the sole objective of improving the achievable iterative decoding gain. More explicitly, the convergence behaviour of the proposed iterative decoding techniques is analysed using the binary EXIT charts introduced in Section 3.3 and the specific SBC achieving the best possible performance is found. In this chapter the prototype system investigated relies on the Data-Partitioned (DP) H.264 coded video stream described in Section 3.8 and the video source is described by the parameter set of Section 3.5.

Robust transmission of multimedia source coded streams over diverse wireless communication networks constitutes a challenging research topic [1, 187]. Since the early days of wireless video communications [189–192] substantial further advances have been made both in the field of proprietary and standard-based solutions [186, 193]. Furthermore, the discovery of Turbo Codes [176, 194] made it practical to achieve transmission close to Shannon limit with moderate computational complexity and delay. The extension of the Turbo principle of exchanging extrinsic information have been successfully applied to various receiver components. One specific extension category of this field is ISCD. The joint optimisation of different functions such as

JSCD gained considerable attention in the recent decade. Fingscheidt and Vary [195, 196] proposed SBSD to exploit the natural residual redundancy of the source-coded bit-stream for improving the convergence of ISCD [188, 197]. Therefore, ISCD employ a decoding algorithm to exploit the explicit artificial redundancy due to channel encoding and implicit natural redundancy in terms of non-uniform distribution or correlation of the source encoded parameters. ISCD consists of Soft-Input Soft-Output (SISO) channel decoder and an Unequal Source-Symbol Probability Aided (USSPA) source decoder. The USSPA is a soft-decision version of source decoding, which utilises the source redundancy to estimate the source coding parameters.

In our performance evaluation setup the H.264/AVC video codec [74] is used to encode the input video sequence and to generate the source coded bitstream. However, only moderate residual redundancy is left in the source coded bit-stream, when using advanced state-of-the-art coding techniques. Therefore we propose to deliberately impose additional redundancy on the source coded bit-stream with the aid of the novel class of Short Block Codes (SBCs) proposed. The H.264/AVC codec employs heterogeneous Variable Length Coding (VLC) and predictive coding techniques to achieve a high compression efficiency, which makes the compressed bit-stream susceptible to transmission errors [1]. A single bit error in the coded stream may corrupt the decoding of numerous future codewords. Moreover, due to predictive coding the effects of channel errors may affect the neighboring video blocks due to error propagation. Therefore the transmission of compressed video over wireless systems is a challenging task.

Various Error resilient schemes have been proposed in [1], in order to alleviate these problems, but the price paid is a potential reduction of the achievable compression efficiency and increase in computational complexity. An iterative joint source-channel decoding procedure inspired by the concept of serial concatenated codes was presented in [198]. A symbol-based soft-input *A Posteriori* Probability (APP) decoder was presented in [199], where the residual redundancy was exploited for improved error protection. Instead of the traditional serial concatenation of the classic VLC with a channel code, a parallel concatenated coding scheme was presented in [200], where the VLCs were combined with a turbo code. On the other hand, a novel Irregular Variable Length Coding (IrVLC) scheme designed for near-capacity joint source and channel coding was proposed in [173]. A range of Adaptive Orthogonal Frequency Division Multiplex (AOFDM) video systems were proposed in [201] for robust, flexible, and low-delay interactive video telephony over wireless channels. Likewise, [202] advocated the employment of state-of-the-art High-Speed Packet Access (HSPA)-style [203] burst-by-burst adaptive transceivers for interactive cellular and cordless video telephony, which are capable of accommodating the time-variant channel quality fluctuation of wireless channels.

An iterative source and channel decoding aided IRCC videophone scheme using Reversible Variable-Length Codes (RVLC) and the Maximum A-Posteriori (MAP) [69] detection algorithm was proposed in [121]. The performance characteristics of different video transceivers assisted by decision feedback equaliser-aided wide-band burst-by-burst adaptive trellis-coded modulation, Turbo Trellis-Coded Modulation (TTCM) and bit-interleaved-coded modulation designed for H.263 assisted video telephony were presented in [122]. The performance analysis of soft bit assisted iterative JSCD was presented in [4], where Differential Space Time Spreading (DSTS) aided Sphere Packing (SP) modulation was invoked which dispensed with channel estimation and provided both spatio-temporal diversity as well as a multi-user support. Furthermore, a joint source-channel decoding method based on the MAP algorithm was proposed by Wang and Yu [123]. Instead of the well

known convolutional coded ISCD, an ISCD based on two serial concatenated SBCs was proposed by Clevorn *et al*. [204]. In the above-mentioned scheme a $(6, 3)$ outer block code served as a redundant index assignment, while a *rate*-1 block code was used as inner code. Similarly, Thobaben [205] provided the performance analysis of a rate $R = \frac{4}{5}$ single parity check code used for protecting the quantised source symbols combined with specifically designed VLCs.

Joint source-channel coding schemes employing a rate $R = \frac{4}{5}$ linear block code for mapping the quantised source symbols to a binary representation were combined with an inner irregular channel encoder in [206]. An optimised bit rate allocation scheme using a rate $r^\star = 1$[1] inner channel encoder along with $k = 3$ to $k^\star = 6$ source mapping was proposed in [207], and its performance was evaluated relative to conventional ISCD using a rate $r = \frac{1}{2}$ Recursive Non-Systematic Convolutional (RNSC) inner code. The Turbo DeCodulation scheme presented in [208, 209] consisted of two iterative loops. The inner loop was constituted by the two components of Bit-Interleaved Coded Modulation using Iterative Decoding Bit-Interleaved Coded Modulation using Iterative Decoding (BICM-ID) and the outer loop by the ISCD scheme. By contrast, in [210] a SBC based redundant index assignment and multi-dimensional mapping were used to artificially introduce redundancy and a single iterative loop was employed. Similarly, Clevorn *et al*. [211] presented a new design and provided optimisation guidelines for the ISCD's performance improvement using the concept of redundant index assignment with the aid of specific generator matrices. Additional overview of the past progression in the field of robust video transmission is presented in Table 4.1 and Table 4.2.

Against this background, in this chapter we commence with an overview of the ISCD process, highlighting the various building blocks, such as the transmitter and receiver. Furthermore, in contrast to the state-of-the-art research in the field of ISCD, where specific mapping examples were provided for ISCD [204], we present powerful yet low-complexity algorithms for SBCs, which can be used to design them for a variety of mapping/coding rates associated with diverse $d_{H,min}$ values that are applicable to wide-ranging multimedia services. Additionally, instead of modelling the sources with the aid of Markov models mimicking their correlation, the practically achievable interactive video performance trends are quantified by using state-of-the-art video coding techniques, such as the H.264/AVC codec. More explicitly, instead of assuming a specific source-correlation model, such as the first-order Markov model, we based our system design examples on the simulation of the actual H.264/AVC source coded bit-stream. The SBC-aided USSPA scheme of Section 4.5 was utilised for protecting the H.264 coded bit-stream using RSC codes [69]. The SBC coding scheme was incorporated by carefully partitioning the total available bit-rate budget between the source and channel codecs, which results in an improved performance, when ISCD is employed.

The novelty and rationale of this chapter can be summarised as follows [3]:

1. *We apply iterative soft-bit source decoding to decode the unequal error protected video stream encoded using the sophisticated state-of-the-art H.264 video scheme operated at a low target bit-rate of* 64 $kbps$. *The achievable performance improvements are also quantified.*

2. *We conceive a powerful yet low-complexity algorithms for SBCs, which can be used to generate SBCs for*

[1] The superscript $\star$ was adopted from [207]

Year	Author(s) name and contribution
1997	*Author(s):* E. Steinbach, N. Farber and B. Girod [212] *Contribution:* a robust video transmission scheme was designed for error-prone environments using the ITU-T H.263 video coding standard.
1999	*Author(s):* J. H. Lu, K. B. Letaief and M. L. Liou [129] *Contribution:* a unique set of techniques was developed to support reliable video transmission over bandwidth-limited mobile networks using the H.263 coding standard. The transmission scheme was based on combined source and channel coding, diversity reception, and pre/postprocessing techniques.
2000	*Author(s):* C. S. Kim, R. C.Kim and S. U. Lee [213] *Contribution:* a robust video transmission algorithm was designed using redundant parity-check DC coefficients (PDCs) that were systematically inserted into the compressed bit stream for protecting the compressed video stream.
2001	*Author(s):* C. S. Kim, S. U. Lee [214] *Contribution:* a Multiple Description Motion Coding (MDMC) algorithm was proposed to enhance the robustness of the motion vectors, which constitute one of the most important component in the compressed bitstream.
2003	*Author(s):* M. Grangetto, E. Magli and G. Olmo [124] *Contribution:* a joint entropy coding and error protection scheme was designed for robust H.264 coded video, which relied on Arithmetic Codes (ACs) combined with a forbidden symbol and with DP for transmission over error-prone channels.
2004	*Author(s):* T. Fang and L. P. Chau [215] *Contribution:* a robust video transmission framework was presented to ensure that the image quality of the foreground was improved at the expense of sacrificing the video quality of the less important background.
2005	*Author(s):* C. S. Kim, J. W. Kim, I. Katsavounidis and C. J. Kuo [216] *Contribution:* a novel video decoding algorithm based on the Minimum Mean Square Error (MMSE) criterion was advocated in order to alleviate the effects of transmission errors.
2006	*Author(s):* A. Ksentini and M. Naimi and A. Gueroui [118] *Contribution:* a robust cross-layer architecture was presented that exploits the advantages of the H.264/AVC error resilience features combined with the Medium Access Control (MAC) capabilities of 802.11 Wireless Local Area Networks (WLANs) [118].

Table 4.1: Advances in Robust Video Transmission (Part 1).

Year	Author(s) name and contribution
2007	*Author(s):* R. Puri and A. Majumdar and K. Ramchandran [217] *Contribution:* introduced the concept of a novel scheme termed as Power-efficient, Robust, hIghcompression, Syndrome-based Multimedia coding (PRISM), which is a video coding paradigm based on the principles of lossy distributed source coding operating with the aid of side information or Wyner-Ziv coding. Specifically, PRISM enables the designer to dispense with the computationally complex motion-search module at the encoder, which is invoked then at the video decoder.
2008	*Author(s):* P. Ferre and A. Doufexi and J. Chung-How and A. R. Nix and D. R. Bull [128] *Contribution:* a packetization method was proposed for robust H.264 video transmission over the IEEE 802.11 wireless local area network (WLAN) to overcome the poor throughput efficiency of the IEEE 802.11 Medium Access Control (MAC). The proposed scheme maps several IP packets (each containing a single H.264 video packet) into a single larger MAC frame. The attainable video robustness is enhanced by using small video packets and by retrieving potentially error-free IP packets from the received MAC frame.
2009	*Author(s):* V. Sgardoni, M. Sarafianou, P. Ferre, A. Nix and D. Bull [129] *Contribution:* a robust video solution was developed for H.264 video broadcast sequences over 802.11a/g for transmission over time-correlated fading channels.
	Author(s): J. Wang and A. Majumdar and K. Ramchandran [218] *Contribution:* an auxiliary codec was proposed that sends additional information alongside an MPEG or H.26x compressed video stream to correct the errors in the decoded frames. The proposed system was based on the principles of distributed source coding and used the erroneous by reconstructed frames as side information at the auxiliary decoder.
2010	*Author(s):* M. H. Lu and P. Steenkiste and T. Chen [219] *Contribution:* a hybrid spatio-temporal retransmission protocol was proposed for wireless video streaming applications. The system uses an opportunistic retransmission protocol that relies on overhearing nodes distributed in the physical space, which may then be invoked to retransmit failed packets on behalf of the source. To meet strict timing constraints, a Time-based Adaptive Retransmission strategy (TAR) was applied by both the source and the relays based on the retransmission deadline of the packets.

Table 4.2: Advances in Robust Video Transmission (Part 2).

a variety of mapping rates associated with diverse $d_{H,min}$ values, which is applicable to wide-ranging multimedia services.

3. *We characterise the attainable performance of diverse coding rates, while using SBCs generated by the proposed Algorithm-I. The resultant performance is also characterised with the aid of EXIT charts along with the corresponding bit-by-bit Monte-Carlo simulation based decoding trajectory. The performance of the proposed Algorithm-II is also quantified, demonstrating that Algorithm-II results in an increased $d_{H,min}$ value at the same overall code rate as Algorithm-I.*

4. *The practically achievable interactive video performance trends are quantified using the state-of-the-art H.264/AVC codec. We demonstrate that conventional ISCD schemes result in relatively modest performance improvements for the H.264 codec, because the achievable performance benefits of USSPA are dependent on the amount of redundancy inherent in the source-coded bit-stream. Therefore, we demonstrate that substantial system performance improvements are achieved by specifically designing the proposed EXIT-chart-based SBCs.*

The rest of the chapter is organised as follows. In Section 4.2 we portray the architecture of the transmitter and of the conventional receiver. The detailed description of the receiver using iterative source and channel decoding is provided in Section 4.3. In Section 4.4 we describe an architecture employing soft-bit video source decoding assisted UEP using RSC codes. The achievable iterative decoding convergence is characterised in Section 4.5 for the proposed SBC algorithms, accompanied by a design example based on EXIT chart analysis and followed by our video performance results. Similarly, the performance analysis of SBCs using a rate-1 inner precoder is provided in Section 4.6 along with the corresponding EXIT chart analysis and video performance results. The concept of Redundant Source Mapping is detailed in Section 4.7 with the aid of a design example. Finally, the chapter is concluded with a summary in Section 4.9.

4.2 Transmitter and Non-Iterative Receivers for Video Transmission

The building blocks of the transmitter and conventional receiver are presented in Figure 4.1 and are explained below.

4.2.1 Source Encoding

The source coded video bit stream representing a video clip s is partitioned into relatively short transmission frames, labeled with the index k. The parametric source encoder extracts a parameter set v_k that consists of M scalar source codec parameters $v_{m,k}$, $m = 1, \cdots, M$ where again, k is the frame index. The M source codec

parameters $v_{m,k}$ are mapped to a bit-sequence according to $v'_{m,k} \in V_m$. The mapping codebook V_m of size $|V_m|$ is time invariant. The length of the resultant bit pattern $x_{m,k}$ is $w_m = log_2(|V_m|)$. Due to the limited affordable complexity and delay of the encoding process, there is a certain residual source redundancy associated with the bit pattern after source encoding. The residual source redundancy is quantified in terms of the probability values $P(x)$ and $P(x_k|x_{k-1})$.

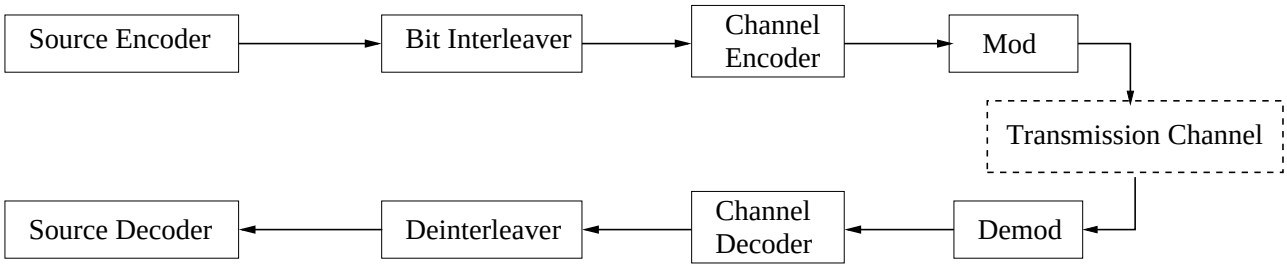

Figure 4.1: Transmitter and conventional receiver for video transmission.

4.2.2 Bit Interleaving

The employment of interleaving in data transmission disperses the bursts of transmission errors and hence substantially improves the attainable performance of most channel codecs by more-or-less uniformly distributing the errors. To elaborate a little further, burst errors often corrupt numerous consecutive bits in a row, therefore error correcting codes whose performance is directly dependent on the near-uniform distribution of errors are gravely affected by bursty errors. Therefore interleaving may be used to overcome this problem. Hence in the ISCD the bit interleaver Π of Figure 4.2 is used before the channel decoder in order to permute the $M \cdot w$ data bits x of a frame in a deterministic manner. The task of the deinterleaver is then to reinstate the bits in their original position. Furthermore, in iterative decoding aided turbo-receivers the achievable iteration gain also improves upon increasing the interleaver-length owing to the provision of more independent extrinsic information.

4.2.3 Modulation and Equivalent Transmission Channel

In this chapter we restrict our considerations to Quadrature Phase Shift Keying (QPSK) modulation in order to investigate the concept of ISCD. The QPSK modulated bit-stream is transmitted over a temporally correlated narrowband Rayleigh fading channel, associated with the normalised Doppler frequency of $f_d = f_D T_s = 0.01$, where f_D is the Doppler frequency and T_s is the symbol duration.

4.2.4 Non-Iterative Receiver

In a conventional non-iterative receiver, the inverse of the signal processing steps of the transmitter is invoked in order to regenerate the original transmitted signal. The received signal is first demodulated and subsequently

channel decoded. After deinterleaving the channel decoded stream is fed into the source decoder, which performs parameter reconstruction in order to generate the estimated video source signal $\hat{s}$.

4.3 Iterative Source-Channel Decoding Aided Receivers

4.3.1 Log-Likelihood Ratio

The channel coded symbols y are transmitted over an AWGN channel, where they are contaminated by the noise process n. The corresponding noise-contaminated symbols at the input of the receiver are represented as $z = y + n$ with $z \in R$.

The reliability of the received symbols can be expressed in terms of the transition probabilities $p(z|y)$ or in form of the corresponding Log-Likelihood Ratio (LLR)(L-values) or $L(z|y)$

$$L(z|y) = log\frac{p(z|y=+1)}{p(z|y=-1)}. \tag{4.1}$$

The LLRs at the output of the AWGN channel may be expressed as [176]

$$L(z|y) = 4a.E_s/N_0 \cdot z. \tag{4.2}$$

In Equation (4.2) the term $N_0/2$ represents the power spectral density of the AWGN noise n, while a denotes the fading factor.

4.3.2 Iterative Source-Channel Decoding

ISCD is used to jointly exploit both the natural residual source redundancy and the artificial channel coding redundancy as well. The optimum bit-decision is based on the *a posteriori* probability $P(x|z)$ or the corresponding *a posteriori* LLR-value $L(x|z)$ for a single data bit x, given the complete received sequence z. In case of a memoryless channel the *a posteriori* L-value is composed of four additive terms [220–222]

$$L(x|z) = L(z|x) + L(x) + L_{CD}^{extr}(x) + L_{USSPA}^{extr}(x). \tag{4.3}$$

The LLR-value $L(z|x)$ specifies the bitwise channel-dependent reliability for systematic channel encoding. In case of a non-systematic channel code the original input bit x is not directly copied to the encoder's output to become part of the coded symbol z, hence we have no probabilistic dependence between them, yielding $L(z|x)$=0. The second term of Equation (4.3), namely $L(x) = log(P(x=+1)/P(x=-1))$ presents the bitwise *a-priori* information and determines, whether a data bit of $x=+1$ or $x=-1$ is more likely to have been transmitted by the source, which can be determined in advance i.e. prior to transmission for the source considered. The remaining two terms of Equation (4.3), namely $L_{CD}^{extr}(x)$ and $L_{USSPA}^{extr}(x)$ quantify the so-called extrinsic information, which arises from the evaluation of the artifical channel coding redundancy and natural residual source redundancy, respectively. Again, the natural redundancy manifests itself in terms of having an unequal probability of occurance for the particular bit-pattern combinations. Having independent extrinsic LLR-values is essential for the success of iterative decoding in order for the components to assist each other

in an iterative fashion. In ISCD the de-interleaved extrinsic output of the channel decoder serves as additional soft-input for the source decoder. Similarly, the interleaved extrinsic output of the soft-bit source decoder serves as additional input to the inner channel decoder. This way the iterative refinement of the extrinsic LLR-values results in step-wise reliability improvements.

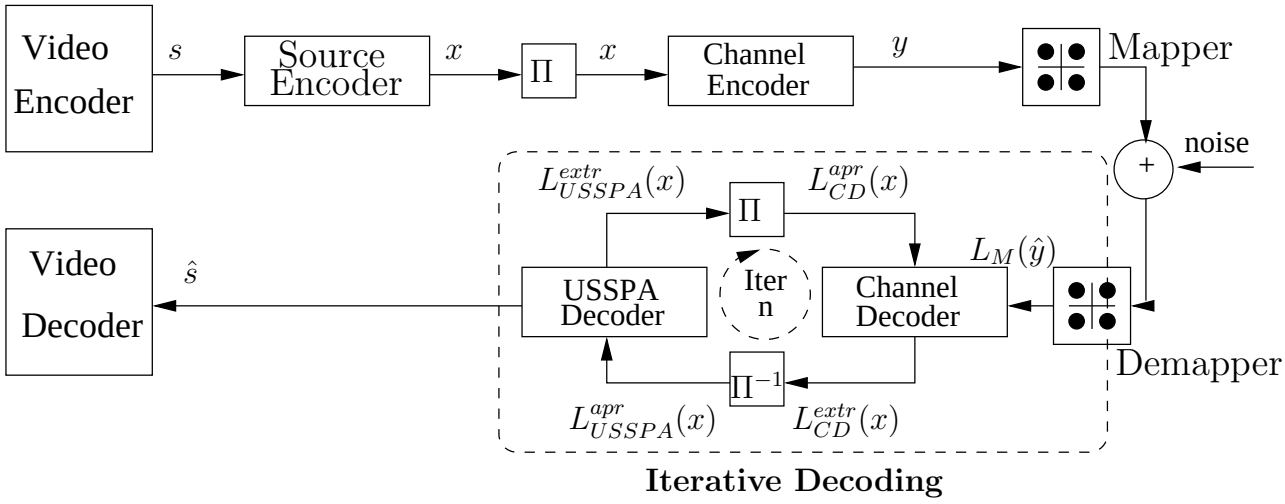

Figure 4.2: ISCD System Model.

Extrinsic Output of Channel Decoder [$L^{extr}_{CD}(x)$]:
The extrinsic LLR-values generated by the channel decoder of Figure 4.2 during the n-th iteration are given by

$$L^{extr,n}_{CD}(x) = \mathbb{F}_{CD}\{L_M(\hat{y}), L^{apr}_{CD}(x)\}, \tag{4.4}$$

where the functions $L_M(\hat{y})$ and $L^{apr}_{CD}(x)$ are given by

$$L_M(\hat{y}) = L(z|y) \tag{4.5}$$

$$L^{apr}_{CD}(x) = L(z|x) + L(x) + L^{extr,n-1}_{USSPA}(x). \tag{4.6}$$

The LLR-values $L(z|y)$ at the output of the channel are available for all the received bits y. Similarly, the LLR-value $L(z|x)$ is available for all data bits x, provided that the channel coding is of the systematic form. Additionally, $L(x)$ represents the bit-wise *a-priori* LLR-values, while $L^{extr,n-1}_{USSPA}(x)$ is the interleaved soft-input feedback information provided by the USSPA for the data bits x. The variable $L^{extr,n-1}_{USSPA}(x)$ represents the LLR-value generated by the previous iteration, which is available for the subsequent iteration in the ISCD process of Figure 4.2.

Extrinsic Output of Unequal Source-Symbol Probability Aided Decoding of the Source Code [$L^{extr}_{USSPA}(x)$]:
The Unequal Source-Symbol Probability Aided Decoding (USSPA) is used to exploit the natural residual source redundancy that remains in the source-coded stream during the iterative decoding process. The reason for this residual source redundancy is the latency and complexity limitations of the encoding process. The amount of redundancy inherent after source compression can be quantified in advance for a sufficiently long training sequence having statistical properties reminiscent of those of the typical signals considered in terms of the *probability density values* $P(x)$ or $P(x_k|x_{k-1})$, respectively.

The detailed procedure used for determining the *extrinsic* LLR-value $L_{USSPA}^{extr}(x)$ of USSPA while utilising the *a-priori* knowledge $P(x)$ or $P(x_k|x_{k-1})$ of the bit-pattern x given by the LLR-value $L_{USSPA}^{apr}(x)$ was provided in [220, 221, 223–225].

The extrinsic LLR-value generated by the USSPA decoder of Figure 4.2 is given by

$$L_{USSPA}^{extr,n}(x) = \mathbb{F}_{USSPA}\{L_{USSPA}^{apr}(x)\}, \tag{4.7}$$

where $L_{USSPA}^{apr}(x)$ is formulated as

$$L_{USSPA}^{apr}(x) = L(z|x) + L(x) + L_{CD}^{extr,n}(x). \tag{4.8}$$

The initial LLR-values $L_{USSPA}^{extr,0}(x)$ are set to zero at the commencement of the iterative decoding process, since no previous extrinsic information is available. This correspond to having logical 0 and 1 values with a probability of 0.5. After the first iteration the updated LLR-values become available from the USSPA decoder of Figure 4.2, which are forwarded to the channel decoder as *a priori* information, in order to assist the channel decoder in achieving an iterative performance improvement. Therefore the iterative calculation of Equation (4.4) and (4.7) results in successive improvements of the reliability information $L_{USSPA}^{extr,n}(x)$ and $L_{CD}^{extr,n}(x)$ concerning the data bits x. During the iterative decoding process of Figure 4.2, further performance improvements are possible, as long as the extrinsic information terms remain mutually independent. In order to ensure this independence, an interleaver is inserted in Figure 4.2 between the two constituent decoders for providing a certain level of independence after a specific number of iterations. Nonetheless, due to the iterative interaction, between the component decoders the extrinsic information terms become dependent on each other after a certain number of iterations, unless an 'infinite' interleaver length was used, which is impractical.

4.3.3 Soft-Input/Soft-Output Channel Decoding

The best known method found in the literature for determining the bit-wise extrinsic LLR-values $L_{CD}^{extr}(x)$ from the artificial redundancy introduced by the channel encoder is the symbol-by-symbol MAP estimation or Bahl-Cocke-Jelinek-Raviv (BCJR)-channel decoder detailed for example in [167, 226]. A convolutional code may be described with the aid of its trellis diagram. As an example, the l-th stage trellis diagram of a memory $J = 2$ Recursive Systematic Convolutional (RSC) code having a code rate of $r = 1/\lambda = 1/2$, $\lambda \in N$ and an octally represented generator polynomial of $G(1, 5/7)_8$ is given in Figure 4.3. In this diagram the *encoder states* are represented with nodes, while the branches indicate the legitimate *state transitions*. During the channel encoding process, the l-th input bit results in the encoder state transition emerging from $S(l-1)$ to $S(l)$. As seen in Figure 4.3, given a binary input, there are only two legitimate branches emerging from the current encoder state and traversing to the next state. Similarly, due to the employment of a convolutional code having $J = 2$ there are four different states. Figure 4.4 shows the dependencies of S_l on both the past bit-pattern S_{l-1} and on the future bit-pattern S_{l+1}. The probability of each valid state transition from $S(l-1)$ to $S(l)$ is computed at the receiver using the BCJR algorithm [69, 176] according to

$$\gamma_l[S(l-1), S(l)] = P[S(l)|S(l-1)] \cdot \exp\left(\sum_{j=1}^{1/r} \frac{y_l(j)}{2}.L_{CD}^{apr}[y_l(j)]\right), \tag{4.9}$$

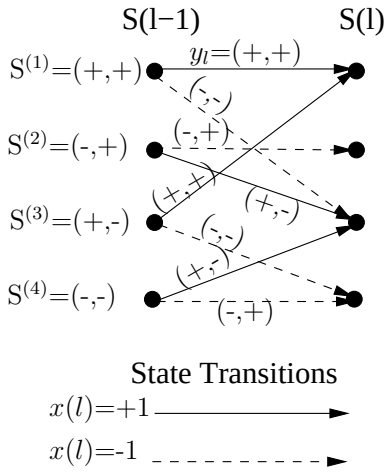

Figure 4.3: Intermediate trellis state of RSC code.

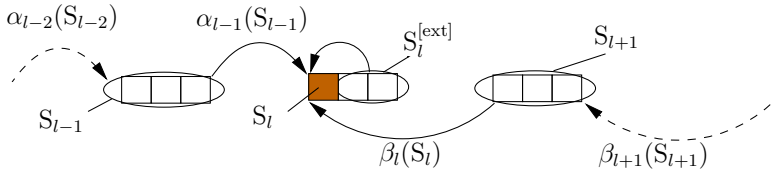

Figure 4.4: Dependencies of the current trellis state on the past and future bit-patterns.

where $L_{CD}^{apr}[y_l(j)]$ is calculated according to Equation (4.6) if $y_l(j)$ consists of systematic data bits $x(l)$ or according to Equation (4.5) otherwise. Since the encoder's operation may be modeled by that of a state-machine, which changes states on the basis of the current state and the incoming bit, we have to take into account the contribution from the preceding trellis stages $0, \cdots, (l-1)$, which can be included recursively as [195]

$$\alpha(S^{(i)}(l)) = \sum_{j=1}^{2^J} \gamma_l[S^{(j)}(l-1), S^{(i)}(l)].\alpha_{l-1}[S^{(j)}(l-1)]. \tag{4.10}$$

Similarly, the reliability information contribution from the future stages $l+1, \cdots, Mw+J$ as regards to the state transition from $S(l)$ to $S(l+1)$ is given by [195]

$$\beta_l(S^{(j)}(l)) = \sum_{i=1}^{2^J} \gamma_{l+1}[S^{(i)}(l), S^{(j)}(l+1)] \cdot \beta_{l+1}[S^{(i)}(l+1)]. \tag{4.11}$$

Finally, the overall rule for the extrinsic LLR-value evaluation of channel decoding is given by [195]

$$L_{CD}^{extr}(x) = log \frac{\sum_{i=1}^{2^J} \beta_l[S^{(i)}(l)] \cdot \sum_{j=1}^{2^J} \gamma_l^{extr}[S^{(j)}(l-1)], S^{(i)}(l)|x=+1) \cdot \alpha_{l-1}[S^{(i)}(l-1)]}{\sum_{i=1}^{2^J} \beta_l[S^{(i)}(l)] \cdot \sum_{j=1}^{2^J} \gamma_l^{extr}[S^{(j)}(l-1)], S^{(i)}(l)|x=-1) \cdot \alpha_{l-1}[S^{(i)}(l-1)]}. \tag{4.12}$$

Furthermore, the contribution of the systematic data bit $x(l) = y_l(j_{sys})$ at position j_{sys} in y_l is excluded from Equation (4.9) when generating the extrinsic information, because this information was already exploited, hence would not provide an independent source of extrinsic information and hence would prevent us from achieving

an iteration gain. The exclusion of this information from the summation is formulated as

$$\gamma_l[S(l-1), S(l)] = P[S(l)|S(l-1)] \cdot \exp\left(\sum_{j=1, j\neq j_{sys}}^{1/r} \frac{y_l(j)}{2}.L_{CD}^{apr}[y_l(j)]\right). \quad (4.13)$$

Therefore, considering our example the sum in Equation (4.13) runs over $j = 2, \cdots, 1/r$.

4.3.4 Unequal Source-Symbol Probability Aided Decoding of the Source Code

In this section, we will elaborate on how to exploit the natural residual source redundancy that remains in the source-coded stream during the iterative decoding process. The reason for this residual source redundancy is the latency and complexity limitations of the encoding process. The amount of redundancy inherent after source compression can be quantified in advance for a sufficiently long training sequence having statistical properties reminiscent of those of the typical signals considered in terms of the *probability density values* $P(x)$ or $P(x_k|x_{k-1})$, respectively.

The detailed procedure used for determining the *extrinsic* LLR-value $L_{USSPA}^{extr}(x)$ of USSPA while utilising the *a priori* knowledge $P(x)$ or $P(x_k|x_{k-1})$ of the bit-pattern x given by the LLR $L_{USSPA}^{apr}(x)$ according to Equation (4.8) was provided in [220, 221, 223–225].

There are two different scenarios, in which the extrinsic information may be extracted from the residual source redundancy, namely;

1. extrinsic information resulting from a non-uniform parameter distribution $P(x)$.
2. extrinsic information resulting from the correlation parameter of a specific source codec across the consecutive frames, which is formulated as $P(x_k|x_{k-1})$.

These concepts of extrinsic information extraction from the residual source redundancy may be demonstrated with the aid of the following example.

4.3.4.1 Extrinsic Information Resulting From a Non-uniform Parameter Distribution $P(x)$

The USSPA of Figure 4.2 is used for extracting the extrinsic information x^{extr} for the data-bit x from all the other surrounding data-bits of the bit-pattern x. For example, in case of the bit pattern x= $[x(1), x(2), x(3)]$ and w=3, the extrinsic information generated for the bit $x(1)$ is extracted from the pattern $\mathrm{x}^{ext]} = [x(2), x(3)]$.

Consider the simple case, in which the extrinsic information is extracted from the probability mass values $P(x)$ and from the $(w-1)$ x^{extr} bits of the bit pattern x, which are assumed to be perfectly known at the receiver. In this case the extrinsic LLR-value becomes

$$L_{USSPA,P(x)}^{extr}(x) = log\frac{P(\mathrm{x}^{extr}|x=+1)}{P(\mathrm{x}^{extr}|x=-1)}. \quad (4.14)$$

However, normally the $(w-1)$ x^{extr} bits are not known with absolute certainty at the receiver, hence the reliability or probability of all possible realizations of x^{extr} has to be determined on the basis of the input

LLR-values of the USSPA [195] seen in Figure 4.2, yielding

$$\theta(\mathrm{x}^{extr}) = \exp \sum_{l=1, l\neq l_x}^{w} \frac{x(l)}{2} \cdot L_{USSPA}^{apr}[x(l)]. \tag{4.15}$$

Observe from Equation (4.15) that the data bit x associated with the bit-index l_x under consideration is excluded from the calculation. Evoluating (4.15) for the $2^w - 1$ possible realizations of x^{extr} allows us to extend Equation (4.14) to [220, 221, 223–225]:

$$L_{USSPA,P(x)}^{extr}(x) = log \frac{\sum_{\mathrm{x}^{extr}} P(\mathrm{x}^{extr}|x=+1).\theta(\mathrm{x}^{extr})}{\sum_{\mathrm{x}^{extr}} P(\mathrm{x}^{extr}|x=-1).\theta(\mathrm{x}^{ext})}. \tag{4.16}$$

Equation (4.16) may be used for determining the extrinsic LLR-value provided by the USSPA of Figure 4.2 for the subsequent iteration, when a non-uniform distribution of the bit-pattern x is utilised as *a priori* knowledge. In case of perfect knowledge about the bit-pattern x^{extr}, the term $\theta(\mathrm{x}^{extr})$ in Equation (4.16) becomes non-zero only for one specific pattern and it is zero for all other patterns, since the summations in Equation (4.16) run over all possible 2^{w-1} realizations of x^{extr}. Bearing this in mind, Equation (4.16) reduces to (4.14).

4.3.4.2 Extrinsic Information Resulting From Parameter Correlation $P(\mathrm{x}_k|\mathrm{x}_{k-1})$

In order to exploit the mutual dependencies of the the bit-pattern x^{extr} as a function of time, the conditional probability $P(\mathrm{x}^{extr}|x)$ in (4.16) has to be replaced by

$$P(\mathrm{x}_k^{extr}|x_k) = \sum_{\mathrm{x}_{k-1}} P(\mathrm{x}_k^{extr}|\mathrm{x}_k, \mathrm{x}_{k-1}).\alpha_{k-1}(\mathrm{x}_{k-1}). \tag{4.17}$$

It becomes explicit from Equation (4.17) that the correlation between x_k and the past patterns $\mathrm{x}_{k-1}, \cdots, \mathrm{x}_1$ is exploited using the factor $\alpha_{k-1}(\mathrm{x}_{k-1})$.

Similarly, future bit-patterns $\mathrm{x}_{k+1}, \cdots, \mathrm{x}_z$ may be taken into account using an extra factor $\beta_k(\mathrm{x}_k)$: Figure 4.5

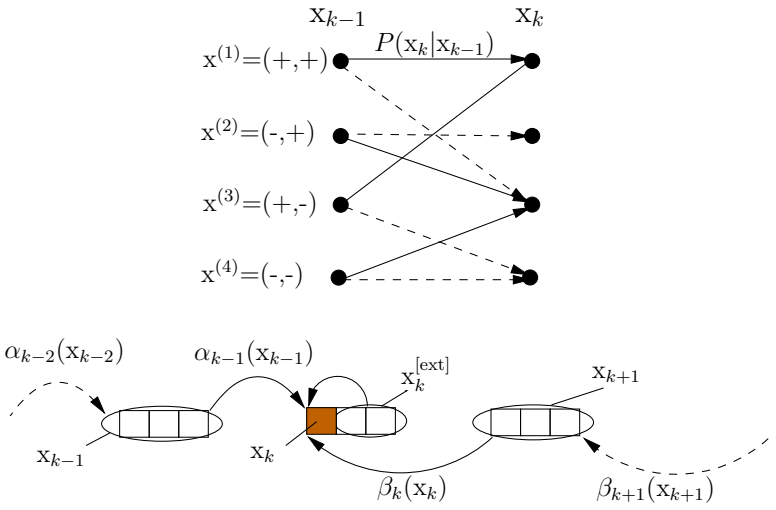

Figure 4.5: Intermediate trellis state of the USSPA of Figure 4.2 exploiting the correlation of the consecutive values of a given parameter.

shows the dependencies of x_k on both the past bit-pattern x_{k-1} and the future bit-pattern x_{k+1}. Therefore, similar to Equations (4.10) and (4.11), both the past terms $\alpha_{k-1}(\mathrm{x}_{k-1})$ and future term $\beta_k(\mathrm{x}_k)$ can be efficiently computed in a recursive manner as follows [195]

$$\alpha_k(\mathrm{x}_k) = \sum_{\mathrm{x}_{k-1}} \gamma_k(\mathrm{x}_k, \mathrm{x}_{k-1}).\alpha_{k-1}(\mathrm{x}_{k-1}) \tag{4.18}$$

$$\beta_k(\mathrm{x}_k) = \sum_{\mathrm{x}_{k+1}} \gamma_{k+1}(\mathrm{x}_{k+1}, \mathrm{x}_k).\beta_{k+1}(\mathrm{x}_{k+1}), \tag{4.19}$$

where the summations in the forward recursion of Equation (4.18) and in the backward recursion of Equation (4.19) are performed over all 2^w realizations of x_{k-1} or x_{k+1}, respectively.

The reliability information for each valid state transition from x_k to x_{k-1} is computed at the receiver similar to Equation (4.9) as

$$\gamma_k(\mathrm{x}_k, \mathrm{x}_{k-1}) = P(\mathrm{x}_k|\mathrm{x}_{k-1}) \cdot \exp\left(\sum_{l=1}^{w} \frac{x(l)}{2}.L_{USSPA}^{apr}(x(l))\right), \tag{4.20}$$

where the input $L_{USSPA}^{apr}[x(l)]$ is given by (4.8). Furthermore, for initialisation we used $\alpha_0(\mathrm{x}_0) = p(\mathrm{x}_0)$ and $\beta_z(\mathrm{x}_z) = 1$.

Finally, the extrinsic LLR-value generated by the USSPA of Figure 4.2 for the subsequent iteration is completed by substituting Equations (4.17), (4.18) and (4.19) into Equation (4.16), yielding:

$$L_{USSPA}^{extr}(x) = \log \frac{\sum_{\mathrm{x}^{extr}} \beta_k(\mathrm{x}_k^{extr}, x_k = +1). \sum_{\mathrm{x}_{k-1}} \gamma_k^{extr}(\mathrm{x}_k^{extr}, \mathrm{x}_{k-1}|x_k = +1).\alpha_{k-1}(\mathrm{x}_{k-1})}{\sum_{\mathrm{x}^{extr}} \beta_k(\mathrm{x}_k^{extr}, x_k = +1). \sum_{\mathrm{x}_{k-1}} \gamma_k^{extr}(\mathrm{x}_k^{extr}, \mathrm{x}_{k-1}|x_k = -1).\alpha_{k-1}(\mathrm{x}_{k-1})}. \tag{4.21}$$

Furthermore, the equivalent dual pair of Equation (4.13) may be applied to obtain the extrinsic information

$$\gamma_k^{extr}(\mathrm{x}_k^{extr}, \mathrm{x}_{k-1}|x_k) = P(\mathrm{x}_k^{extr}|x_k, \mathrm{x}_{k-1}).\theta(\mathrm{x}_k^{extr}), \tag{4.22}$$

where $\theta(\mathrm{x}_k^{extr})$ is defined in Equation (4.15), which excludes the data bit x under consideration from the bit-pattern x.

4.4 Unequal Source-Symbol Probability Aided Decoding of Unequal Error Protection RSC Codes

In Figure 3.23 of Section 3.10, UEP was applied to the H.264 coded video stream by employing different-rate RSC codes to the three different types of H.264 bit-stream partitions. Observe in the schematic of Figure 3.23 that at the receiver, the received signal is QPSK demodulated and forwarded to the RSC decoder, which simply decodes the input signal. This signal is then forwarded to the source decoder of Figure 3.23 in order to reconstruct the source signal. In contrast to the system architecture of Figure 3.23 in Section 3.10, in this section the iterative USSPA technique of Figure 4.6 in Section 4.3.4 is employed in addition to UEP using RSC codes. As seen in Figure 4.6 and described in Section 4.3.4, USSPA is used to exploit the natural residual redundancy of the source-coded bit-stream with the aid of ISCD [188, 195].

To elaborate a little further, in our system the USSPA scheme of Figure 4.6 exploits the residual redundancy, which inherently remains in the H.264 coded bit-stream after source encoding and exploits this to generate extrinsic information. We will demonstrate the presence of residual redundancy in the H.264 encoded video-stream, which becomes evident from the non-uniform M-ary symbol probability distribution $P(s_k), s_k = [b^k(1), b^k(2), ...b^k(M)]$.

Table 4.3: Symbols Probabilities

Symbol	Bit Pattern	Probability
0	000	0.22791640
1	001	0.11727422
2	010	0.11945398
3	011	0.10481402
4	100	0.11365674
5	101	0.10745756
6	110	0.10268064
7	111	0.10674644

In more general terms, this distribution may be obtained by partitioning the H.264 coded bit-stream into a $\log_2 M$-bit/symbol sequence and evaluating the probability of occurrence for all the M $\log_2 M$-bit symbols [227].

For our simulations the H.264 video encoded bit-stream of 300-frame *"Akiyo"* video sequence, 150-frame *"MissAmerica"* video clip and 300-frame *"mother&daughter"* video sequences were used as training sequences. The reason for selecting these specific video sequences is that all of them have relatively slow head-motion without any fast object motion and screen changes, which are typical for video-phone scenarios. The resultant video bit-stream was partitioned into 3-bit segments and the relative frequency of each of the eight 3-bit segments was recorded in Table 4.3.

4.4.1 System Overview

Again, the schematic of the proposed video coding and transmission arrangement is shown in Figure 4.6. The video sequence was compressed using the H.264/AVC video codec, which was configured using the codec parameters described in Section 3.5. Then, the output bit-stream was de-multiplexed into three different-sensitivity bit-streams, namely Steam A, Stream B and Stream C, containing the sequentially concatenated partitions of type A, B and C of all the slices per frame, respectively. The de-multiplexer's binary output sequences x_a, x_b, and x_c are seen in Figure 4.6, where we have a = 1,2, ..., b_a, b = 1,2, ..., b_b, c = 1,2, ..., b_c, and $B = b_a+b_b+b_c$, which are then interleaved using the bit-interleavers Π of Figure 4.6, generating the interleaved sequences $\bar{x}_a$, $\bar{x}_b$ and $\bar{x}_c$ and are encoded by RSC codes having different code-rates.

The unequal error protected and partitioned H.264/AVC bit-stream y_i of Figure 4.6 is QPSK modulated and transmitted over a temporally correlated narrowband Rayleigh fading channel, associated with a normalised

Doppler frequency of $f_D = f_d T_s = 0.01$, where f_d is the Doppler frequency and T_s is the symbol duration. The normalised Doppler frequency of $f_D = 0.01$ represents a relatively slowly fading channel and is commonly used in the literature to characterise the performance of multimedia transmission systems.

The received signal of Figure 4.6 is QPSK demodulated and the soft information $L_M(\hat{y}_a)$ is passed from the QPSK demapper to the respective RSC decoders after de-multiplexing the LLRs into three partitions corresponding to the partitions A, B and C. After RSC decoding the resultant bit-streams are de-interleaved using the respective de-interleavers specified in Figure 4.6. Subsequently, the de-interleaved bit-stream is demultiplexed into its original input order and then decoded by the H.264 decoder to produce the reconstructed video. The decoded video sequence is then used to evaluate the Peak Signal-to-Noise Ratio (PSNR) performance.

The RSC decoder of Figure 4.6 processes the respective input information and feeds back the *a priori* information $L^{apr}_{RSC}(\bar{x})$ from the outer decoder to the inner decoder in order to generate the extrinsic LLR values $L^{extr}_{RSC}(\bar{x})$, which are subsequently deinterleaved by the soft-bit interleavers of Figure 4.6, yielding $L^{apr}_{USSPA}(x)$. Then, the soft bits $L^{apr}_{USSPA}(x)$ are passed to the USSPA, which uses the USSPA algorithm of [188, 195] to compute the *a posteriori* LLR values $L_M(x)$ and the extrinsic LLR values $L^{extr}_{USSPA}(\bar{x})$ for subsequent iterations. The USSPA scheme generates extrinsic information by exploiting the residual redundancy, which inherently remains in the H.264/AVC coded bit-stream after source encoding. Again, the presence of residual redundancy manifests itself in terms of the non-uniform *M-ary* symbol probability distribution $P[s_n(k)]$, $s_n(k)$=[$s_n(1)$,$s_n(2)$, $\cdots$ $s_n(M)$] shown in Table 4.13, which was generated by dividing the H.264/AVC encoded bit-stream into an n=3-bit/symbol sequence, yielding [$b_1(1)$, $b_1(2)$, $b_1(3)$, $b_2(1)$, $b_2(2)$,$\cdots$,$b_k(3)$]=[S_1,S_2, $\cdots$ S_p]. Observe in Figure 4.6 that $L^{apr}_{RSC}(\bar{x})$ is the interleaved version of $L^{extr}_{USSPA}(\bar{x})$, which in turn is generated by processing the *a priori* information $L^{apr}_{USSPA}(x)$ using the outer USSPA. During the iterations the RSC decoder exploits the *a priori* information for the sake of providing improved *a priori* LLR values for the outer soft-bit source decoder, which in turn exploits the input LLR values for the sake of providing improved *a priori* information for the RSC decoder in the subsequent iteration. Further details about iterative decoding are provided in [137].

The different RSC coding rates applied using Equal Error Protection (EEP) and UEP schemes are shown below in Table 4.4(a). The performance of the system is evaluated by keeping the same overall system code rate for each of the EEP and UEP coding schemes. In UEP1 highest level of error protection is provided to Partition A, while in UEP2 high code rate convolutional coding is applied to Partition B to protect it more from channel errors compare to Partition A and C, because based on the Slice Error Sensitivity results, its observed that Partition B is also very important and in case of its corruption it results in significant PSNR degradation because of error propagation to neighbouring coding blocks in both spatial and temporal directions. Also Partition B has the lowest over all size, therefore it does not results in significant increase in over all system bit-rate, as a result reasonable protection can still be provided to Partition A and C.

4.4.2 Performance Results

The performance results obtained for the video system of Figure 4.6 are presented in this section. For the sake of improving the confidence interval of our statistical evaluations, we ran each experiment 160 times and recorded the average results. The BER versus E_b/N_0 curves obtained using the system of Figure 4.6 and

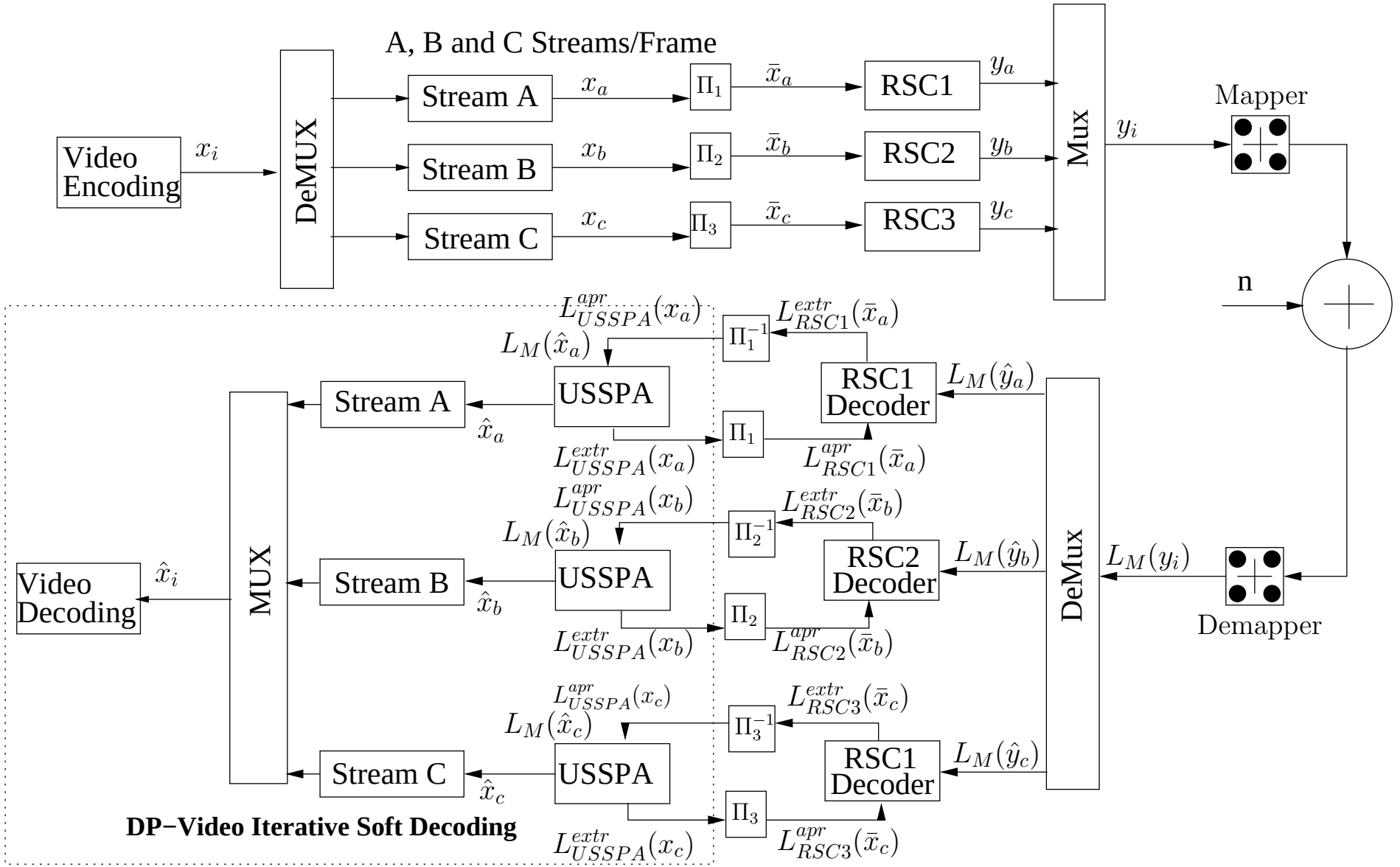

Figure 4.6: USSPA aided DP H.264 UEP iterative decoding with RSC codes while using SP modulation aided DSTS transmission system. All system parameters were summarized in Table 4.4

Error Protection Scheme	Code Rate			
	A	B	C	Overall
EEP	1/2	1/2	1/2	1/2
UEP1	1/3	2/3	3/4	1/2
UEP2	1/2	1/3	2/3	1/2

(a) Code rates for Different Error Protection schemes

System Parameters	**Value**	**System Parameters**	**Value**
Source Coding	H.264/AVC	Video Bit Rate (Kbps)	64
Source Codec	JM 12.1	Video Frame Rate (fps)	15
Video Test Sequence	*Akiyo*	Channel Coded Rate (Kbps)	128
No of Frames	45	Baud-rate (Kbps)	64
Frame Format	QCIF	Channel Coding	RSC
No of 'P' frames between	44	Over-all Code Rate	1/2
two 'I' frames		Code Memory	3
Time Period of 'I' frame (sec)	3	Generator Polynomials	
Use of 1/4-Pixel Motion Estimation	Yes	RSC 1/2,2/3 & 3/4 (G_1, G_2)	$(5,7)_8$
Intra-frame MB update/frame	3	RSC 1/3 (G_1, G_2, G_3)	$(3,7,5)_8$
per QCIF frame		Modulation Scheme	QPSK
Use of 'B' Pictures	No	Number of Transmitters, N_t	1
Use of FMO	No	Number of Receivers, N_r	1
No of Frames Used for	1-Frame	Channel	Correlated Rayleigh Fading
Inter-Frame Motion Search		Normalised Doppler Frequency	0.01
No of Video Slices/frame	9	Interleaver Length	$\approx (64000/15)$
No of Video MBs/Slice	11	No System Iterations I_t	10

(b) Systems parameters used in the schematic of Figure 4.6

Table 4.4: Code rates and systems parameters used in the schematic of Figure 4.6

described in Section 4.4.1, while employing the various error protection scheme of Table 4.4(a) is presented in Figure 4.7. Additionally, the BER versus E_b/N_0 curves evaluated for the individual partitions A, B and C are given in Figure 4.8, which characterise the relative performance of the different partitions depending on the code-rate of the RSC code given in Table 4.4(a). Furthermore, the performance of the error protection scheme applied is characterised in terms of the objective video quality curves in Figure 4.9. It may also be observed from Figure 4.9 that the UEP2 scheme of Table 4.4(a) which has a high error protection for partion B provides a better performance than UEP1, while UEP1 performs better than EEP. Explicitly, an E_b/N_0 gain of 5 dB is attained using the UEP1 scheme of Table 4.4(a) with reference to UEP2, while an E_b/N_0 gain of 3 dB is achieved by UEP2 with reference to the EEP scheme, at the PSNR degradation point of 2 dB.

It may also be observed from the results of Figure 4.9 that the USSPA aided iterative decoder using RSC codes results in a modest performance improvement relative to the otherwise identical scheme dispensing with iterative USSPA, as described in Section 3.10.1. The reason for this relatively modest performance improve-

ment is the lack of adequate amount of residual redundancy in the bit-stream encoded using the sophisticated state-of-the-art H.264 video codec, especially when operated at a low target bit-rate, such as 64 $kbps$, for example. Therefore, in the subsequent sections of this chapter we will elaborate on the inclusion of artifical redundancy in the coded video stream in order to improve the iterative decoding performance. The major findings of the consider error protection schemes at PSNR degradation points of $1dB$ and $2dB$ are summarised in Table 4.5

Table 4.5: Performance Summary Table

Schematic	Parameters	Iterations	Video rate	FEC Code rate	Highest PSNR value	E_b/N_0 at $1dB$ PSNR degradation	E_b/N_0 at $2dB$ PSNR degradation
Figure 4.6	Table 4.4(a)	10	64 $kbps$	0.5	41.7582	EEP = 37 dB UEP1 = 39 dB UEP2 = 35 dB	EEP = 33.5 dB UEP1 = 35 dB UEP2 = 31 dB

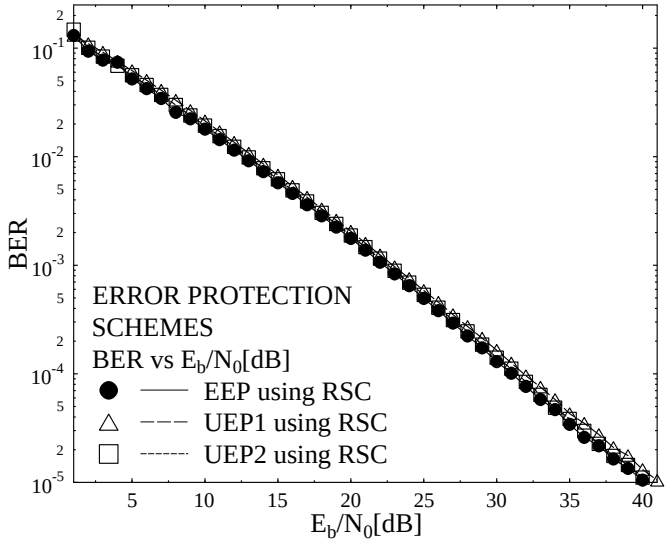

Figure 4.7: BER versus E_b/N_0 for the EEP and UEP schemes of Figure 4.6 using the parameters of Table 4.4

4.5 Video Transmission Using Short Block Code Based Iterative Source-Channel Decoding

In this Section, we analyse the performance gain achieved using SBC based iterative source-channel decoding. We will demonstrate the improved error correction capability of the ISCD using simulation examples, as follows.

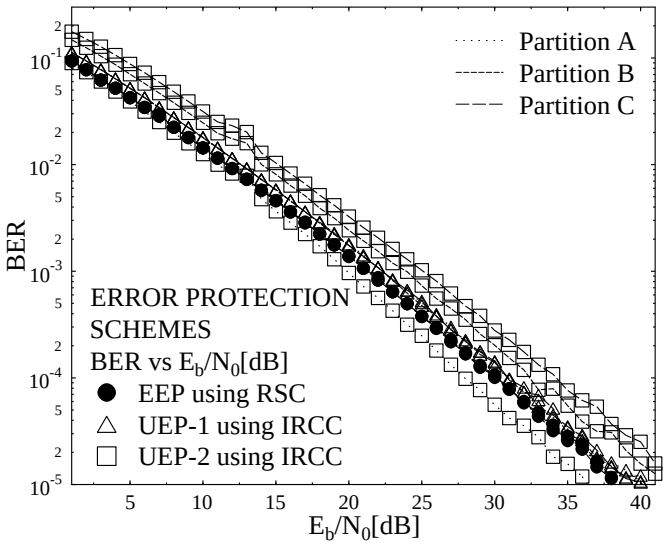

Figure 4.8: BER (A, B and C) versus Eb/N0 for the EEP and UEP schemes of Figure 4.6 using the parameters of Table 4.4

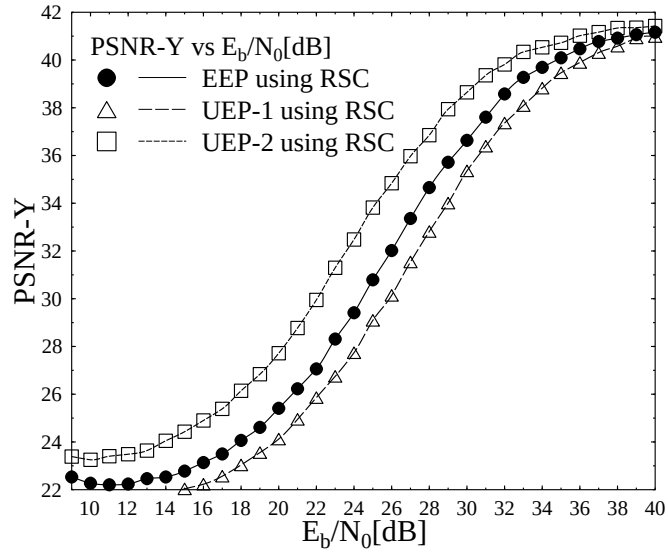

Figure 4.9: PSNR versus E_b/N_0 for EEP and UEP schemes of Figure 4.6 using the parameters of Table 4.4

4.5.1 Iterative Convergence Analysis Using Short Block Codes

The purpose of ISCD is to utilise the constituent inner and outer decoders in order to assist each other in an iterative fashion to glean the highest possible extrinsic information [$L^{extr}_{USSPA}(x')$ and $L^{extr}_{RSC}(\bar{x})$] from each other. In fact, the achievable performance of USSPA is limited by the factor that its achievable iteration gain is actually dependent on the residual redundancy or correlation that remains in the coded bit pattern x_i after limited-complexity, limited-delay, lossy source encoding [1]. However, despite using limited-complexity, limited-delay, lossy compression, the achievable performance improvements of USSPA may remain limited due to the limited residual redundancy in the video-encoded bit-stream, when using the high-compression H.264/AVC video codec. It may be observed from the simulation results of [228] that typically using USSPA for H.264/AVC coded bit-stream results in negligible system performance improvements beyond two decoding iterations. Hence, in order to improve the achievable ISCD performance gain, we *artificially* introduce redundancy in the source coded bit-stream using a technique which we refer to as SBC coding. The novel philosophy of our SBC design is based on exploiting a specific property of Extrinsic Information Transfer (EXIT) Charts [158]. More explicitly, an iterative decoding aided receiver is capable of near-capacity operation at an infinitesimally low decoded BER, if there is an open tunnel between the EXIT curves of the inner and outer decoder components. We will demonstrate that this condition is clearly satisfied, when these two EXIT curves have a point of intersection at the $(I_A, I_B) = (1, 1)$ corner of the EXIT chart, where $I_A = I(x'_i, L_{D,a})$, $0 \leq I_A \leq 1$, is the mutual information between the outer encoded bits x'_i and the LLR values $L_{D,a}$, while $I_E = I(x'_i, L_{D,e})$, $0 \leq I_E \leq 1$, represents the mutual information between the outer channel coded bits x'_i and the LLR values $L_{D,e}$. The *sufficient and necessary* condition for this iterative detection convergence criterion to be met in the presence of perfect *a priori* information, was shown by Kliewer *et al.* [163] to be that the legitimate codewords have a minimum Hamming distance of $d_{H,min} = 2$. Then the ISCD scheme becomes capable of achieving the highest possible source entropy denoted as $H(X) = L^{extr}_{USSPA} = 1\ bit$, provided that the input *a-priori* information of the USSPA is perfect, i.e. we have $H(X) = L^{apri}_{USSPA} = 1\ bit$. At first sight this condition may not seem physically plausible, since without *a priori* information a codeword having a single bit error has a minimum Hamming distance of $d_{H,min} = 1$ from at least two legitimate codewords and hence the decoder remains unable to resolve, which of these two were transmitted. However, in the presence of perfect *a priori* information, i.e. at the $I_A = 1$ point of the EXIT chart this ambiguity may be readily resolved. In fact, this raises the question, why is it not automatically guaranteed that at the $I_A = 1$ point perfect decoding convergence associated with $I_E = 1$ is reached? In other words, what property of a decoding component results in 'mis-information', hence leading to $I_E < 1$ at $I_A = 1$, i.e. to a failure to converge to an infinitesimally low BER even in the presence of perfect *a priori* information at $I_A = 1$? This generic question cannot be readily answered, but again, satisfying the condition of having $d_{H,min} = 2$ is both sufficient and necessary. This motivates the design of the proposed SBC schemes, because it is plausible that using our design procedure all legitimate SBC codewords having a specific mapping-rate equivalent to the reciprocal of the classic code-rate results in a code-table satisfying the condition of $d_{H,min} \geq 2$. Using appropriately designed SBCs it may be guaranteed that the EXIT curve of the combined source codec and SBC block becomes capable of reaching the $(I_A, I_E) = (1, 1)$ point of perfect convergence, regardless of the EXIT-curve shape of the stand-alone source encoder.

Having outlined the theoretical justification for achieving perfect convergence to an infinitesimally low BER, let us now introduce the proposed SBC$_{[K,\ N]}$ encoding algorithms, which maps or encodes each *K-bit* symbol of the source set X to the *N-bit* code words of the SBC set $f(X)$, while maintaining a minimum Hamming distance of $d_{H,min} \geq 2$. According to our SBC$_{[K,\ N]}$ encoding procedure, the video-stream x_k is partitioned into $M = 2^K$*-ary*, or *K-bit* source symbols, each of which has a different probability of occurance and will be alternatively termed as the information word to be encoded into $N = (K + P)$*-bits*, where P represents the number of redundant bits per *K-bit* source symbol.

Algorithm-I:

For $P = 1$, the redundant bit r_τ is generated for the τ*-th M-ary* source symbol by calculating the *exclusive OR* (XOR) function of its *K* constituent bits, as follows:

$$r_\tau = [b^\tau(1) \oplus b^\tau(2) ... \oplus b^\tau(K)], \tag{4.23}$$

where $\oplus$ represents the XOR operation.

The resultant redundant bit can be incorporated in any of the $[K + 1]$ different bit positions, in order to create $[K + 1]$ different legitimate SBC-encoded words, as presented in Table 4.7, each having a minimum Hamming distance of $d_{H,min} = 2$ from all the others. The encoded symbols of the rate-$\frac{2}{3}$, $\frac{3}{4}$, $\frac{4}{5}$ and $\frac{5}{6}$ SBCs along with their corresponding minimum Hamming distance $d_{H,min}$ is summarised in Table 4.8 for the specific case of incorporating the redundant bit r_τ at the end of the τ*-th K-bit* source symbol.

Table 4.6: (K+1) Different SBC combinations using Algorithm-I.

	INPUT	OUTPUT COMBINATIONS			
	Symbols	C_1	C_2	$\cdots$	C_{K+1}
ALGORITHM-I	$S_{(1)}$	$r_1 x_1 x_2 \cdots x_K$	$x_1 r_1 x_2 \cdots x_K$	.	$x_1 x_2 \cdots x_K r_1$
	$S_{(2)}$	$r_2 x_1 x_2 \cdots x_K$	$x_1 r_2 x_2 \cdots x_K$	.	$x_1 x_2 \cdots x_K r_2$
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
	$S_{(2^K)}$	$r_{2^K} x_1 x_2 \cdots x_K$	$x_1 r_{2^K} x_2 \cdots x_K$	.	$x_1 x_2 \cdots x_K r_{2^K}$

Algorithm-II:

For $P = (m \times K)$ with $m \geq 1$, we propose the corresponding SBC$_{[K,\ N]}$-encoding procedure, which results in a gradual increase of $d_{H,min}$ for the coded symbols upon increasing both K and N of the SBC$_{[K,\ N]}$, while the code-rate is fixed. This K to *N-bit* encoding method consists of two steps,

1. STEP-1:
   First $I = [(m - 1) \times K]$ number of redundant bits $r_\tau(i)$, for $i = 1, 2 \cdots I$ are concatenated to the τ*-th K-bit* source symbol by repeated concatenation of K additional source coded bits $(m - 1)$ times, yielding a total of $[(m - 1) \times K]$ bits, as shown in Table 4.7.

2. STEP-2:
   In the Second step the last set of K redundant bits $r_\tau(k)$, for $k = 1, 2 \cdots K$ is generated by calculating the XOR function of the K source bits $b_\tau(j)$, while setting $b_\tau[j = k]$ equal to 0, yielding:

$$r_\tau(k) = [b_\tau(1) \oplus b_\tau(2) ... \oplus b_\tau(K)];\ for\ k = 1, 2 \cdots K,\ while\ setting\ b_\tau(k) = 0, \tag{4.24}$$

   as presented in Table 4.7, where $\oplus$ represents the XOR operation.

Using this method a carefully controlled redundancy is imposed by the specific rate $r = [\frac{K}{N}]$ SBC$_{[K,\ N]}$ to ensure that the resultant N-*bit* codewords exhibit a minimum Hamming distance of $d_{H,min} \geq 2$ between the $M = 2^K$ number of legitimate K-*bit* source code words. This method also results in a gradual increase of the $d_{H,min}$ of the coded symbols upon increasing both K and N of the SBC$_{[K,\ N]}$ considered, as shown in Table 4.8, until the maximum achievable $d_{H,min}$ is reached for the specific SBC coding rate.

Table 4.7: SBC coding procedure using Algorithm-II.

ALGORITHM-II	INPUT	$x_1 x_2 \cdots x_K$				
	STEPS	STEP-1				STEP-2
	Repeated K-bit Concatenation	1^{st}	2^{nd}	$\cdots$	$(m+1)^{th}$	m^{th}
	OUTPUT	$x_1 x_2 \cdots x_K$	$x_1 x_2 \cdots x_K$	$\cdots$	$x_1 x_2 \cdots x_K$	$x'_1 x'_2 \cdots x'_K$
						where we have, $x'_1 = (0 \oplus x_2 \oplus \cdots x_K)$ $x'_2 = (x_1 \oplus 0 \oplus \cdots x_K)$ $\vdots$ $x'_K = (x_1 \oplus x_2 \oplus \cdots x_{(K-1)} \oplus 0)$

4.5.2 Design Example: System Model

The schematic of our proposed videophone arrangement used as our design example for quantifying the performance of various ISCD schemes is shown in Figure 4.10. For the sake of getting insight into the practical system and realistic performance improvement, instead of using a generic model for the source codec parameter set, we employed the H.264/AVC video codec as our source encoder. The compressed video source bit-stream x_k generated using the H.264 video codec is mapped or encoded into the bit-string x'_m. Subsequently the output bit-string is interleaved using the bit-interleaver Π of Figure 4.10, yielding the interleaved sequence $\bar{x}_m$, which is then encoded by a RSC code having a specific code rate given in Table 4.8.

Interleaving and de-interleaving constitute an important step in the iterative decoder of Figure 4.10. As stated in Section 4.2.2, the job of interleaver is to ensure that the bits are input in their expected original order to the component decoders and ascertaining that the statistical independence of the extrinsic LLRs is retained.

Since the degree of statistical independence guaranteed by an interleaver is always related to its length [185], instead of performing the ISCD operation on the various frame slices independently, we concatenated all the bits generated by each type of partition for the different Macro-Blocks (MBs) within each slice of a given

frame, which results in a longer interleaver without extending the video delay and hence improves the achievable performance of iterative decoding. The resultant bit-stream is QPSK modulated and transmitted over a temporally correlated narrowband Rayleigh fading channel, associated with the normalised Doppler frequency of $f_d = f_D T_s = 0.01$, where f_D is the Doppler frequency and T_s is the symbol duration. At the receiver the signal is QPSK demodulated and the resultant soft-information is passed to the RSC decoder. The extracted extrinsic information is then exchanged between the USSPA and RSC decoders of Figure 4.10 [188]. Following QPSK demodulation at the receiver the soft information is extracted in the form of its LLR representation $D_{M,a}$. This soft-information $D_{M,a}$ is forwarded to the RSC inner decoder, which processes it along with the *a priori* information $L_{M,a}$ fed back from the outer decoder of Figure 4.10 in order to generate the *a posteriori* LLR values $L_{M,p}$. The *a priori* LLR values $L_{M,a}$ fed back from the outer decoder are subtracted from the inner decoder's output *a posteriori* LLR values $L_{M,p}$ to generate the extrinsic LLR values $L_{M,e}$, which are subsequently deinterleaved by the soft-bit interleaver of Figure 4.10, yielding the soft-bits $L_{D,a}$ that are input to the outer decoder to compute the *a posteriori* LLR value $L_{D,p}$. Observe in Figure 4.10 that $L_{D,e}$ is generated by subtracting the *a priori* information $L_{D,a}$ from the *a posteriori* information $L_{D,p}$, which in turn results in $L_{M,a}$ after interleaving. During iterative decoding the outer decoder exploits the input LLR values for the sake of providing improved *a priori* information for the inner channel decoder of Figure 4.10, which in turn exploits the fed back *a priori* information in the subsequent iteration for the sake of providing improved *a posteriori* LLR values for the outer decoder. Further details about iterative decoding are provided in [137].

4.5.3 H.264 Coded Video Stream Structure

The H.264/AVC codec's structure is notionally divided into the Video Coding Layer (VCL) and the Network Abstraction Layer (NAL) [31]. The hybrid video coding functions designed for improved coding efficiency are part of the VCL, while the NAL is designed for improved network adaptation and is responsible for the reliable transport of the resultant bit-stream generated by VCL over a wide range of transport layer protocols. The H.264/AVC generates a number of video-frame slices, which are formed by an integer number of consecutive MBs of a picture. The number of MBs per slice may vary from a single one to all MBs of a picture in a given slice. Error resilient DP [112] has been incorporated in the H.264/AVC codec in order to mitigate the effects of channel errors. In the H.264/AVC codec DP results in three different types of streams per slice, which are referred to as Type A, B and C partitions, each containing specific sets of coding parameters having different levels of importance. Additionally, the H.264 coded stream contains information related to the Group of Pictures (GOP) sequence and a so-called picture parameter set containing information related to all the slices of a single picture. All these different syntax elements are contained in Network Abstraction Layer Units (NALUs), consisting of 1-byte header and a payload of a variable number of bytes containing the coded symbols of the corresponding H.264 syntax element. The type of data contained by the NALU is identified by the 5-bit NALU type field contained in the NALU header.

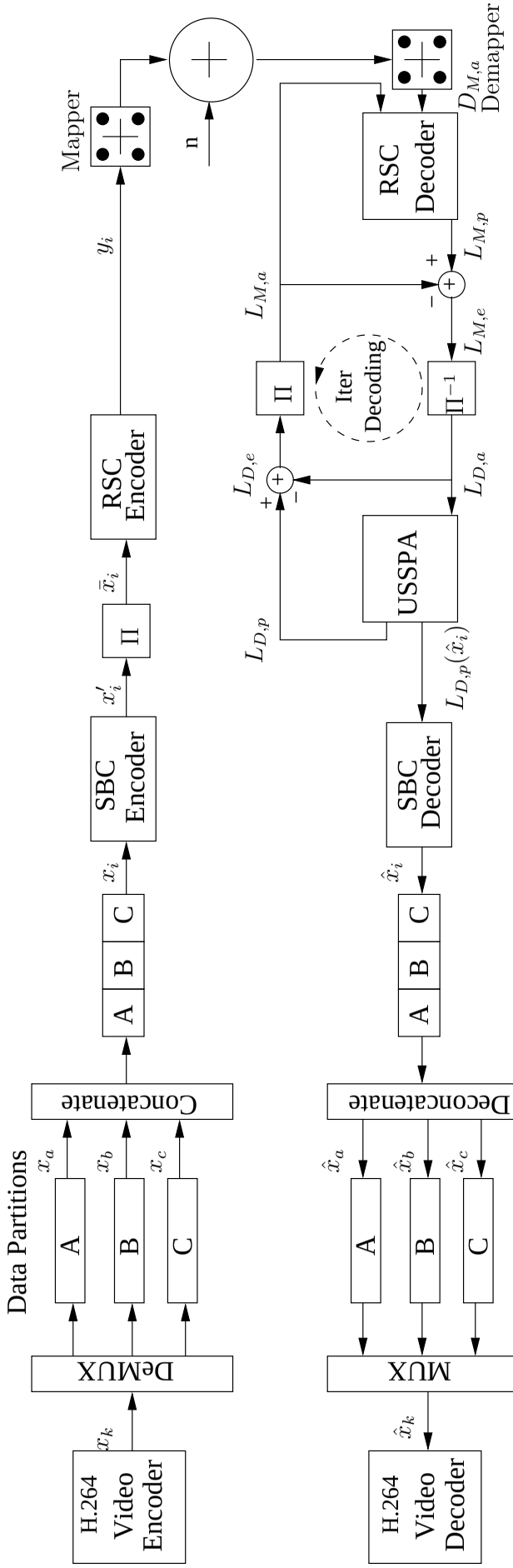

Figure 4.10: The proposed system model. All system parameters were summarised in Table 4.9. The difference in comparison to Figure 4.6 includes the concatenation of stream A, B and C to form a single long stream. Additionally, various rate SBCs were incorporated along with diverse RSC codes which resulted in the overall code rate is 1/4.

4.5.4 The Proposed Short Block Codes

Let us now demonstrate the power of ISCD and the effect of our proposed SBCs on the performance of ISCD with the aid of a design example. As an example, the $SBC_{[K,\ N]}$-encoded symbols generated by applying rate-$\frac{2}{3}$, $\frac{3}{4}$, $\frac{4}{5}$, $\frac{5}{6}$ and rate-$\frac{1}{3}$ coding schemes generated using Algorithm-I and Algorithm-II are detailed in Table 4.8, along with their corresponding minimum Hamming distances $d_{H,min}$. Again, as it becomes evident from Table 4.8, the EXIT-chart optimised SBCs ensure that the encoded N-bit symbols exhibit a minimum Hamming distance of $d_{H,min} \geq 2$. Additionally, only 2^K out of the 2^N possible *N-bit* symbols are legitimate in the mapped source coded bit-stream, which exhibits a non-uniform probability of occurance for the *N-bit* source symbols. Figure 4.11 depicts the EXIT characteristics of the USSPA scheme of Figure 4.10 using either the rate-1[4] or the rate < 1 $SBC_{[K,\ N]}$ schemes of Algorithm-I and II shown in Table 4.8. More specifically, the EXIT curve of USSPA using rate-$\frac{1}{3}$, $\frac{2}{3}$, $\frac{3}{4}$, $\frac{4}{5}$ and $\frac{5}{6}$ SBCs does indeed reach to the top right corner of the EXIT chart at $(I_A, I_E) = (1, 1)$ and hence results in an infinitesimally low BER. By contrast, the USSPA scheme using a rate-1 SBC, i.e. no SBC fails to do so. In conclusion, our simulation results recorded for the system presented in Figure 4.10 reveal that the performance of USSPA strongly depends on the presence or absence of residual source redundancy, which typically manifests itself in the form of non-uniform probability of occurance for the *N-bit* source coded symbols. The coding parameters of the different SBC schemes used in our design example are shown in Table 4.9(a). We considered a concatenated rate $R = \frac{1}{4}$ RSC encoder with constraint length $L = 4$ and generator sequences $g_1 = [1011]$, $g_2 = [1101]$, $g_3 = [1101]$ and $g_4 = [1111]$ represented as $G = [1, g2/g1, g3/g1, g4/g1]$, where '1' denotes the systematic output, the first output g_1 is fed back to the input and g_2, g_3, g_4 denotes the feed forward output of the RSC encoder. Observe from the table that an overall code-rate of $R = \frac{1}{4}$ was maintained by adjusting the puncturing rate of the concatenated RSC in order to accommodate the different SBC rates of Table 4.9(a).

4.5.5 Exit Chart Analysis

At the receiver seen in Figure 4.10, iterative soft-bit source and channel decoding is applied by exchanging extrinsic information between the receiver blocks, which has the capability of improving the achievable subjective video quality. EXIT charts were utilised to characterise the mutual information exchange between the input and output of both the inner and outer components of an iterative decoder and hence to analyse its decoding convergence behaviour. Additionally, the actual decoding trajectories acquired while using various SBCs generated using Algorithm-I and II were presented by recording the mutual information at the input and output of both the inner and outer decoder during the bit-by-bit Monte-Carlo simulation of the iterative USSPA algorithm.

Figures 4.12, 4.13, 4.14 and 4.15 present the decoding trajectories recorded both at $E_b/N_0 = 0\ dB$ and $-1\ dB$, when employing the rate-$\frac{2}{3}$, $\frac{3}{4}$, $\frac{4}{5}$ and $\frac{5}{6}$ SBCs of Algorithm-I as the outer code along with their corresponding rate-$\frac{3}{8}$, $\frac{1}{3}$, $\frac{5}{16}$ and $\frac{3}{10}$ RSC, respectively. Observe from the decoding trajectories that the convergence behaviour of the considered SBCs degrades upon increasing their coding rate. Furthermore, the decoding trajectories obtained by employing various rate-$\frac{1}{3}$ outer SBCs of type $SBC_{[2,\ 6]}$ and $SBC_{[5,\ 15]}$, which were generated using *Algorithm-II* as well as using the rate-$\frac{3}{4}$ inner RSC detailed in Table 4.8 was recorded at

[4]For the sake of using a unified terminology, we refer to the scheme using no SBC as the rate-1 SBC.

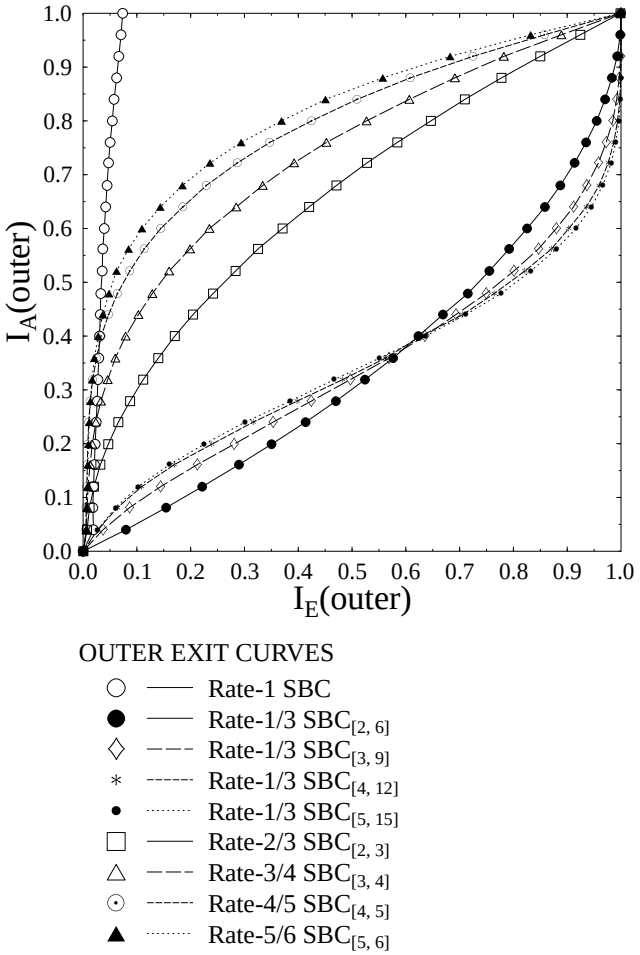

Figure 4.11: EXIT characteristics of the USSPA scheme of Figure 4.10 using the different SBCs of Table 4.8.

$E_b/N_0 = -4\ dB$ and $-4.5\ dB$, as portrayed in Figures 4.16 and 4.19, respectively. It may be observed from the EXIT trajectories of Figures 4.16, and 4.19 that as expected, the convergence behaviour of SBCs improves upon increasing $d_{H,min}$.

4.5.6 System Performance Results

In this Section we present our performance results for the proposed system. A 45 frame *"Akiyo"* video sequence [1] in (176x144)-pixel Quarter Common Intermediate Format (QCIF) was used as our test sequence and was encoded using the H.264/AVC JM 13.2 reference video codec at 15 frames-per-second (fps) at the target bit-rate of 64 $Kbps$. Using H.264/AVC codec each QCIF frame was partitioned into 9 slices and each slice was composed of 11 MBs. The resultant video encoded clip consisted of an intra-coded 'I' frame followed by 44 predicted or 'P' frames, corresponding to a lag of 3 seconds between the 'I' frames at a frame-rate of 15 fps. The periodic insertion of 'I' frames curtailed error propagation beyond 45 frames.

Additional source codec parameters were set as follows,

Table 4.8: Different SBCs with corresponding symbols and minimum Hamming distances [$d_{H,min}$]

SBC Type		Symbols in Decimal	$d_{H,min}$
	Rate1 SBC	{0,1}	1
Algorithm-I	Rate-$\frac{2}{3}$ SBC$_{[2,\ 3]}$	{0,3,5,6}	2
	Rate-$\frac{3}{4}$ SBC$_{[3,\ 4]}$	{0,3,5,6,9,10,12,15}	2
	Rate-$\frac{4}{5}$ SBC$_{[4,\ 5]}$	{0,3,5,6,9,10,12,15,17,18,20,23,24,27,29,30}	2
	Rate-$\frac{5}{6}$ SBC$_{[5,\ 6]}$	{0,3,5,6,9,10,12,15,17,18,20,23,24,27,29,30, 33,34,36,39,40,43,45,46,48,51,53,54,57,58,60,63}	2
Algorithm-II	Rate-$\frac{1}{2}$ SBC$_{[2,\ 4]}$	{0,6,9,15}	3
	Rate-$\frac{1}{2}$ SBC$_{[3,\ 6]}$	{0,14,21,27,35,45,54,56}	4
	Rate-$\frac{1}{2}$ SBC$_{[4,\ 8]}$	{0,30,45,51,75,85,102,120,135,153,170,180, 204,210,225,255}	5
	Rate-$\frac{1}{3}$ SBC$_{[2,\ 6]}$	{0,22,41,63}	3
	Rate-$\frac{1}{3}$ SBC$_{[3,\ 9]}$	{0,78,149,219,291,365,438,504}	4
	Rate-$\frac{1}{3}$ SBC$_{[4,\ 12]}$	{0,286,557,819,1099,1365,1638,1912,2183, 2457,2730,2996,3276,3538,3809,4095}	5
	Rate-$\frac{1}{3}$ SBC$_{[5,\ 15]}$	{0,1086,2141,3171,4251,5285,6342,7416,8471,9513, 10570,11636,12684,13746,14801,15855,16911,17969, 19026,20076,21140,22186,23241,24311,25368,26406, 27461,28539,29571,30653,31710,32736}	6

- All macroblock types were enabled;
- No multiframe prediction was used;
- No B-slices were used;
- Universal Variable Length Coding (UVLC) type entropy coding was used;
- Error concealment was performed using the motion vector recovery algorithm of [76];

To control the effects of error propagation, we incorporated error resilience features, such as DP and intra-frame coded MB updates of three randomly distributed MBs per frame. The insertion of 'B' pictures was avoided because it results in an unacceptable loss of lip-synchronisation as a result of the corresponding delay incurred due to the bi-directionally predicted video coding operations [76]. Additionally, only the immediately preceding frame was used for motion search, which results in a reduced computational complexity compared to using multiple reference frames. These video coding parameters were chosen, bearing in mind that the error-resilience of the DP aided H.264/AVC stream is directly related to the number of 'P' frames inserted between two consecutive 'I' frames.

The remaining error resilient encoding techniques, such as the employment of multiple reference frames for inter-frame motion compensation and Flexible Macro-block Ordering (FMO) [81] were turned off, because they typically result in modest video performance improvements in low-motion head-and-shoulders video sequences, such as the *"Akiyo"* clip, despite their substantially increased complexity. These encoder settings

Scheme	Code Rate			
	SBC Type	RSC	SBC	Overall
	Rate1 SBC	1/4	1	1/4
Algorithm-I	Rate-$\frac{2}{3}$ SBC	3/8	2/3	1/4
	Rate-$\frac{3}{4}$ $SBC_{[3,\ 4]}$	1/3	3/4	1/4
	Rate-$\frac{4}{5}$ $SBC_{[4,\ 5]}$	5/16	4/5	1/4
	Rate-$\frac{5}{6}$ $SBC_{[5,\ 6]}$	3/10	5/6	1/4
Algorithm-II	Rate-$\frac{1}{3}$ $SBC_{[2,\ 6]}$	3/4	1/3	1/4
	Rate-$\frac{1}{3}$ $SBC_{[3,\ 9]}$	3/4	1/3	1/4
	Rate-$\frac{1}{3}$ $SBC_{[4,\ 12]}$	3/4	1/3	1/4
	Rate-$\frac{1}{3}$ $SBC_{[5,\ 15]}$	3/4	1/3	1/4

(a) Code rates for Different Error Protection schemes

System Parameters	**Value**	**System Parameters**	**Value**
Source Coding	H.264/AVC	Video Bit Rate (Kbps)	64
Source Codec	JM 12.1	Video Frame Rate (fps)	15
Video Test Sequence	*Akiyo*	Channel Coded Rate (Kbps)	256
No of Frames	45	Baud-rate (Kbps)	128
Frame Format	QCIF	Channel Coding	RSC
No of 'P' frames between	44	Over-all Code Rate	1/4
two 'I' frames		Code Memory	4
Time Period of 'I' frame (sec)	3	Generator Polynomials	
Use of 1/4-Pixel Motion Estimation	Yes	RSC 1/4 (G_1, G_2, G_3, G_4)	$(13, 15, 15, 17)_8$
Intra-frame MB update/frame	3		
per QCIF frame		Modulation Scheme	QPSK
Use of 'B' Pictures	No	Number of Transmitters, N_t	1
Use of FMO	No	Number of Receivers, N_r	1
No of Frames Used for	1-Frame	Channel	Correlated Rayleigh Fading
Inter-Frame Motion Search		Normalised Doppler Frequency	0.01
No of Video Slices/frame	9	Interleaver Length	$\approx (64000/15)$
No of Video MBs/Slice	11	No System Iterations I_t	10

(b) Systems parameters used in the schematic of Figure 4.10

Table 4.9: Code rates and systems parameters used in the schematic of Figure 4.10

result in a reduced encoder complexity and in a realistic real-time implementation. The remaining system parameters are listed in Table 4.9(b).

Moreover, since hand-held videophones have to have a low complexity, we limited the number of iterations between the RSC and USSPA decoders to $I_t = 10$, when using a rate-1 SBC $-$ i.e. no SBC. Similarly, we used $I_t = 10$ iterations, when applying SBCs having a rate below unity. For the sake of increasing the confidence in our results, we repeated each 45-frame experiment 160 times and averaged the results generated. A range of different SBCs generated using our proposed *Algorithm-I* and *II* are given in Table 4.8, which are used as

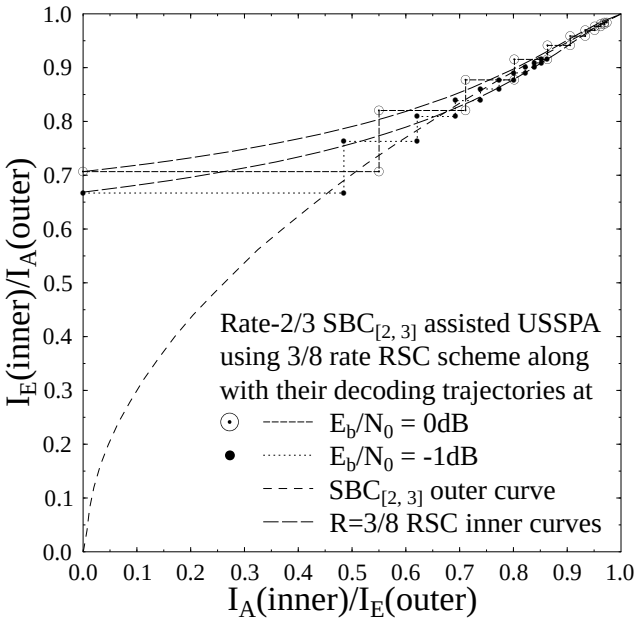

Figure 4.12: The EXIT chart and simulated decoding trajectories of the $\text{SBC}_{[2,\ 3]}$ scheme of Figure 4.10 using the parameters of Table 4.9, at E_b/N_0 = 0 and -1 dB.

the outer codes of Figure 4.10 in order to evaluate their achievable system performance improvements. We evaluated the performance of our proposed system by keeping the same overall code rate as well as video rate for the different considered error protection schemes.

Figure 4.20 presents the performance of the various rate-$\frac{2}{3}$, $\frac{3}{4}$, $\frac{4}{5}$ and $\frac{5}{6}$ SBCs along with the rate-$\frac{1}{3}$ SBC based error protection schemes of Table 4.8 in terms of the attainable BER, while their comparison with the rate-1 SBC based schemes is offered in Figures 4.22. The major findings of the consider error protection schemes at PSNR degradation points of $1dB$ and $2dB$ are summarised in Table 4.10

Additionally, the performance trends expressed in terms of the $PSNR$ versus E_b/N_0 curves are portrayed in Figures 4.21 and 4.23. It may be observed in Figure 4.21 that the $\text{SBC}_{[5,\ 15]}$ scheme having $d_{H,min} = 6$ provides the best $PSNR$ performance among the eight different SBC schemes of Table 4.8 across the entire E_b/N_0 region considered. Furthermore, observe from Figure 4.21 that the lowest rate-$\frac{2}{3}$ outer SBC combined with rate-$\frac{3}{8}$ inner RSC results in the best PSNR performance, outperforming the rate-$\frac{3}{4}$, $\frac{4}{5}$ and $\frac{5}{6}$ SBCs generated using Algorithm-I. It may also be observed in Figure 4.23 that using USSPA in conjunction with the rate-1 outer SBC and rate-$\frac{1}{4}$ inner RSC results in a worse $PSNR$ performance than the outer SBCs having a less than unity rate combined with the corresponding inner RSC of Table 4.8, while maintaining the same overall code rate. Quantitatively, using the SBCs of Table 4.8 having a rate lower than 1, an additional E_b/N_0 gain of upto 25 dB may be achieved over the rate-1 SBC at the PSNR degradation point of 1 dB.

Finally, the achievable subjective video qualities of the video telephone schemes utilising various types of SBCs generated using *Algorithm-I* and *II* is presented in Figures 4.24 and 4.25, respectively. In order to

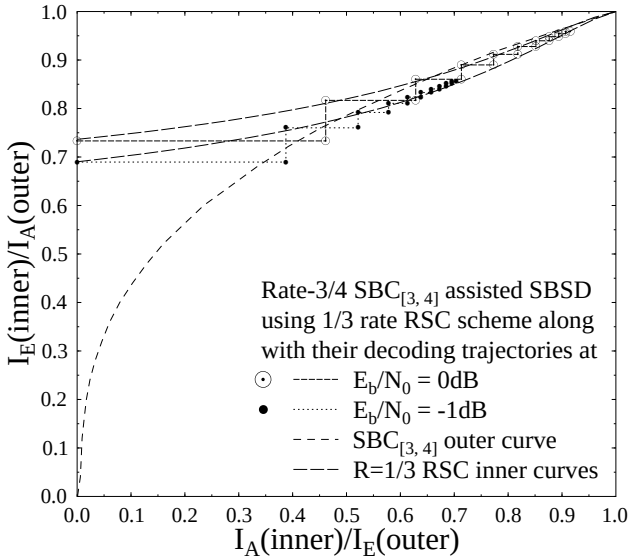

Figure 4.13: The EXIT chart and simulated decoding trajectories of the $SBC_{[3,\,4]}$ scheme of Figure 4.10 using the parameters of Table 4.9, at E_b/N_0 = 0 and -1 dB.

have a fair subjective video quality comparison, we averaged both the luminance and chrominance components of the 30 video test sequences, decoded using the H.264 video codec for each type of setup. The achievable subjective video quality recorded at the channel E_b/N_0 value of 0.5 dB using rate-$\frac{2}{3}$, $\frac{3}{4}$, $\frac{4}{5}$, and $\frac{5}{6}$ SBCs of $Algorithm-I$ can be seen in Figure 4.24. Observe from Figure 4.24 that the achievable video quality improves upon decreasing the SBC code rate. Similarly, Figure 4.25 presents the subjective video quality obtained at (from left to right) $E_b/N_0 = -4.1\ dB$, $-3.9\ dB$, $-3.0\ dB$ and $-2.1\ dB$ using rate-$\frac{1}{3}$ SBCs of the type (from top to bottom) $SBC_{[2,\,6]}$, $SBC_{[3,\,9]}$, $SBC_{[4,\,12]}$ and $SBC_{[5,\,15]}$. Observed from Figure 4.25 that a nearly unimpaired quality is obtained for the rate-$\frac{1}{3}$ SBCs having (from top) $d_{H,min}$ =3, 4, 5 and 6 at E_b/N_0 values of $-2.5\ dB$, $-3.0\ dB$, $-3.9\ dB$ and $-4.1\ dB$, respectively. This implies that the subjective video quality of the system improves upon increasing $d_{H,min}$ of the SBCs employed.

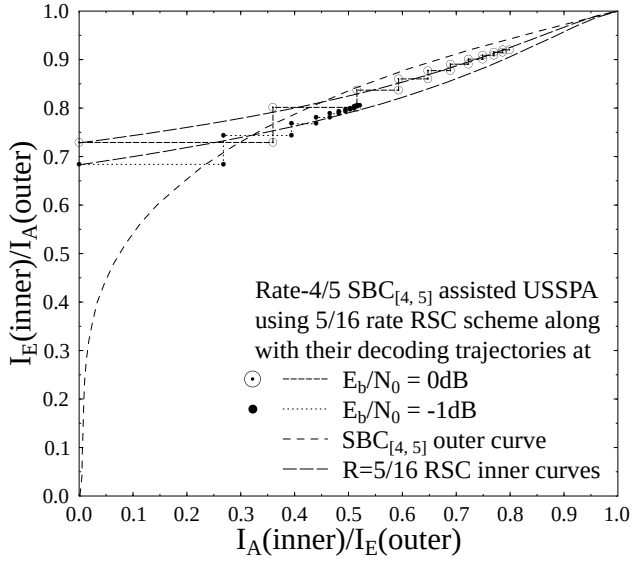

Figure 4.14: The EXIT chart and simulated decoding trajectories of the $SBC_{[4,\ 5]}$ scheme of Figure 4.10 using the parameters of Table 4.9, at $E_b/N_0 = 0$ and -1 dB.

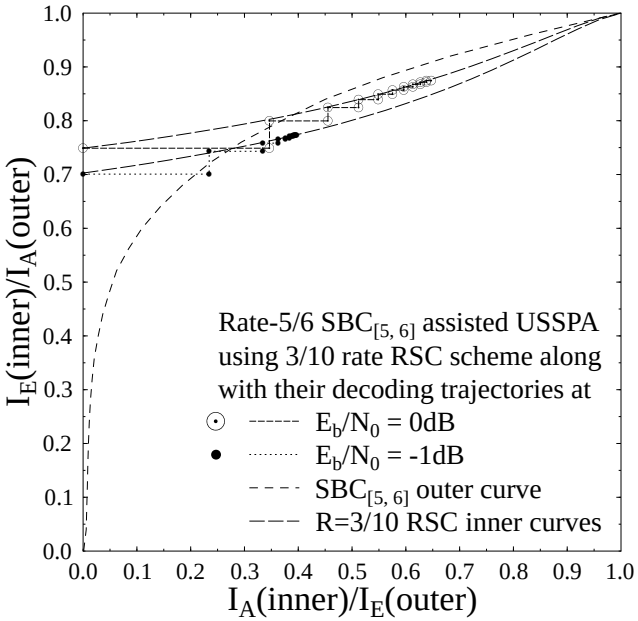

Figure 4.15: The EXIT chart and simulated decoding trajectories of the $SBC_{[5,\ 6]}$ scheme of Figure 4.10 using the parameters of Table 4.9, at $E_b/N_0 = 0$ and -1 dB.

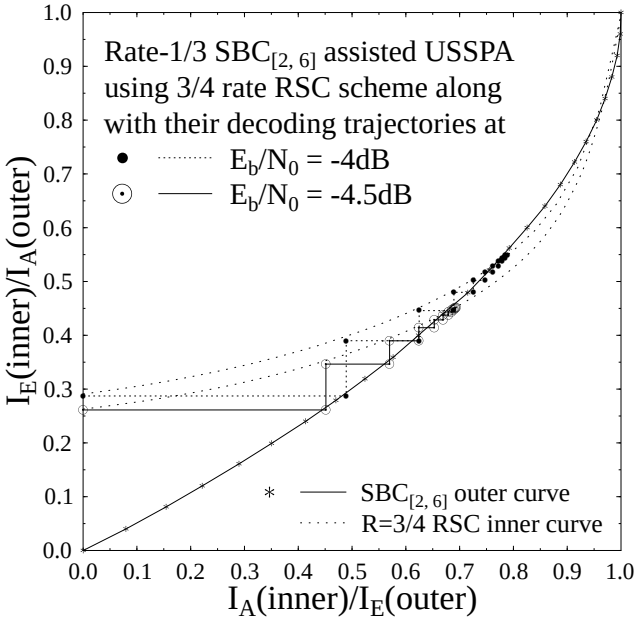

Figure 4.16: The EXIT chart and simulated decoding trajectories of the $SBC_{[2,\ 6]}$ scheme of Figure 4.10 using the parameters of Table 4.9, at $E_b/N_0 = -4$ and -4.5 dB.

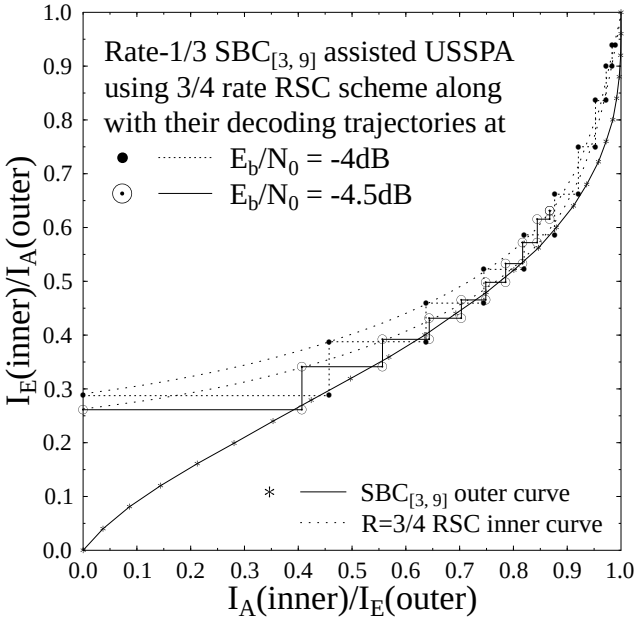

Figure 4.17: The EXIT chart and simulated decoding trajectories of the $SBC_{[3,\ 9]}$ scheme of Figure 4.10 using the parameters of Table 4.9, at $E_b/N_0 = -4$ and -4.5 dB.

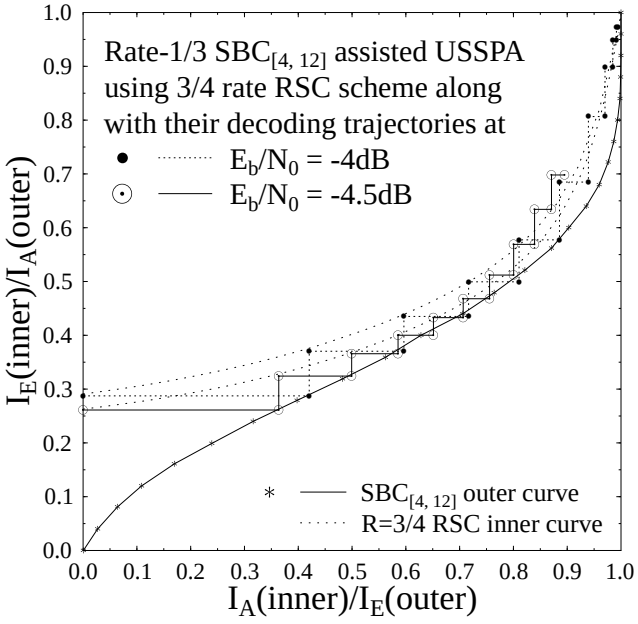

Figure 4.18: The EXIT chart and simulated decoding trajectories of the $SBC_{[4,\ 12]}$ scheme of Figure 4.10 using the parameters of Table 4.9, at $E_b/N_0 = -4$ and -4.5 dB.

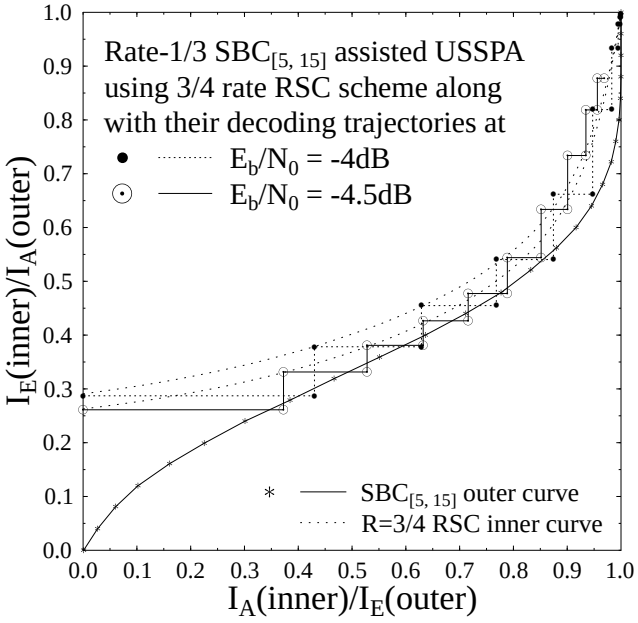

Figure 4.19: The EXIT chart and simulated decoding trajectories of the $SBC_{[5,\ 15]}$ scheme of Figure 4.10 using the parameters of Table 4.9, at $E_b/N_0 = -4$ and -4.5 dB.

Table 4.10: Performance Summary Table for the shematic of Figure 4.10

Schematic	Parameters	Iterations	Video rate	FEC Code rate	Highest PSNR value	SBC-rate	$E_b/N_0[dB]$ at $1dB$ PSNR degradation	$E_b/N_0[dB]$ at $2dB$ PSNR degradation
Figure 4.10	Table 4.9(a)	10	64 $kbps$	0.25	41.7582	$\frac{2}{3}$	3.0	1.5
						$\frac{3}{4}$	3.5	2.5
						$\frac{4}{5}$	4.0	3.0
						$\frac{5}{6}$	5.0	3.5
						$\frac{2}{6}$	-1.7	-2.0
						$\frac{3}{9}$	-3.2	-3.5
						$\frac{4}{12}$	-3.7	-3.8
						$\frac{5}{15}$	-4.0	-4.1
						Rate-1	23	21

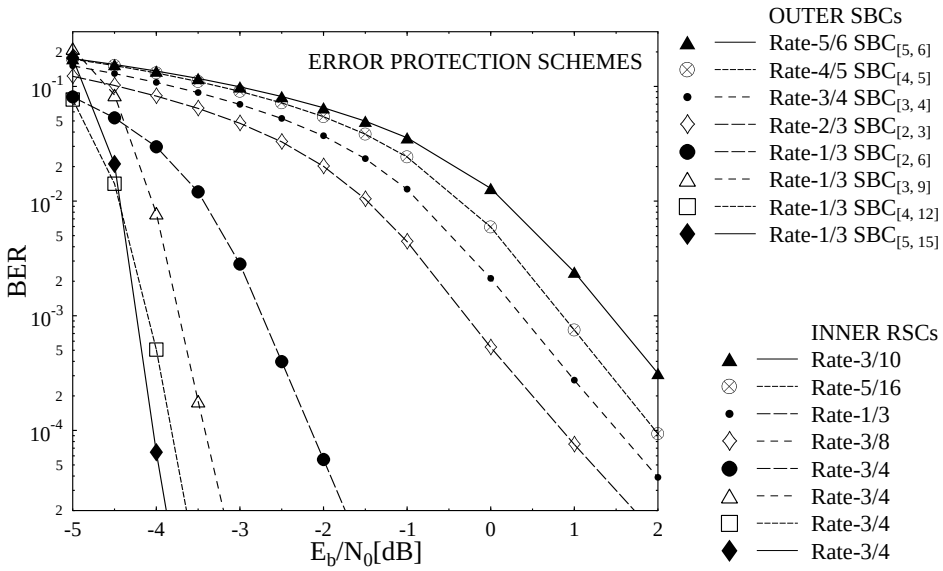

Figure 4.20: BER versus E_b/N_0 performance of the various error protection schemes of Figure 4.10 using the parameters of Table 4.9

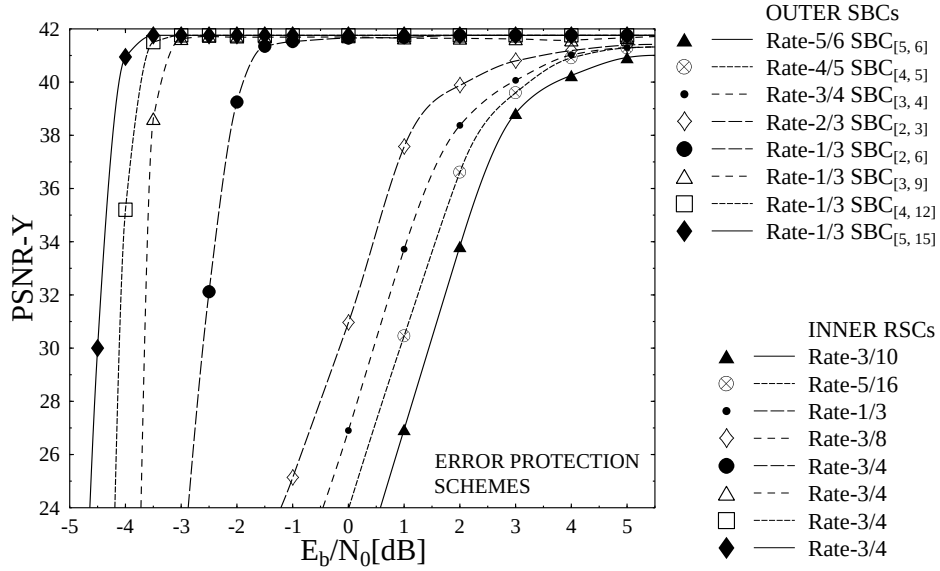

Figure 4.21: PSNR-Y versus E_b/N_0 performance of the various error protection schemes of Figure 4.10 using the parameters of Table 4.9

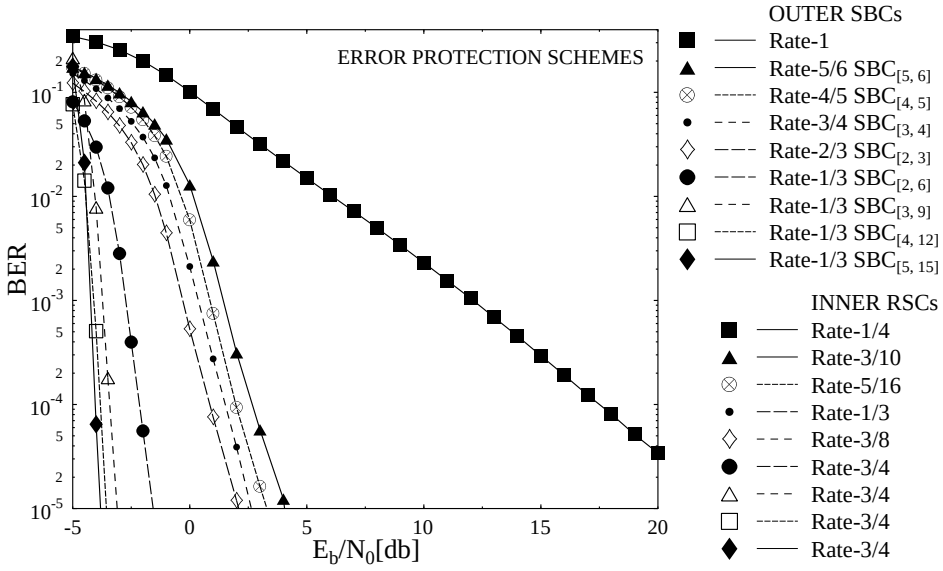

Figure 4.22: BER versus E_b/N_0 performance of the various error protection schemes of Figure 4.10 using the parameters of Table 4.9

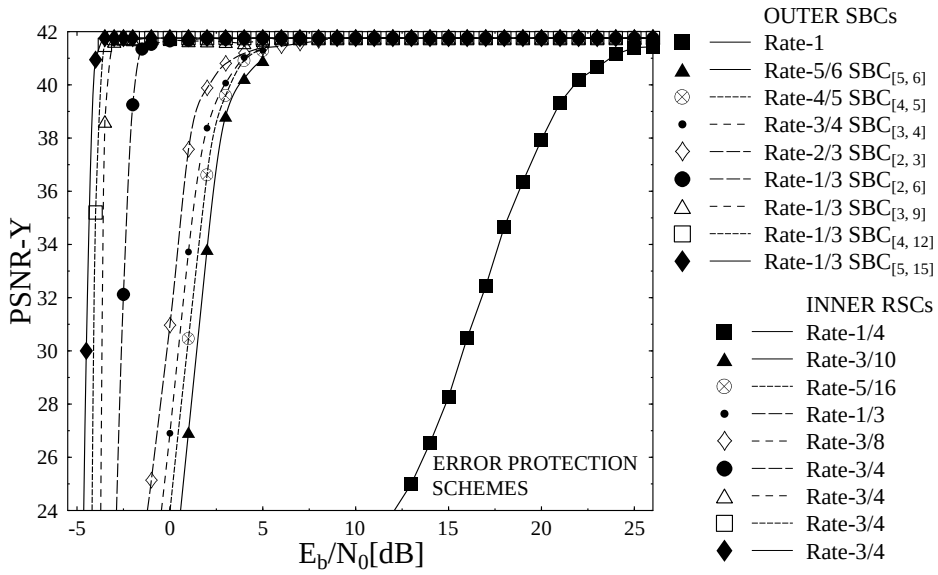

Figure 4.23: PSNR-Y versus E_b/N_0 performance of the various error protection schemes of Figure 4.10 using the parameters of Table 4.9

Figure 4.24: Subjective video quality of the 45^{th} *"Akiyo"* video sequence frame using (from left) Rate-$\frac{2}{3}$, $\frac{3}{4}$, $\frac{4}{5}$ and $\frac{5}{6}$ SBCs of Algorithm-I summarised in Table 4.8 at E_b/N_0=0.5 dB.

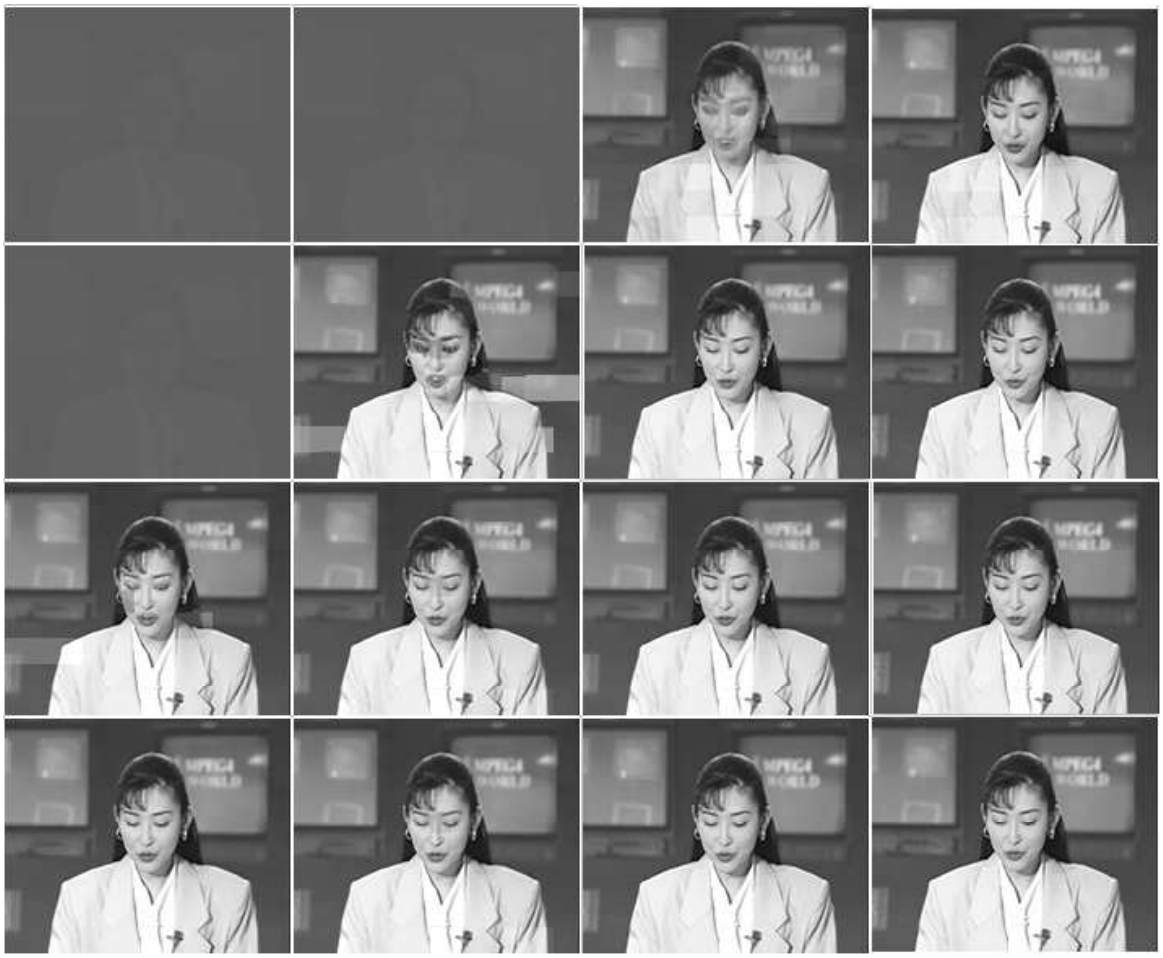

Figure 4.25: Subjective video quality of the 45^{th} *"Akiyo"* video sequence frame using Rate-$\frac{1}{3}$ SBCs of type (from top) $SBC_{[2,\ 6]}$, $SBC_{[3,\ 9]}$, $SBC_{[4,\ 12]}$ and $SBC_{[5,\ 15]}$ of Algorithm-II summarised in Table 4.8 at (from left) E_b/N_0=-4.1 dB, -3.9 dB, -3 dB and -2.1 dB.

4.6 Performance Analysis of EXIT-Chart Optimised SBCs Using Rate-1 Inner Precoder

In this section we analyse the performance of our proposed SBC *Algorithm-II* using Rate-1 precoder as inner code. The details about this system are given below.

4.6.1 System Overview

The structure of our proposed ISCD videophone arrangement used as our design example for quantifying the performance of various SBC schemes is shown in Figure 4.26. At the transmitter side the H.264 video codec is used to compress the source video sequence and the generated bit-stream x_k is mapped or encoded into the bit-string x'_i by employing the SBC scheme. Subsequently, the encoded bit-string x'_i is interleaved using the bit-interleaver Π of Figure 4.10, yielding the interleaved sequence $\bar{x}_i$, which is encoded by the rate-1 precoder of Table 4.11(a) before transmission.

In the iterative decoder of Figure 4.10, the extent of the statistical independence of the extrinsic information provided by an interleaver is always related to its length [185]. Therefore instead of performing the ISCD operation on the various frame slices independently, we concatenated all the bits generated by the MBs of the slices within a given frame, which results in a longer interleaver and hence improves the attainable performance of iterative decoding without extending the video delay. The precoded bit-stream y_i is QPSK modulated and transmitted over a temporally correlated narrowband Rayleigh fading channel, associated with the normalised Doppler frequency of $f_d = f_D T_s = 0.01$, where f_D is the Doppler frequency and T_s is the symbol duration. At the receiver the soft-information obtained after QPSK demodulation is forwarded to the inner decoder. The extracted extrinsic information is then exchanged between the inner and outer decoder of Figure 4.10, in order to attain the lowest possible BER [188].

4.6.2 Design Example

The coding parameters of the different SBC schemes used in our design example are shown in Table 4.11(a). Observe from the table that an overall code-rate of $R = \frac{1}{3}$ was maintained by using a concatenated rate-1 precoder in order to accommodate the different rate-$\frac{1}{3}$ SBCs of Table 4.8. Moreover, for the rate-1 SBC benchmarker scheme the rate-$\frac{1}{3}$ concatenated RSC is used as inner code in order to achieve an iteration gain, while keeping the overall bit-rate budget constant.

4.6.3 EXIT Chart Analysis

The actual decoding trajectories of the various error protection schemes employing the different rate-$\frac{1}{3}$ outer SBCs as well as using the rate-1 inner precoder detailed in Table 4.11(a) were recorded both at $E_b/N_0 = -2\,dB$ and $-2.5\,dB$, as portrayed in Figures 4.27, 4.28, 4.29, and 4.30, respectively. These trajectories were recorded by acquiring the mutual information at the input and output of both the inner and outer decoder during the

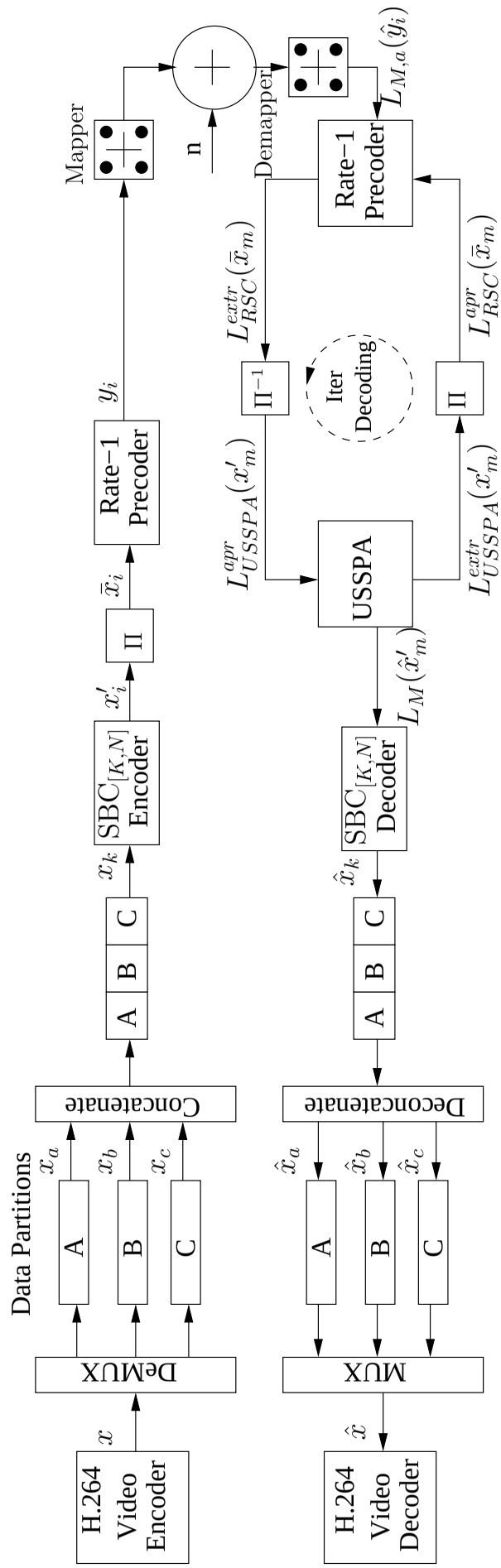

Figure 4.26: The Proposed ISCD System Model. All system parameters were summarized in Table 4.11. The difference in comparison to Figure 4.10 is that a rate-1 inner precoder and a rate-1/3 outer SBC was included, which resulted in the overall code rate is 1/3.

Error Protection Scheme	Code Rate		
	Outer Code (SBC)	Inner Code	Overall
$SBC_{[2,\ 6]}$	Rate-$\frac{1}{3}$ SBC	Rate-1 Precoder	1/3
$SBC_{[3,\ 9]}$	Rate-$\frac{1}{3}$ SBC	Rate-1 Precoder	1/3
$SBC_{[4,\ 12]}$	Rate-$\frac{1}{3}$ SBC	Rate-1 Precoder	1/3
$SBC_{[5,\ 15]}$	Rate-$\frac{1}{3}$ SBC	Rate-1 Precoder	1/3
SBC Rate-1	Rate-1 SBC	Rate-$\frac{1}{3}$ RSC	1/3

(a) Code rates for Different Error Protection schemes

System Parameters	**Value**	**System Parameters**	**Value**
Source Coding	H.264/AVC	Video Bit Rate (Kbps)	64
Source Codec	JM 12.1	Video Frame Rate (fps)	15
Video Test Sequence	*Akiyo*	Channel Coded Rate (Kbps)	192
No of Frames	45	Baud-rate (Kbps)	96
Frame Format	QCIF	Channel Coding	RSC
No of 'P' frames between	44	Over-all Code Rate	1/3
two 'I' frames		Code Memory	4
Time Period of 'I' frame (sec)	3	Generator Polynomials	
Use of 1/4-Pixel Motion Estimation	Yes	RSC 1/3 (G_1, G_2, G_3)	$(11, 13, 15)_8$
Intra-frame MB update/frame	3		
per QCIF frame		Modulation Scheme	QPSK
Use of 'B' Pictures	No	Number of Transmitters, N_t	1
Use of FMO	No	Number of Receivers, N_r	1
No of Frames Used for	1-Frame	Channel	Correlated Rayleigh Fading
Inter-Frame Motion Search		Normalised Doppler Frequency	0.01
No of Video Slices/frame	9	Interleaver Length	$\approx (64000/15)$
No of Video MBs/Slice	11	No System Iterations I_t	10

(b) Systems parameters used in the schematic of Figure 4.26

Table 4.11: Code rates and systems parameters used in the schematic of Figure 4.26

bit-by-bit Monte-Carlo simulation of the iterative soft-bit source and channel decoding algorithm. It may be analysed from the EXIT trajectories of Figures 4.27, 4.28, 4.29, and 4.30 that as expected, the convergence behaviour of the SBCs improves upon increasing $d_{H,min}$.

4.6.4 Performance Results

In this section we present our performance results characterising the proposed system. For the performance analysis of our system we used the same video test sequence presented in Section 4.6.4. Furthermore, due to the limited residual redundancy inherent in the source encoded bit-stream and for the sake of reducing the computational complexity imposed, we limited the number of iterations between the RSC and USSPA decoders to $I_t = 5$, when using a rate-1 SBC $-$ i.e. no SBC. By contrast we used $I_t = 10$ iterations, when applying

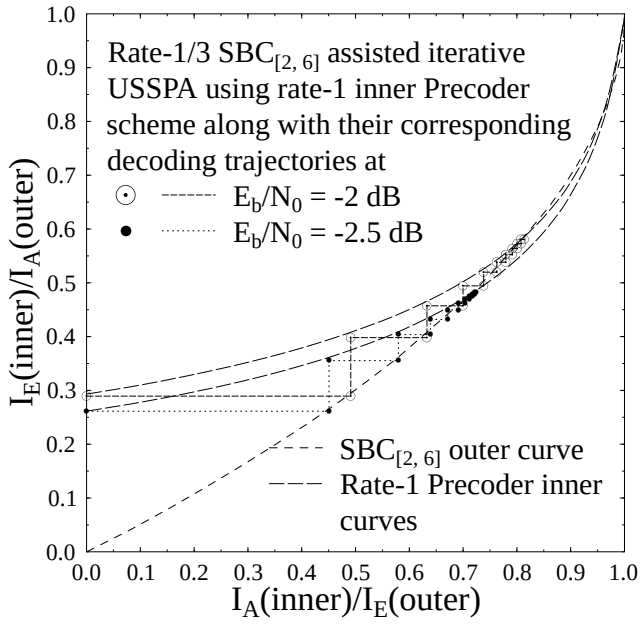

Figure 4.27: The EXIT chart and decoding trajectories of the $SBC_{[2,\ 6]}$ scheme of Figure 4.26 using the parameters of Table 4.11.

SBCs having a rate below unity. For the sake of increasing the confidence in our results, we repeated each 45-frame experiment 160 times and averaged the generated results. Figure 4.31 presents the performance of the various rate-$\frac{1}{3}$ SBC based error protection schemes of Table 4.11(a) in terms of the attainable BER, while their comparison with the rate-1 SBC based schemes is offered in Figures 4.33. Finally, the performance trends expressed in terms of the $PSNR$ versus E_b/N_0 curves are portrayed in Figures 4.32 and 4.23. It may be observed in Figure 4.32 that the $SBC_{[5,\ 15]}$ scheme having $d_{H,min} = 6$ provides the best $PSNR$ performance among the four different SBC schemes of Table 4.11(a) across the entire E_b/N_0 region considered. Its also observed in Figure 4.23 that upon using USSPA in conjunction with the rate-1 outer SBC and rate-$\frac{1}{3}$ inner RSC results in a worse $PSNR$ performance than the employment of rate-$\frac{1}{3}$ outer SBCs combined with the rate-1 inner precoder, as mentioned in Table 4.11(a), although they have the same over-all code rate. Quantitatively, using the rate-$\frac{1}{3}$ SBCs of Table 4.11(a), an additional E_b/N_0 gain of upto 27 dB may be achieved over the identical-rate benchmarker scheme dispensing with SBC. The major findings recorded for the error protection schemes considered were recorded at the PSNR degradation points of $1dB$ and $2dB$ which are summarised in Table 4.12

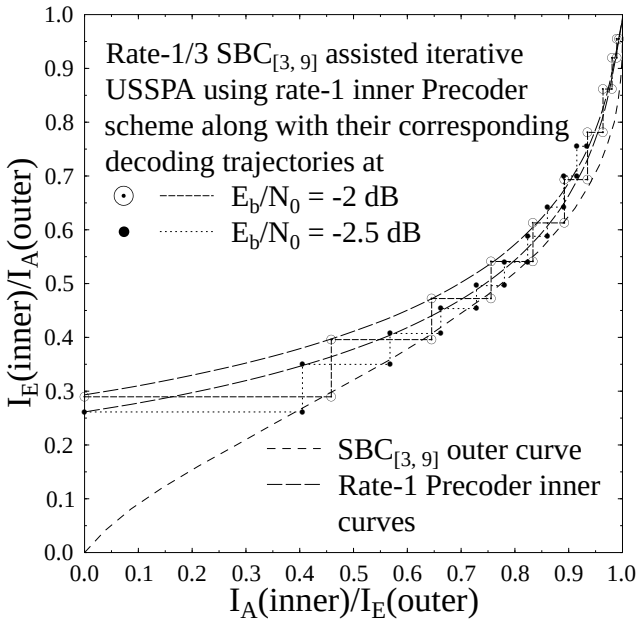

Figure 4.28: The EXIT chart and decoding trajectories of the $SBC_{[3,\ 9]}$ scheme of Figure 4.26 using the parameters of Table 4.11.

Table 4.12: Performance Summary Table for the schematic of Figure 4.26

Schematic	Parameters	Iterations	Video rate	FEC Code rate	Highest PSNR value	SBC-rate	$E_b/N_0[dB]$ at $1dB$ PSNR degradation	$E_b/N_0[dB]$ at $2dB$ PSNR degradation
Figure 4.26	Table 4.11(a)	10	64 *kbps*	0.333	41.75	$\frac{2}{6}$	1.0	0.5
						$\frac{3}{9}$	-1.5	-1.6
						$\frac{4}{12}$	-1.7	-1.8
						$\frac{5}{15}$	-2.0	-2.2
						Rate-1	26	24

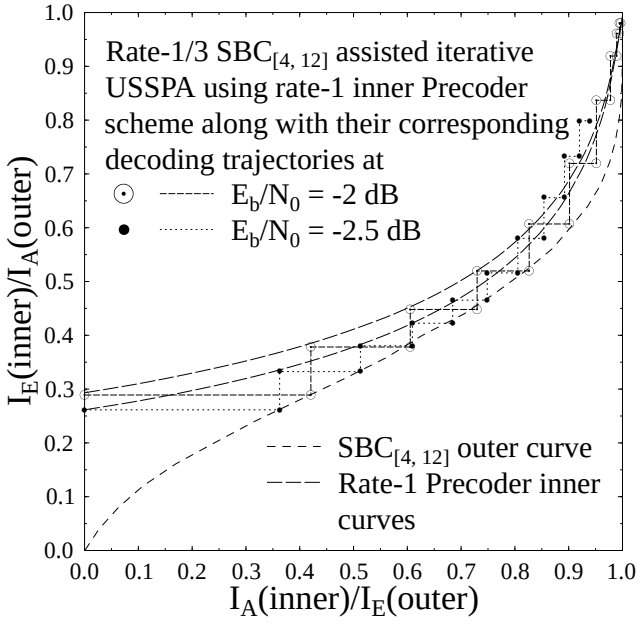

Figure 4.29: The EXIT chart and decoding trajectories of the $SBC_{[4, 12]}$ scheme of Figure 4.26 using the parameters of Table 4.11.

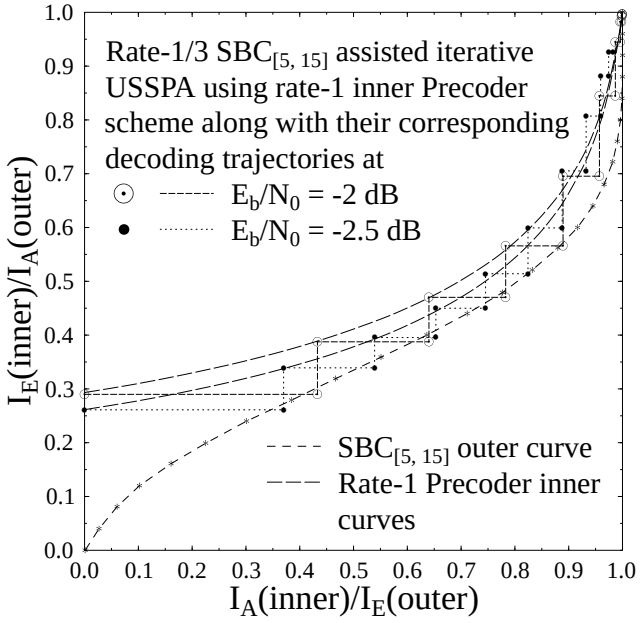

Figure 4.30: The EXIT chart and decoding trajectories of the $SBC_{[5, 15]}$ scheme of Figure 4.26 using the parameters of Table 4.11.

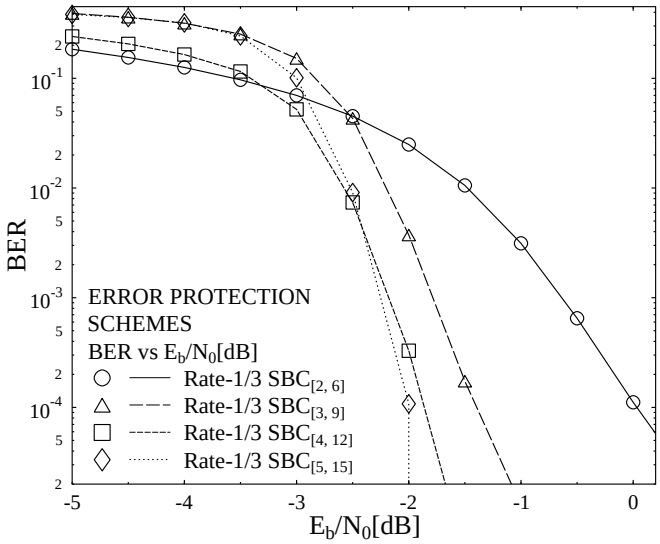

Figure 4.31: BER versus E_b/N_0 performance of the various error protection schemes of Figure 4.26 using the parameters of Table 4.11

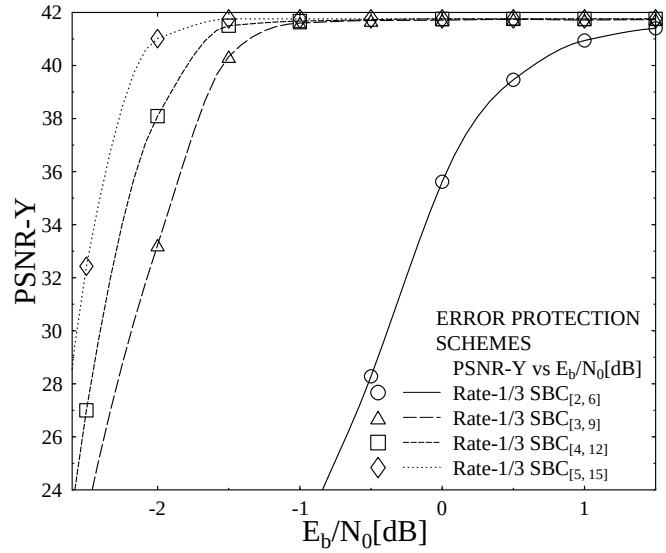

Figure 4.32: PSNR-Y versus E_b/N_0 performance of the various error protection schemes of Figure 4.26 using the parameters of Table 4.11

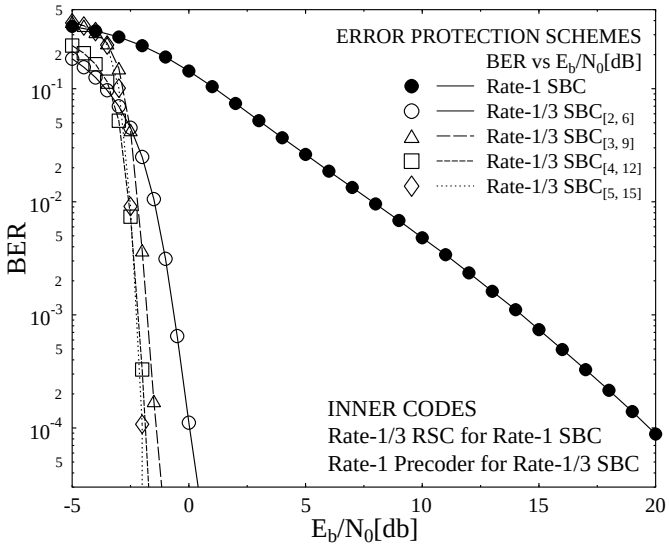

Figure 4.33: BER versus E_b/N_0 performance of the various error protection schemes of Figure 4.26 using the parameters of Table 4.11

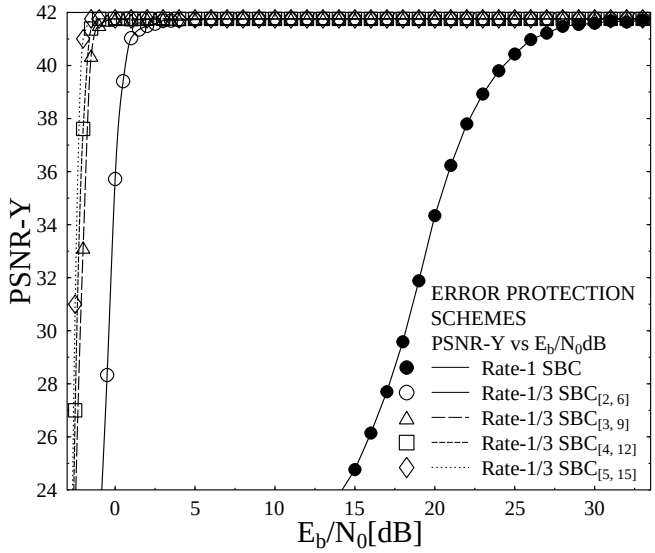

Figure 4.34: PSNR-Y versus E_b/N_0 performance of various error protection schemes of Figure 4.26 using the parameters of Table 4.11

4.7 Performance Improvement of SBCs Using Redundant Source Mapping

In this section, we will analyse the performance improvement of our proposed Mapping-I SBC coding algorithm by introducing further artificial redundancy using our proposed Redundant Source Mapping (RSM) Algorithm.

4.7.1 System Model

The schematic of our proposed videophone arrangement used as our design example for quantifying the performance of the proposed RSM schemes is shown in Figure 4.35. At the transmitter side the video sequence is compressed using the H.264 video codec and the generated video source bit-stream x_i is mapped or encoded into the bit-string x'_i by employing the RSM scheme. Afterwards, the output bit-string is interleaved using the bit-interleaver Π of Figure 4.35, yielding the interleaved sequence $\bar{x}_i$, which is then encoded by the RSC code having a specific code rate given in Table 4.15(a).

4.7.2 Redundant Source Mapping Assisted Iterative Source-Channel Decoding

The RSM was used to introduce further redundancy in the source coded streams by transforming the $Algorithm\text{-}I$ generated symbols in a systematic way which results in further increase in the $d_{H,min}$ of the generated symbols. Additionally, RSM also increases the EXIT chart flexibility of the outer code, which results in iterative convergence at further lower E_b/N_0 value relative to the corresponding SBC coding. The proposed RSM coding algorithm is described below.

4.7.2.1 RSM Coding Algorithm

In order to further decrease the SBC coding rate of $Algorithm\text{-}I$ described in Section 4.5.1 and to increase its $d_{H,min}$, we introduce RSM Coding Algorithm in which the N additional bits are concatenated to the bits encoded according to $Algorithm\text{-}I$ by repeating the same coded bits in a reverse order, which results in a K to $(2 \times N)$-bit mapping, where we have $N = (K + 1)$, as depicted in Table 4.13. Let us now demonstrate the

Table 4.13: [N+1] Different RSM combinations.

Input Symbols	*Algorithm-I* Symbols	*RSM* Symbols
$S_{(1)}$	$r_1 b_1 b_2 .. b_N$	$r_1 b_1 b_2 \cdots b_N b_N \cdots b_2 b_1 r_1$
$S_{(2)}$	$r_2 b_1 b_2 .. b_N$	$r_2 b_1 b_2 \cdots b_N b_N \cdots b_2 b_1 r_2$
$\vdots$	$\vdots$	$\vdots$
$S_{(2^K)}$	$r_{2^K} b_1 b_2 .. b_N$	$r_{2^K} b_1 b_2 \cdots b_N b_N \cdots b_2 b_1 r_{2^K}$

power of RSM with the aid of a design example. As an example, the various SBC mapping symbols along with the corresponding RSM mapping symbols generated by applying the proposed RSM_K^N encoding schemes along with their corresponding $d_{H,min}$ is summarised in Table 4.14. Again, as it becomes evident from Table 4.14,

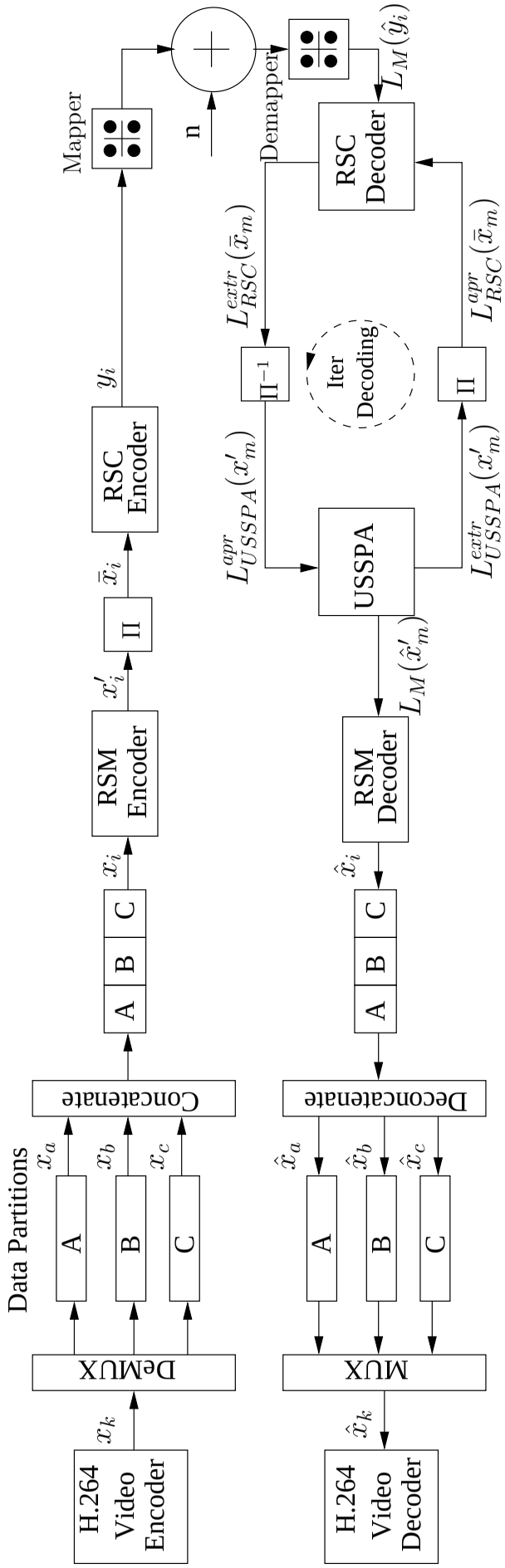

Figure 4.35: The proposed system model. All system parameters were summarized in Table 4.15. The difference in comparison to Figure 4.26 is the employment of an RSC inner code. Additionally, RSM schemes with high $d_{H,min}$ were incorporated, which resulted in the overall code rate of 1/4.

the EXIT-chart optimised RSM also ensures that the mapped symbols exhibit $d_{H,min} \geq 2$. Additionally, only 2^K out of $2^{(N\times 2)}$ possible $(N \times 2)$-*bit* symbols of $RSM\ Mapping$ are legitimate in the mapped source coded bit-stream,where $N = (K + 1)$, which exhibits a non-uniform probability of occurance for the N-*bit* mapped source symbols. Figure 4.36 portray the EXIT characteristics of the USSPA scheme of Figure 4.10 using either the rate-1 RSM$^{\star 3}$ or the rate < 1 SBC and RSM schemes shown in Table 4.14. More specifically, the EXIT curve of USSPA using rate < 1 SBC and RSM schemes does indeed reach to the top right corner of the EXIT chart at $(I_A, I_E) = (1, 1)$ and hence results in an infinitesimally low BER. By contrast, the USSPA scheme using a rate-1 RSM*, i.e. no RSM fails to do so.

Table 4.14: Different RSM schemes with corresponding symbols and $d_{H,min}$

Mapping Type	Symbols in Decimal	$[d_{H,min}]$
Rate1 RSM	{0,1}	1
Rate-$\frac{2}{3}$ SBC$_2^3$	{0,3,5,6}	2
Rate-$\frac{3}{4}$ SBC$_3^4$	{0,3,5,6,9,10,12,15}	2
Rate-$\frac{4}{5}$ SBC$_4^5$	{0,3,5,6,9,10,12,15,17,18,20,23,24,27,29,30}	2
Rate-$\frac{5}{6}$ SBC$_5^6$	{0,3,5,6,9,10,12,15,17,18,20,23,24,27,29,30,33, 34,36,39,40,43,45,46,48,51,53,54,57,58,60,63}	2
Rate-$\frac{1}{3}$ RSM$_2^6$	{0,30,45,51}	4
Rate-$\frac{3}{8}$ RSM$_3^8$	{0,60,90,102,153,165,195,255}	4
Rate-$\frac{2}{5}$ RSM$_4^{10}$	{0,120,180,204,306,330,390,510,561,585,645, 765,771,891,951,975}	4
Rate-$\frac{5}{12}$ RSM$_5^{12}$	{0,240,360,408,612,660,780,1020,1122,1170, 1290,1530,1542,1782,1902,1950,2145,2193, 2313,2553,2565,2805,2925,2973,3075,3315,} 3435,3483,3687,3735,3855,4095}	4

The coding parameters of the different SBC and RSM schemes used in our design example are shown in Table 4.15(a). We considered a concatenated rate $R = \frac{1}{4}$ RSC encoder having a code memory of 4 and octally represented generator polynomials of $(G_1, G_2, G_3, G_4) = (13, 15, 15, 17)_8$. Observe from the Table 4.15(a) that an overall code-rate of $R = \frac{1}{4}$ was maintained by adjusting the puncturing rate of the concatenated RSC in order to accommodate the different RSM rates of Table 4.14, while keeping the overall bit-rate budget constant.

4.7.3 System Performance Results

In this section, we present the performance results of our proposed system. For the performance evaluation of our proposed RSM, we used the "Akiyo" test video sequence described in the Section 4.5.6 as our video test sequence. Moreover, due to the limited residual redundancy inherent in the source encoded bit-stream and for the sake of reducing the computational complexity imposed, we limited the number of iterations between the

[3]For the sake of using a unified terminology, we refer to the scheme using no RSM as the rate-1 RSM*.

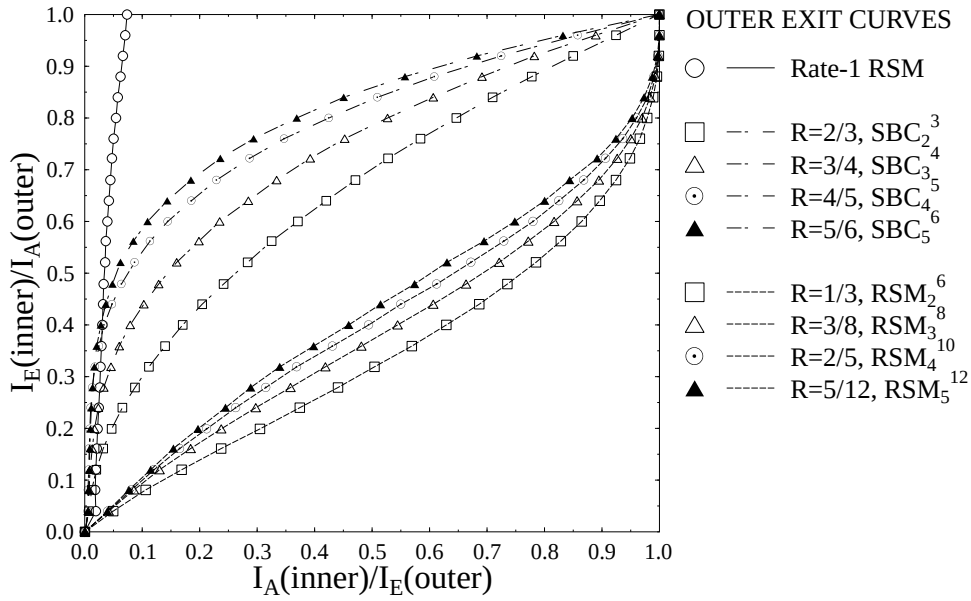

Figure 4.36: EXIT characteristics of the USSPA scheme of Figure 4.35 with the aid of various-rate RSMs, when using the system parameters of Table 4.15.

RSC and USSPA decoders to $I_t = 5$, when using a rate-1 RSM − i.e. no RSM. By contrast we used $I_t = 10$ iterations, when applying RSM schemes having a rate below unity. For the sake of increasing the confidence in our results, we repeated each 45-frame experiment 160 times and averaged the generated results. Addtitionally, the performance of our proposed system was evaluated by keeping the same overall code rate as well as video rate for the different considered error protection schemes.

The actual decoding trajectories of the various error protection schemes employing the different SBC *Algorithm-I* schemes along with their corresponding *RSM Mapping* schemes as well as using the respective constituent inner RSCs detailed in Table 4.15(a) was recorded at $E_b/N_0 = -0\ dB$, $-1\ dB$ and $E_b/N_0 = -3.0\ dB$, $-3.5\ dB$ respectively, as portrayed in Figures 4.37, 4.38, 4.39, and 4.40. These trajectories were recorded by acquiring the mutual information at the input and output of both the inner and outer decoder during the bit-by-bit Monte-Carlo simulation of the iterative soft-bit source and channel decoding algorithm. It may be inferred from the EXIT trajectories of Figures 4.37, 4.38, 4.39, and 4.40 that as expected, the convergence behaviour of the SBC *Algorithm-I* coding improves upon the RSM coding, with additional redundancy and improved $d_{H,min}$.

Figure 4.41 presents the performance of the various rate RSM based error protection schemes of Table 4.14 in terms of the attainable BER along with the AWGN optimum performance curve, while their comparison with the rate-1 RSM* based schemes is offered in Figures 4.43. The performance trends expressed in terms of the $PSNR$ versus E_b/N_0 curves are portrayed in Figures 4.42 and 4.44 along with the reference AWGN performance curves. It may be observed in Figure 4.42 that the RSM_2^6 scheme with lowest coding rate among the different considered RSM schemes of Table 4.15(a) provides the best $PSNR$ performance across the entire

Error Protection Scheme	Code Rate		
	RSC	RSM	Overall
rate-1 RSM*	1/4	1	1/4
SBC_2^3	3/8	2/3	1/4
SBC_3^4	1/3	3/4	1/4
SBC_4^5	5/16	4/5	1/4
SBC_5^6	3/10	5/6	1/4
RSM_2^6	3/4	2/6	1/4
RSM_3^8	2/3	3/8	1/4
RSM_4^{10}	5/8	4/10	1/4
RSM_5^{12}	3/5	5/12	1/4

(a) Code rates for Different Error Protection schemes

System Parameters	**Value**	**System Parameters**	**Value**
Source Coding	H.264/AVC	Video Bit Rate (Kbps)	64
Source Codec	JM 12.1	Video Frame Rate (fps)	15
Video Test Sequence	*Akiyo*	Channel Coded Rate (Kbps)	256
No of Frames	45	Baud-rate (Kbps)	128
Frame Format	QCIF	Channel Coding	RSC
No of 'P' frames between	44	Over-all Code Rate	1/4
two 'I' frames		Code Memory	4
Time Period of 'I' frame (sec)	3	Generator Polynomials	
Use of 1/4-Pixel Motion Estimation	Yes	RSC 1/4 (G_1, G_2, G_3, G_4)	$(11, 15, 15, 17)_8$
Intra-frame MB update/frame	3		
per QCIF frame		Modulation Scheme	QPSK
Use of 'B' Pictures	No	Number of Transmitters, N_t	1
Use of FMO	No	Number of Receivers, N_r	1
No of Frames Used for	1-Frame	Channel	Correlated Rayleigh Fading
Inter-Frame Motion Search		Normalised Doppler Frequency	0.01
No of Video Slices/frame	9	Interleaver Length	$\approx (64000/15)$
No of Video MBs/Slice	11	No System Iterations I_t	10

(b) Systems parameters used in the schematic of Figure 4.35

Table 4.15: Code rates and systems parameters used in the schematic of Figure 4.35

E_b/N_0 region considered. Its also observed in Figure 4.44 that when using USSPA in conjunction with the rate-1 outer RSM* and rate-$\frac{1}{4}$ inner RSC results in a worse $PSNR$ performance than the RSM schemes having a rate below unity combined with their respective inner RSCs, at the same over-all code rate of $\frac{1}{4}$, as mentioned in Table 4.15(a). Quantitatively, using the RSM of Table 4.15(a) having a rate lower than 1, an additional E_b/N_0 gain of upto 20 dB may be achieved over the rate-1 RSM*.

Finally, the subjective video quality achieved by the proposed error protection schemes consisting of SBC

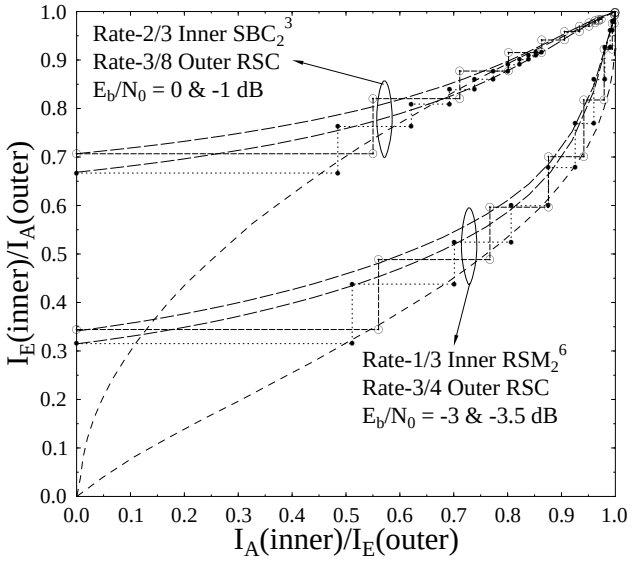

Figure 4.37: The EXIT chart and simulated decoding trajectories using the SBC_2^3 and RSM_2^6 schemes of Figure 4.35 as well as the parameters of Table 4.15.

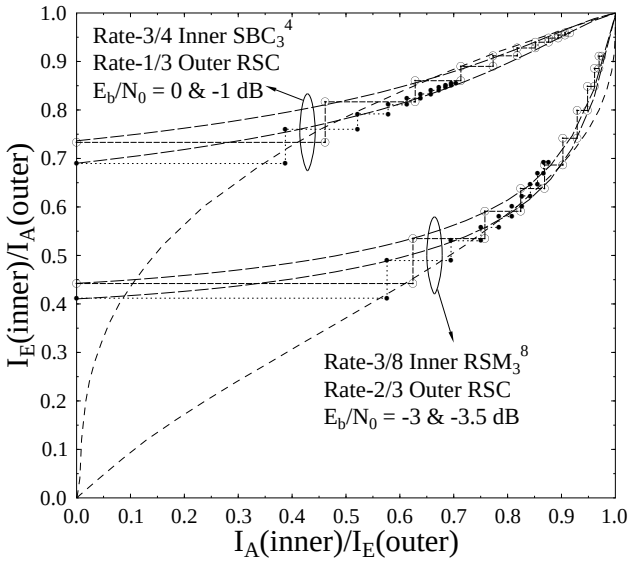

Figure 4.38: The EXIT chart and simulated decoding trajectories of the SBC_3^4 and RSM_3^8 schemes of Figure 4.35 using the parameters of Table 4.15.

Algorithm-I coding was recorded in Figure 4.46 at the channel E_b/N_0 value of 0.5 dB. The corresponding results achieved by *Mapping-II* RSM coding at E_b/N_0 value of -3.0 dB were presented in Figure 4.46. In order to have a fair subjective video quality comparison, we present both the average and cumulative-error results of both the luminance and chrominance components of the 30 *"Akiyo"* video test sequences described

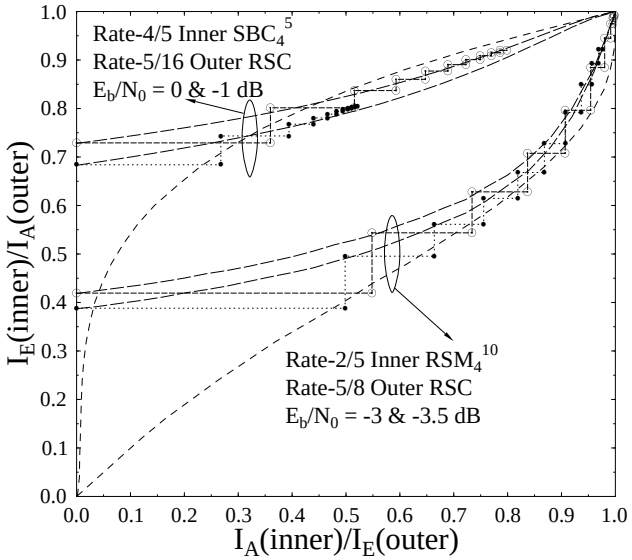

Figure 4.39: The EXIT chart and simulated decoding trajectories of the SBC_4^5 and RSM_4^{10} schemes of Figure 4.35 using the parameters of Table 4.15.

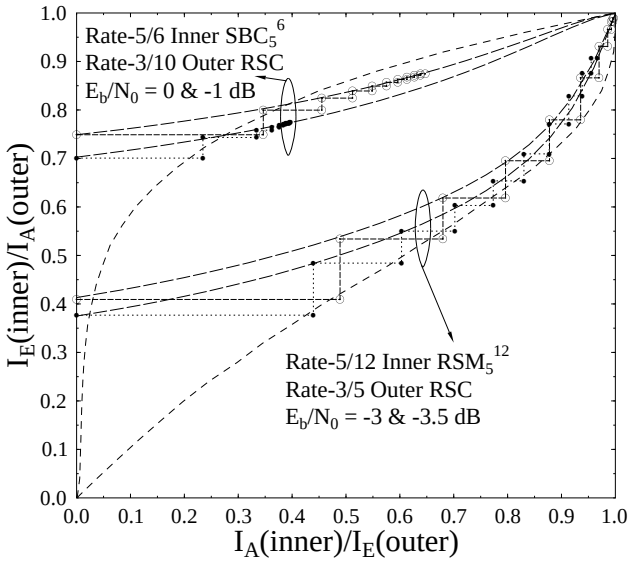

Figure 4.40: The EXIT chart and simulated decoding trajectories of the SBC_5^6 and RSM_5^{12} schemes of Figure 4.35 using the parameters of Table 4.15.

in Section 4.7.3, decoded using the H.264 video codec after transmission through our proposed system for each type of setup. Observe from Figure 4.46 that the achievable video quality improves upon decreasing the SBC *Algorithm-I* code rate. Additionally, its clear from Figure 4.46 that the employment of the *RSM Mapping* scheme provides an improved video quality at 3.5 dB lower E_b/N_0 value relative to the results of various

Mapping-I RSM schemes presented in Figure 4.46. The major findings of the consider error protection schemes at PSNR degradation points of $1dB$ and $2dB$ are summarised in Table 4.16

Table 4.16: Performance Summary Table for the schematic of Figure 4.35

Schematic	Parameters	Iterations	Video rate	FEC Code rate	Highest PSNR value	RSM-rate	$E_b/N_0[dB]$ at $1dB$ PSNR degradation	$E_b/N_0[dB]$ at $2dB$ PSNR degradation
Figure 4.35	Table 4.15(a)	10	64 *kbps*	0.25	41.75	$\frac{2}{3}$	3.0	1.5
						$\frac{3}{4}$	3.5	2.5
						$\frac{4}{5}$	3.7	3.0
						$\frac{5}{6}$	4.5	3.2
						$\frac{2}{6}$	-3.1	-3.2
						$\frac{3}{8}$	-2.2	-2.2
						$\frac{4}{10}$	-2.3	-2.8
						$\frac{5}{12}$	-2.7	-2.9
						Rate-1	23.0	21.0

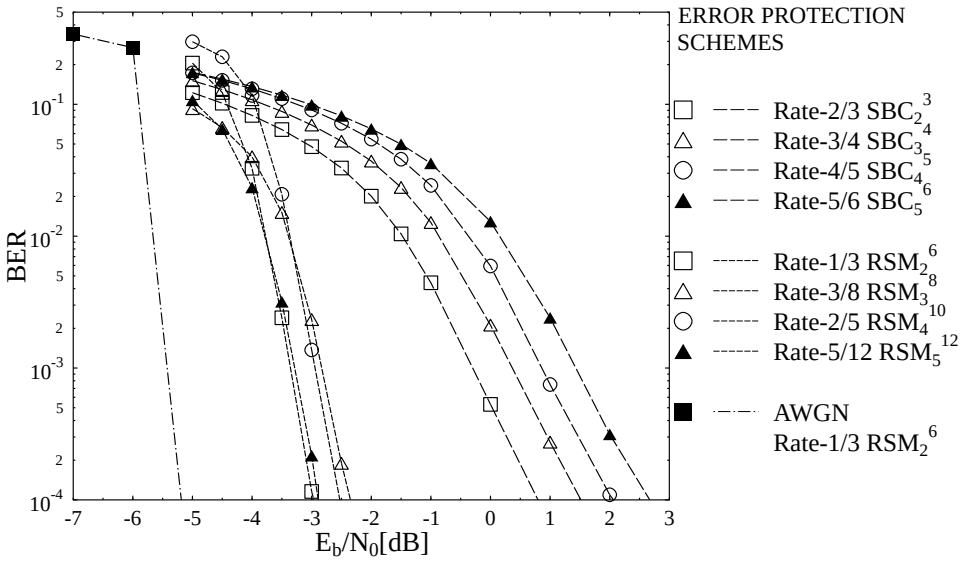

Figure 4.41: BER versus E_b/N_0 performance of the various error protection schemes of Figure 4.35 using the parameters of Table 4.15

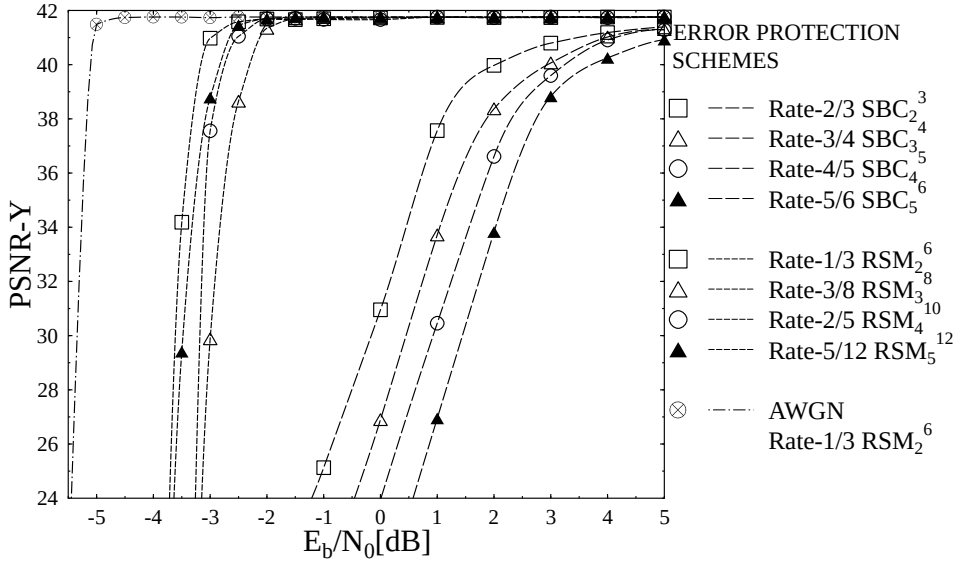

Figure 4.42: PSNR-Y versus E_b/N_0 performance of the various error protection schemes of Figure 4.35 using the parameters of Table 4.15

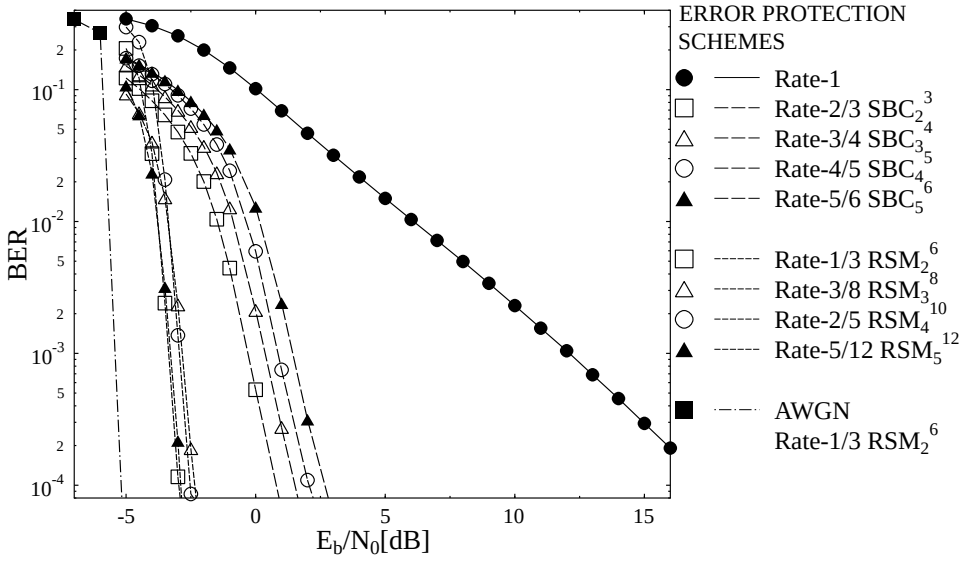

Figure 4.43: BER versus E_b/N_0 performance of the various error protection schemes of Figure 4.35 using the parameters of Table 4.15

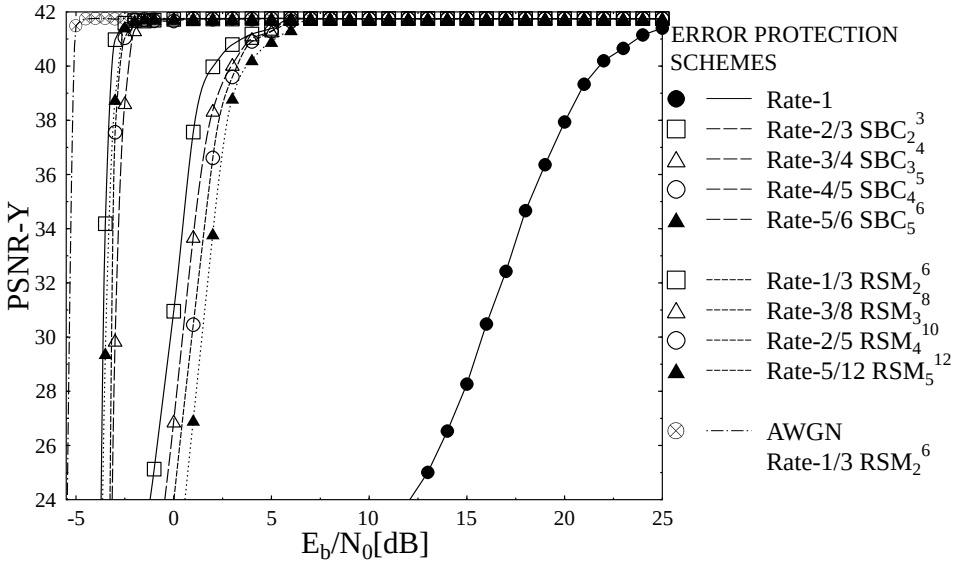

Figure 4.44: PSNR-Y versus E_b/N_0 performance of various error protection schemes of Figure 4.35 using the parameters of Table 4.15

Figure 4.45: Subjective video quality of of Figure 4.35 using the 45^{th} *"Akiyo"* video sequence frame in terms of (from top) average and cumulative-error video quality using (from left) Rate-$\frac{2}{3}$, $\frac{3}{4}$, $\frac{4}{5}$ and $\frac{5}{6}$ SBC *Algorithm-I* using the parameters of Table 4.15 at E_b/N_0=0.5 dB.

Figure 4.46: Subjective video quality of the 45^{th} *"Akiyo"* video sequence frame in terms of (from top) average and cumulative-error video quality using (from left) Rate-$\frac{2}{6}$, $\frac{3}{8}$, $\frac{4}{10}$ and $\frac{5}{12}$ *RSM Mapping* summarised in Table 4.15(a) at E_b/N_0=-3.0 dB.

4.8 Chapter Conclusions

This chapter has the following findings,

- In Section 4.3 it was observed that in ISCD both the source and channel decoder exploit the residual redundancy for error correction and to determine, where the errors occurred and how to correct them.

- In Section 4.4 we analysed the beneficial effect of soft-bit source decoding on the achievable performance of the iterative video decoding scheme of Figure 4.6. The source video sequence was encoded using the sophisticated state-of-the-art H.264 video codec operated at a low target bit-rate of 64 $kbps$. Explicitly, an E_b/N_0 gain of 5 dB is attained in Figure 4.9 using the UEP1 scheme of Table 4.4(a) with reference to UEP2, while an E_b/N_0 gain of 3 dB is achieved by the latter with reference to the EEP scheme, at the PSNR degradation point of 2 dB.

- It is observed from the results of Figure 3.26 and 4.9 that the USSPA aided iterative decoding scheme of Figure 4.4 using RSC codes resulted in rather modest performance improvements relative to the identical type of setup dispensing with iterative USSPA, as described in Section 3.10.1. The reason for this modest performance improvement was the lack of an adequate amount of residual redundancy in the bit-stream encoded using the sophisticated state-of-the-art H.264 video codec operated at a low target bit-rate of 64 $kbps$, for example.

- In the Section 4.5 we elaborated on the benefits of artifical redundancy imposed on the coded video stream in order to improve the achievable iterative decoding performance. We proposed a family of SBCs in Section 4.5.1 for achieving guaranteed convergence in soft-bit assisted iterative JSCD, which facilitates improved iterative USSPA operations. The DP H.264 source coded video was used to evaluate the attainable performance of our system using the SBC-assisted iterative USSPA scheme of Figure 4.5 in conjunction with RSC codes for transmission over correlated narrowband Rayleigh fading channels. The effect of different SBC schemes having diverse minimum Hamming distances ($d_{H,min}$) and code rates on the attainable system performance was characterised, when using iterative USSPA and channel decoding, while keeping the overall bit-rate budget constant by appropriately partitioning the total available bit rate budget between the source and channel codecs. Naturally, our goal was to improve the overall BER performance and to enhance the objective video quality expressed in terms of PSNR.

- In Section 4.5.5 EXIT charts were used for analysing the attainable system performance, where we observed from the decoding trajectories of Figures 4.12, 4.13, 4.14 and 4.15 that the convergence behaviour of the SBCs considered degrades upon increasing their coding rate. Explicitly, our experimental results of Figure 4.21 show that the proposed error protection scheme using rate-$\frac{1}{3}$ SBCs having $d_{H,min} = 6$ outperforms the identical-rate SBCs having $d_{H,min} = 3$ by about 2.5 dB at the $PSNR$ degradation point of 1 dB. Additionally, an E_b/N_0 gain of 8 dB was achieved in Figure 4.21 compared to the rate-$\frac{5}{6}$ SBC having $d_{H,min} = 2$ and an identical overall code-rate. Furthermore, an E_b/N_0 gain of 25 dB was attained in Figure 4.23 at the PSNR degradation point of 1 dB, while using iterative soft-bit source and channel decoding with the aid of rate-$\frac{1}{3}$ SBCs relative to the identical-rate benchmarker.

- Furthermore, in Section 4.6 we analysed the performance of the proposed SBC algorithms using a Rate-1 precoder as our inner code. In Section 4.6.3 EXIT charts were used to record the actual decoding trajectories of the various error protection schemes employing the different rate-$\frac{1}{3}$ outer SBCs as well as using the rate-1 inner precoder detailed in Table 4.11(a). As expected, the convergence behaviour of the SBCs improves upon increasing $d_{H,min}$.

- Explicitly, the H.264/SBC/rate-1 precoder design example exhibited an E_b/N_0 gain of 3 dB in Figure 4.32 at the $PSNR$ degradation point of 1 dB, when using SBCs having $d_{H,min} = 6$ compared to the identical-rate SBCs having $d_{H,min} = 3$. Moreover, an E_b/N_0 gain of 27 dB was attained in Figure 4.34, when using iterative soft-bit source and channel decoding with the aid of rate-$\frac{1}{3}$ SBCs relative to the identical-rate benchmarker.

- Moreover, in Section 4.7 further redundancy was imposed on the source coded streams by transforming the SBC algorithms of Section 4.5 in a systematic way, which resulted in a further increase in the $d_{H,min}$ of the generated symbols. From the EXIT curves obtained in Section 4.7.3 portrayed in Figures 4.37, 4.38, 4.39, and 4.40 it may be observed that the convergence behaviour of the SBC coding improves upon incorporating RSM associated with additional redundancy and an improved $d_{H,min}$.

- Our design of Figure 4.35 was detailed in Section 4.7 which was based on the H.264, RSM and RSC constituent components. The design exhibited an E_b/N_0 gain of 5 dB in Figure 4.42 at the $PSNR$ degradation point of 2 dB, when using the RSM_2^6 scheme associated with $d_{H,min} = 4$ compared to the employment of the RSM_2^3 having $d_{H,min} = 2$, which in turn outperformed the RSM_5^6 having both an identical $d_{H,min}$ and the same overall system code-rate by about 2 dB at the $PSNR$ degradation point of 2 dB. Moreover, an E_b/N_0 gain of 20 dB was attained in Figure 4.44 using iterative soft-bit source and channel decoding with the aid of the RSM_2^3 relative to the identical-rate benchmarker dispensing with RSM.

4.9 Chapter Summary

We commenced with an overview of both the transmitter and of the conventional receiver in Section 4.2. The details of the receiver using ISCD were provided in Section 4.3, which highlighted the concept of soft-bit source decoding by comparing it to SISO channel decoding along with the concept of extrinsic information generation. In Section 4.4 a design example was provided in order to quantify the iterative source-channel decoder's performance improvements. Additionally, the concept of SBCs was introduced in Section 4.5, which is applicable to wide-ranging multimedia services such speech, audio and video. It was observed from the EXIT-chart analysis of Figures 4.12, 4.13, 4.14 and 4.15 in Section 4.5.5, that the convergence behaviour of ISCD is substantially improved with the aid of SBCs. Explicitly, our experimental results of Figure 4.21 show that the proposed error protection scheme using rate-$\frac{1}{3}$ SBCs having $d_{H,min} = 6$ outperforms the identical-rate SBCs having $d_{H,min} = 3$ by about 2.5 dB at the $PSNR$ degradation point of 1 dB. Additionally, an E_b/N_0 gain of 8 dB was achieved in Figure 4.21 compared to the rate-$\frac{5}{6}$ SBC having $d_{H,min} = 2$ and an identical overall code-rate. Furthermore, an E_b/N_0 gain of 25 dB was attained in Figure 4.23 at the PSNR degradation point

of 1 dB, while using iterative soft-bit source and channel decoding with the aid of rate-$\frac{1}{3}$ SBCs relative to the identical-rate benchmarker.

The beneficial effect of SBCs on the achievable performance improvement of rate-1 inner codes was quantified in Section 4.6. From the EXIT analysis of Figures 4.27, 4.28, 4.29 and 4.30 of Section 4.6.3 we concluded that the performance of the iterative source-channel codec improved as a result of increasing the minimum Hamming distance $d_{H,min}$ of the SBCs employed. Explicitly, the design example of Section 4.6 exhibited an E_b/N_0 gain of 3 dB in Figure 4.32 at the $PSNR$ degradation point of 1 dB, when using SBCs having $d_{H,min} = 6$ compared to the identical-rate SBCs having $d_{H,min} = 3$. Moreover, an E_b/N_0 gain of 27 dB was attained in Figure 4.34, when using iterative soft-bit source and channel decoding with the aid of rate-$\frac{1}{3}$ SBCs relative to the identical-rate benchmarker.

Additionally, the concept of RSMs was described in Section 4.7 along with their EXIT-chart analysis and the associated performance results. From the EXIT curves portrayed in Figure 4.37, 4.38, 4.39 and 4.40 of Section 4.7.3 it was observed that the convergence behaviour of SBC schemes is enhanced upon using RSMs, associated with additional redundancy and improved $d_{H,min}$. Explicitly, the proposed system based on RSM exhibited an E_b/N_0 gain of 5 dB in Figure 4.42 at the $PSNR$ degradation point of 2 dB, when using the RSM_2^6 scheme associated with $d_{H,min} = 4$ compared to the employment of the RSM_2^3 having $d_{H,min} = 2$. The latter outperformed the RSM_5^6 having an identical $d_{H,min}$ and overall system code-rate by about 2 dB at the $PSNR$ degradation point of 2 dB. Moreover, an E_b/N_0 gain of 20 dB was attained in Figure 4.44 using iterative soft-bit source and channel decoding with the aid of the RSM_2^3 relative to the identical-rate benchmarker dispensing with RSM.

Chapter 5

Near Capacity Video Transmission System Design

5.1 Introduction

For wireless multimedia applications maintaining both a high compression efficiency as well as a high integrity, for transmission over heterogeneous communications networks is of primary importance. As discussed in Chapter 2, the state-of-the-art H.264 codec is designed with a primary focus on achieving compression efficiency and providing interoperability within diverse communication networks. Similar to the Chapter 4, in this chapter we also opted for using the high-compression and network-friendly H.264/AVC source codec [76], which employs diverse variable-length coding and predictive coding techniques. However, these coding techniques make the compressed bit-stream more vulnerable to the effects of channel errors. Owing to variable length coding a single bit error in the video stream may render the correct decoding of future code words impossible. Moreover, due to predictive coding the effects of channel errors are likely to be propagated to the neighbouring video blocks. Therefore, as discussed in Chapter 2, various error resilient techniques, such as Data-Partitioning (DP) described in Section 3.8 have been incorporated into the H.264/AVC codec in order to mitigate this error sensitivity problem [189–193].

To elaborate a little further, a conventional transmitter and receiver benchmark-architecture was presented in Section 4.2, followed by the Iterative Source-Channel Decoding (ISCD) concept in Section 4.3.2. Additionally, in Chapter 4 we proposed various error resilient techniques in order to reduce the Bit Error Ratio (BER) and to improve the subjective video quality at the receiver. More specifically, in Section 4.4 we considered unequal error protection using soft-bit source decoding assisted Recursive Systematic Convolutional (RSC) codes described in Section 3.4.1. The error-protected video-stream was QPSK modulated and transmitted over a temporally correlated narrowband Rayleigh fading channel. Additionally, ISCD0-aided performance improvements were achieved in Section 4.5 by introducing redundancy in the source coded bit-stream using a technique, which we referred to as SBC schemes. Furthermore, in Section 4.7 the performance of iterative detection was improved by Redundant Source Mapping (RSM), which increases the flexibility of the outer code in shaping

the EXIT chart and results in iterative decoding convergence at further reduced E_b/N_0 values. By contrast, in this chapter we will further improve the system investigated in Chapter 4 by providing more sophisticated coding and transmission strategies in order approach capacity.

Various error resilient schemes have been proposed in [1], but the price paid for the improved robustness is the reduced compression efficiency and increased computational complexity. A serial concatenated code based iterative joint source-channel decoding procedure was proposed in [198]. Moreover, an IrRegular Variable Length Coding (IrVLC) scheme designed for near-capacity joint source and channel coding was presented in [173]. Likewise, [202] advocates the employment of state-of-the-art system design principles and the performance of burst-by-burst adaptive transceivers designed for interactive cellular and cordless video telephony. Instead of the traditional serial concatenation of the classic Variable Length Codes (VLC) with a channel code, a parallel concatenated coding scheme was presented in [200], where the VLCs were combined with a turbo code. The performance characteristics of different video transceivers assisted by decision feedback equaliser-aided wide-band burst-by-burst adaptive trellis-coded modulation, Turbo Trellis-Coded Modulation (TTCM) and bit-interleaved-coded modulation designed for H.263 assisted video telephony were presented in [122]. A joint video and channel coding system employing an iteratively decoded serial concatenation of a Vector Quantisation (VQ) based video codec and a Trellis-Coded Modulation (TCM) scheme was proposed in [173]. The attainable performance improvements of ISCD schemes employing a specific bit-to-symbol mapping scheme were analysed in [4], in the context of H.264 encoded video transmission using UEP and Sphere Packing (SP) modulation aided Differential Space Time Spreading (DSTS) in a non-coherently detected, diversity-aided multiuser scenario. Furthermore, the performance analysis of diverse low-complexity Short Block Codes (SBCs) conducted for the transmission of an H.264 coded video stream over correlated narrowband Rayleigh fading channels was presented in [2]. On the other hand an effective method of compensating the effects of wireless channels was provided in [229,230] using space time coding by providing a diversity gain. Similarly, Hochwald *et al*. [231] proposed the transmit diversity concept known as Space-Time Spreading (STS), which uses coherent detection relying on channel estimation at the receiver. Additionally, the major coherent spatial diversity techniques are summarised in Tables 5.1 and 5.2.

Against this background, in this chapter we propose near-capacity SP modulation aided DSTS design for the transmission of the video coded stream. SP modulation is a specific scheme, which maintains the highest possible Euclidian distance of the modulated symbols, as detailed in [232]. DSTS is a low-complexity technique that does not require channel estimation, because it relies on non-coherent detection. This low-complexity detection is particularly important in the context of Multiple-Input Multiple-Output (MIMO) systems using N_T transmit and N_R receive antennas, which would require the estimation of (N_TxN_R) MIMO channels, hence substantially increasing both the cost and complexity of the receiver. Furthermore, the pilot-overhead required by the MIMO channel estimator may also be excessive. Additionally, different system architectures such as UEP using RSC codes, Unequal Source-Symbol Probability Aided (USSPA) assisted RSC coded video schemes and SBC aided RSC coded video arrangements were utilised by the SP modulation aided DSTS transmission system. Furthermore, this chapter is concluded with a near-capacity three-stage system designed for iterative detection aided H.264 wireless video telephony.

The novelty and rationale of this chapter can be summarised as follows:

Year	Author(s) name and contribution
1959	*Author(s):* Brennan [233] *Contribution:* introduced and analysed selection combining, maximum ratio combining and equal gain combining.
1991	*Author(s):* Wittneben [234] *Contribution:* proposed a bandwidth-efficient transmit diversity technique, where different base stations transmit the same signal.
1993	*Author(s):* Wittneben [235] *Contribution:* proposed a diversity scheme for multiple transmit antennas. In contrast to other proposals, no bandwidth expansion is required and the signals at the different antennas carry the same digital information, but have different modulation parameters (modulation diversity).
	Author(s): Seshadri *et al*. [236] *Contribution:* proposed a transmit diversity scheme that was inspired by the delay diversity design of Wittneben [235].
1994	*Author(s):* Winters [237] *Contribution:* proved that the diversity advantage of the scheme proposed in [234] is equal to the number of transmit antennas.
1996	*Author(s):* Eng *et al*. [238] *Contribution:* Compared several diversity combining techniques designed for Rayleigh fading channels using coherent detection and proposed a new second-order selection combining technique.
1998	*Author(s):* Alamouti [229] *Contribution:* discovered a transmit diversity scheme using two transmit antennas, which can be detected by the independent linear processing of the two signals at the receiver.
	Author(s): Tarokh *et al*. [230] *Contribution:* proposed new design criteria for achieving both the maximum attainable diversity and coding gains in addition to the design of space-time trellis codes.
1999	*Author(s):* Tarokh *et al*. [239, 240] *Contribution:* generalised Alamouti's diversity scheme [229] to more than two transmit antennas.
	Author(s): Guey [241] *Contribution:* derived the criterion for designing the maximum transmit diversity gain.

Table 5.1: Major coherent spatial diversity techniques (Part 1).

Year	Author(s) name and contribution
2001	*Author(s):* Hochwald *et al*. [231] *Contribution:* proposed the twin-antenna-aided space-time spreading scheme.
	Author(s): Jafarkhani *et al*. [242] *Contribution:* designed rate-one STBC codes which are quasi-orthogonal and provide partial diversity gain.
2002	*Author(s):* Hassibi *et al*. [243] *Contribution:* proposed the LDCs that provide a flexible trade-off between space-time coding and spatial multiplexing.
	Author(s): Stoica *et al*. [244] *Contribution:* compared the performance of STBCs when employing different estimation/detection techniques and proposed a blind detection scheme dispensing with the pilot symbols transmission for channel estimation.
2003	Wang *et al*. [245] derived upper bounds for the rates of complex orthogonal STBCs.
	Su *et al*. [246] introduced the concept of combining orthogonal STBC designs with the principle of sphere packing.
2005	Zhang *et al*. [247] derived the capacity and probability of error expressions for PSK/PAM/QAM modulation combined with STBCs for transmission over Rayleigh-, Ricean- and Nakagami-fading channels.
2006	Liew *et al*. [248] studied the performance of STTC and STBC in the context of wideband channels using adaptive orthogonal frequency division multiplex modulation.
2007	Alamri *et al*. [172] modified the SP demapper of [246] for the sake of accepting the *a priori* information passed to it from the channel decoder as extrinsic information.
2008	Luo *et al*. [249] combined orthogonal STBCs with delay diversity and designed special symbol mappings for maximising the coding advantage.

Table 5.2: Major coherent spatial diversity techniques (Part 2).

1. *We performed UEP based optimised H.264 coded video transmissions by assigning different-rate RSC codes to the three different types of H.264 bit-stream partitions detailed in Section 5.2. Instead of the classic single-input and single-output transmission scheme, we used a sophisticated, yet low-complexity SP modulation aided MIMO scheme employing DSTS for attaining a diversity gain without the need for high-complexity MIMO channel estimation.*

2. *The powerful, yet low-complexity SBCs of Section 4.5 were utilised, which impose carefully designed intentional redundancy on the source coded bit-stream and hence assist us in ensuring that the outer decoder's EXIT curve reaches the* $(1, 1)$ *point of perfect convergence in the EXIT chart.*

3. *The effect of the SBC coding rate on the performance of the system is analysed with the aid of EXIT charts in the context of unequal error protected Joint Source-Channel Decoding (JSCD).*

4. *We also propose a serially concatenated three-stage scheme for near-capacity operation. In contrast to the two-stage system constituted by a single iterative loop, the three-stage system employs two iterative loops, which exchange extrinsic information both between the inner and the intermediate decoder, as well as between the outer decoder and the intermediate decoder. The resultant iterations are referred to as the inner and outer iterations, respectively.*

In the rest of the chapter we offer different system designs with gradually improved architectures, organised as follows.

A system's architecture, which is based on RSC codes aided unequal video error protection and SP modulation assisted near-capacity DSTS is detailed in Figure 5.3 of Section 5.2. In Section 5.3, the iterative USSPA decoding scheme of Figure 5.7 is used in addition to UEP employing RSC codes and SP modulation aided DSTS transmissions in order to exploit the residual redundancy that remains in the video coded stream after encoding. The resultant USSPA assisted UEP aided bit-stream was transmitted using the DSTS based SP modulation scheme of Figure 5.11 for attaining a diversity gain without the need for high-complexity MIMO channel estimation. Furthermore, SBCs were applied in Section 5.4, in order to intentionally introduce additional channel coding redundancy, by appropriately apportioning the total available bit-rate budget between the source and channel codecs. The employment of SBCs assists the outer decoder's EXIT curve in reaching to the $(1, 1)$ point of the EXIT chart. The SBC coded stream after RSC encoding was transmitted using the SP modulation aided DSTS transceiver of Figure 5.11. In Section 5.5, the performance of SBCs was analysed using the system architecture of Figure 5.11. Additionally, in Section 5.5.3 EXIT charts were utilised in ordered to analyse the effect of the SBC coding rate on the achievable performance of the UEP iterative JSCD strategies of Table 5.14(a). Furthermore, in Figure 5.34 of Section 5.6.2 we considered jointly optimised three-stage source and channel decoding, while employing serially concatenated and iteratively decoded SBCs combined with a Unity Rate Code (URC) and multi-dimensional SP modulation. The resultant coded signal was transmitted by a non-coherently detected DSTS MIMO transceiver designed for near-capacity JSCD, as shown in Figure 5.34. Moreover, in Section 5.6.2 we considered a jointly optimised serially concatenated three-stage scheme designed

for near-capacity wireless video transmission. More specifically, we employed serially concatenated and iteratively decoded SBCs combined with a URC and multi-dimensional SP modulation. The resultant coded signal was transmitted by a non-coherently detected DSTS MIMO transceiver designed for near-capacity JSCD, as shown in Figure 5.34. Finally, the chapter's conclusions are provided in Section 5.7, leading to a brief summary in Section 5.8.

5.2 Unequal Error Protection Video Using RSC Codes and SP Aided DSTS

In Section 3.10 we employed the UEP based H.264 coded video transmission scheme of Section 3.9 applying different-rate RSC codes to the three different types of H.264 bit-stream partitions of Section 3.8. In this section we will evaluate the performance of the error protection schemes proposed in Section 3.10. Instead of the single-input and single-output transmission scheme of Section 3.10, in this section we use a sophisticated low-complexity SP modulation aided MIMO DSTS scheme [250] employing two transmit and one receive antenna, while dispensing with CIR estimation.

5.2.1 Sphere Packing Modulation Based Orthogonal Design

The space time signal employing space-time coding aided two transmit antennas is defined by Alamouti's STBC scheme [229]

$$G_2(x_1, x_2) = \begin{bmatrix} x_1 & x_2 \\ -x_2^* & x_1^* \end{bmatrix}, \tag{5.1}$$

With x_1^* denoting the complex conjugate of x_1. Then, $G_2(x_1, x_2)$ constitutes an orthogonal design of size (2×2), which maps the complex variables representing (x_1, x_2) to two transmit antennas. In other words, (x_1, x_2) represents two complex modulated symbols to be transmitted from two transmit antennas in $T = 2$ time slots. In Equation 5.1 the rows and columns represents the temporal and spatial dimensions, corresponding to the two consecutive time slots and the two transmit antennas, respectively. It was shown in [246] that the diversity product quantifying the coding advantage of an orthogonal transmit-diversity scheme is determined by the minimum Euclidean distance of the vectors (x_1, x_2). Let us assume that the signal that has to be transmitted by the two antennas in two consecutive time slots consists of L legitimate space-time signals $G_2(x_{l,1}, x_{l,2})$, $l = 0, 1, \cdots, L-1$, where L represents the number of SP modulated symbols. For example, when jointly transmitting a pair of QPSK symbols, we need L=16 SP signals. The aim of of SP modulation is to design $x_{l,1}$ and $x_{l,2}$ jointly, so that they provide the best minimum Euclidean distance with respect to the remaining $(L-1)$ legitimate transmitted space-time signals in order to improve the system's error resilience. According to [231] for example, x_1 and x_2 represent the conventional BPSK modulated symbols transmitted over the 1^{st} and 2^{nd} time slot, but the design was not optimised by considering the joint constellation for the various x_1 and x_2 signal combinations. Let $(a_{l,1}, a_{l,2}, a_{l,3}, a_{l,4})$, $l = 0, 1, \cdots, L-1$, be the phaser points of the four-dimensional real-valued Euclidean space R^4, where each of the four elements $a_{l,1}, a_{l,2}, a_{l,3}$, and $a_{l,4}$ gives

one coordinate of the phaser point. Generally, $x_{l,1}$ and $x_{l,2}$ can be written as,

$$\begin{aligned}\{x_{l,1}, x_{l,2}\} &= T(a_{l,1}, a_{l,2}, a_{l,3}, a_{l,4}) \\ &= \{a_{l,1} + ja_{l,2}, a_{l,3} + ja_{l,4}\}. \end{aligned} \tag{5.2}$$

In the four-dimensional real-valued Euclidean space R^4, the sphere packing lattice D_4 is defined in a way that each constellation point has the best possible minimum Euclidean distance from all the other $(L-1)$ legitimate constellation points in R^4 [251]. More specifically, D_4 is defined as a lattice of SP constellation points having integer coordinates of $[a_1, a_2, a_3, a_4]$ describing the $x_{l,1}$ and $x_{l,2}$ symbol combinations, while satisfying the SP constraint of $a_1 + a_2 + a_3 + a_4 = k$, where k is an even integer.

5.2.2 Near Capacity Differential Space Time Spreading

Space time coding constitutes an effective diversity technique of compensating the effects of wireless channels by exploiting the independent fading of the two antennas signal. The advantage of Space-time coding is to achieve a substantial diversity and power gain relative to its single-input and single-output counterpart, achieved without any bandwidth expansion. There are numerous coding structures that are capable of achieving diversity, including the simple Space-Time Block Codes (STBC) proposed by Alamouti [229]. Similarly, Hochwald *et al*. [231] proposed the transmit diversity concept known as STS for the downlink of Wideband Code Division Multiple Access (WCDMA) systems. However, these STBC and STS techniques use coherent detection and require channel estimation at the receiver, which is acquired by transmitting known training symbols. However, mentioned above, the channel estimation increases both the cost and complexity of the receiver. More specifically, when the channel experiences fast fading, a commensurately increased number of training symbols has to be transmitted, which results in a high transmission overhead and wastage of transmission power. Relative to these mentioned techniques, DSTS constitutes a low-complexity MIMO-aided technique that does not require channel estimation [252]. However, this low-complexity is achieved at the cost of 3 dB performance loss in comparison to the more complex coherent receivers.

5.2.3 Twin-Antenna Aided Differential Space Time Spreading

The DSTS encoder can be divided into two primary stages, consisting of differential encoding and space-time spreading, as shown in Figure 5.1. The mapped symbols of Figure 5.1 are first differentially encoded and then they are spread using STS. The STS of two symbols transmitted using two transmit antennas within two time slots is exemplified in simple graphical terms in Figure 5.2 [253]. According to DSTS encoding algorithm, at time $t = 0$ the arbitrary dummy reference symbols v_0^1 and v_0^2, carrying no information, are passed to the STS encoder. At time $t \geq 1$, a block of $2B$ bits arrive at the mapper, where each set of B bits is mapped to a symbol x_t^k, where k is selected from a 2^B-ary constellation. The differentially encoded symbols v_t^1 and v_t^2 of Figure 5.1 are obtained as follows:

$$v_t^1 = \frac{(x_t^1 \cdot v_{t-1}^1) + x_t^2 \cdot v_{t-1}^{2*}}{\sqrt{(|v_{t-1}^1|^2 + |v_{t-1}^2|^2)}} \tag{5.3}$$

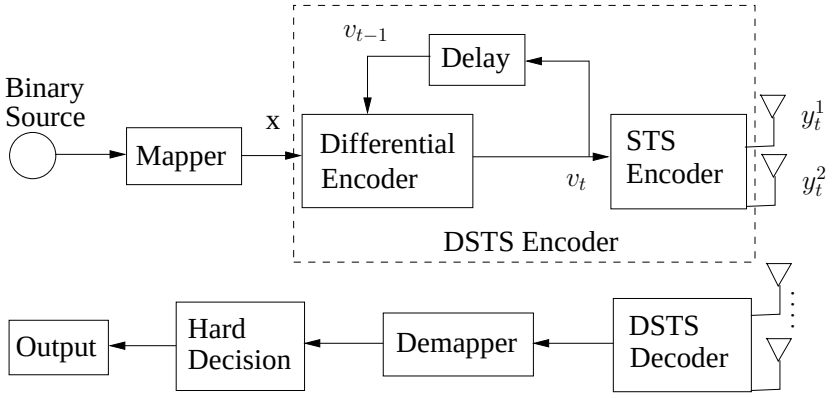

Figure 5.1: The DSTS system block diagram

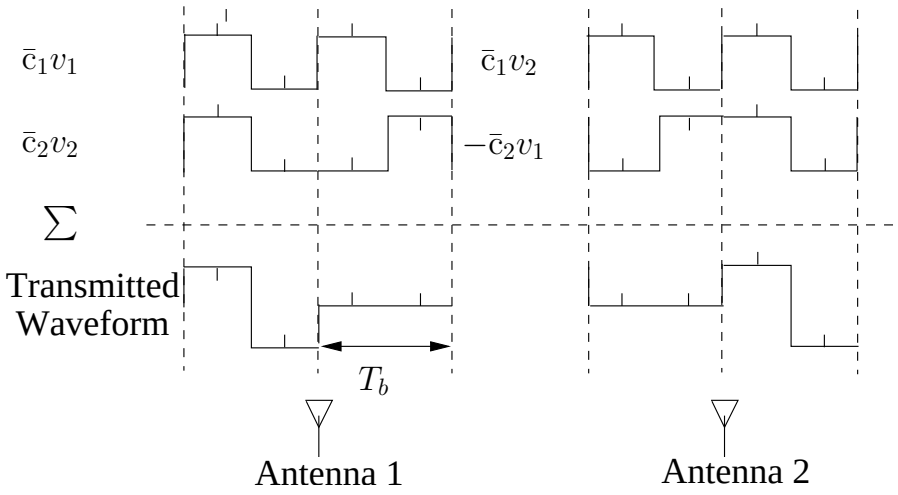

Figure 5.2: STS illustration using two transmit antennas for transmitting two bits within $2T_b$ duration. Assuming $v_1 = v_2 = 1$ where $\bar{c}_1 - [+1+1-1-1+1+1+1-1-1]$ and $\bar{c}_2 - [+1+1-1-1-1-1-1+1+1]$.

$$v_t^2 = \frac{(x_t^1 \cdot v_{t-1}^1) - x_t^2 \cdot v_{t-1}^{1*}}{\sqrt{(|v_{t-1}^1|^2 + |v_{t-1}^2|^2)}}, \tag{5.4}$$

where the superscript * represents the complex conjugate operation.

The differentially encoded symbols v_t^1 and v_t^2 are then spread using the orthogonal spreading codes $\bar{c}_1 - [+1+1-1-1+1+1-1-1]$ and $\bar{c}_2 - [+1+1-1-1-1-1+1+1]$, as shown in Figure 5.2. Finally, the differentially encoded data is divided into two half-rate substreams and the two consecutive symbols are then spread to two-transmit antennas as follows:

$$y_t^1 = \frac{1}{\sqrt{2}}(\bar{c}_1 \cdot v_t^1 + \bar{c}_2 \cdot v_t^{2*}) \tag{5.5}$$

$$y_t^2 = \frac{1}{\sqrt{2}}(\bar{c}_1 \cdot v_t^2 - \bar{c}_2 \cdot v_t^{1*}) \tag{5.6}$$

as exemplified graphically in Figure 5.2.

5.2.4 System Overview

The schematic of the proposed ISCD model of our videophone scheme is shown in Figure 5.3. At the transmitter side, the video sequence is compressed using the H.264/AVC video codec. The video source signal provided as input to the system is encoded using the JM 13.2 encoder of the H.264/AVC standard, which was highlighted in Section 3.5. We considered the source coding parameters presented in Section 3.5. The output bit-stream representing a video frame described by B source coded bits x_i, i = 1,2, ..., B, is de-multiplexed into three different sensitivity bit-streams, namely $StreamA$, $StreamB$ and $StreamC$, containing the sequentially concatenated partitions of type A, B and C of all the slices per frame, respectively. The de-multiplexer's binary output sequences of x_a, x_b, and x_c, where we have a = 1,2, ..., b_a, b = 1,2, ..., b_b, c = 1,2, ..., b_c, and B = b_a+b_b+b_c, are then interleaved using the bit-interleavers Π of Figure 5.3, into the interleaved sequences x'_a, x'_b, and x'_c, as shown in Figure 5.3 and are encoded by RSC codes having different code-rates, as summarised in Table 5.3(a).

We considered the different EEP and UEP system setups described in Section 4.4 for evaluating their effect on the performance of the H.264 coded video. The different code rates of the EEP and UEP schemes considered is presented in Table 5.3(a), which is equivalent to Table 4.4(a).

5.2.5 Transmitter and Receiver

The unequal error protected and partitioned H.264/AVC bit-stream y_i of Figure 5.3 is transmitted using a SP aided DSTS assisted transmitter [172]. Its performance is evaluated when communicating over a temporally correlated narrowband Rayleigh fading channel. The DSTS transceiver is capable of achieving a substantial diversity gain, which results in a beneficial BER reduction and in a concommitant subjective video quality improvement. As shown in Figure 5.3, the video source bits are interleaved and convolutionally encoded using our triple-class RSC error protection scheme and are subsequently multiplexed into a single bit-stream y_i. The multiplexed bit-stream y_i is then mapped by the SP mapper to the SP symbol stream s_i, $i = 0, 1, 2....L-1$, where we have $s_i = map_{sp}(y_i)$, and L represents the number of modulated symbols in the SP signalling alphabet, as described in [172]. Then the SP modulated symbols are transmitted using DSTS via two transmit antennas. We consider transmissions over a correlated narrowband Rayleigh fading channel having a normalised Doppler frequency of $f_d = f_D T_s = 0.01$, where f_D is the Doppler frequency and T_s is the symbol duration. The received signal is also contaminated by the complex-valued AWGN $n = n_I + jn_Q$, where n_I and n_Q are two independent zero-mean Gaussian random variables having a variance of $\sigma_n^2 = \sigma_{nI}^2 = \sigma_{nQ}^2 = N_0/2$ per dimension, with $N_0/2$ representing the double sided noise power spectral density expressed in W/Hz [72]. The received signal is then DSTS decoded and the soft information is passed from the SP demapper to the respective RSC decoder after de-multiplexing the LLRs into three partitions corresponding to the partitions A, B and C. After RSC decoding the decoded streams A, B and C are re-multiplexed into their original syntax suitable for the H.264 decoder.

By using multiple transmit and receive antennas both transmitter and receiver diversity gain is achieved, which results in a substantial BER reduction and in a concomitant subjective quality improvement for the decoded video. According to the system setup shown in the Figure 5.3, the source bits generated are convolutionally coded, depending on the specific error protection scheme of Table 5.3(a). After convolution coding,

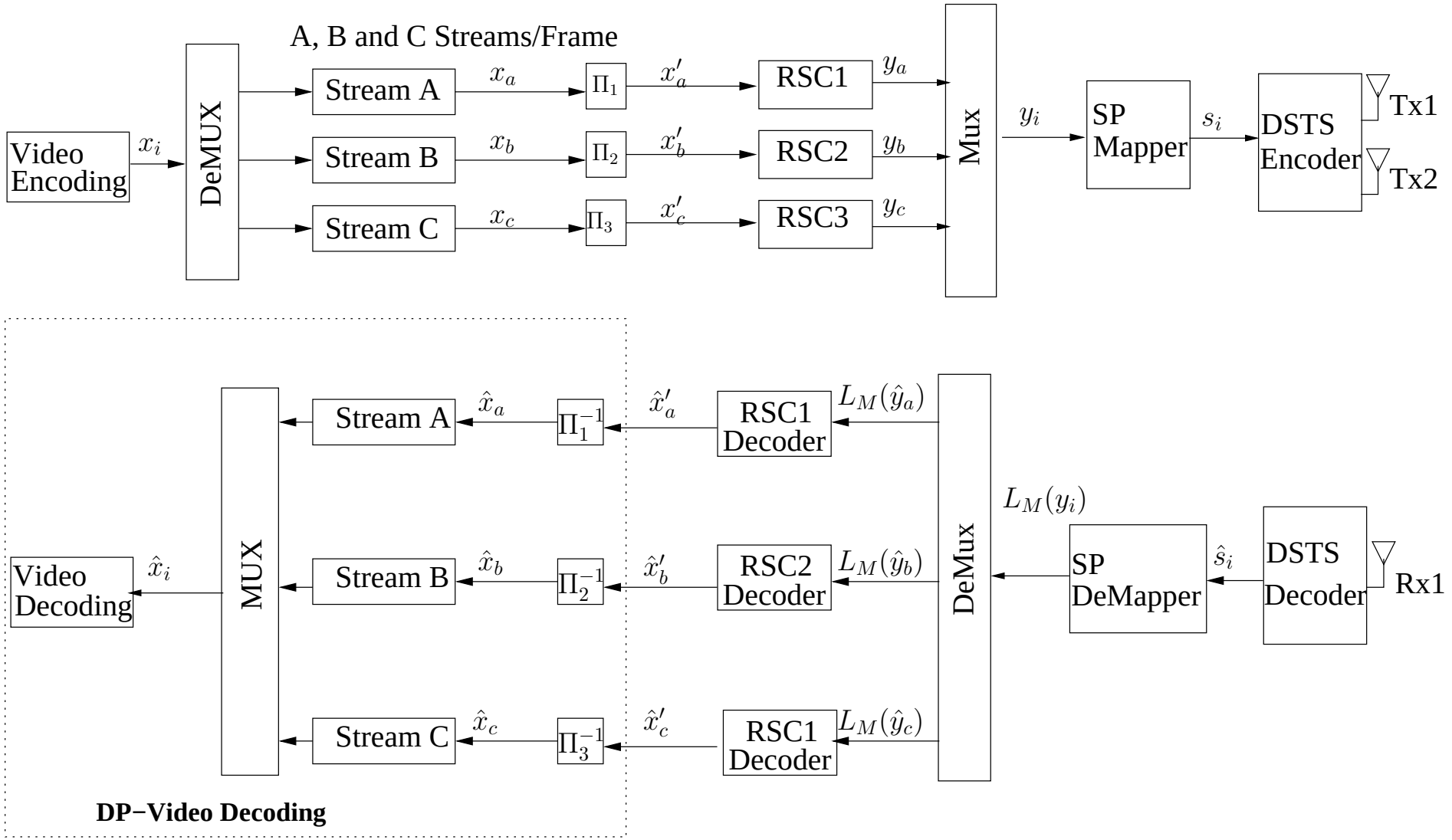

Figure 5.3: DP H.264 UEP with RSC codes. In contrast to Figure 3.23 of Section 3.10, this system employs a sophisticated SP-modulation aided DSTS transmission scheme. The corresponding system parameters were summarized in Table 5.3.

.

Error Protection Scheme	Code Rate			
	A	B	C	Overall
EEP	1/2	1/2	1/2	1/2
UEP1	1/3	2/3	3/4	1/2
UEP2	1/2	1/3	2/3	1/2

(a) Code rates for Different Error Protection schemes

System Parameters	**Value**	**System Parameters**	**Value**
Source Coding	H.264/AVC	Video Bit Rate (Kbps)	64
Source Codec	JM 12.1	Video Frame Rate (fps)	15
Video Test Sequence	*Akiyo*	Channel Coded Rate (Kbps)	128
No of Frames	45	Baud-rate (Kbps)	64
Frame Format	QCIF	Channel Coding	RSC
No of 'P' frames between	44	Over-all Code Rate	1/2
two 'I' frames		Code Memory	3
Time Period of 'I' frame (sec)	3	Generator Polynomials	
Use of 1/4-Pixel Motion Estimation	Yes	RSC 1/2,2/3 & 3/4 (G_1, G_2)	$(5,7)_8$
Intra-frame MB update/frame	3	RSC 1/3 (G_1, G_2, G_3)	$(3,7,5)_8$
per QCIF frame		Modulation Scheme	SP(L=16)
Use of 'B' Pictures	No	Number of Transmitters, N_t	2
Use of FMO	No	Number of Receivers, N_r	1
No of Frames Used for	1-Frame	Channel	Correlated Rayleigh Fading
Inter-Frame Motion Search		Normalised Doppler Frequency	0.01
No of Video Slices/frame	9	Interleaver Length	$\approx (64000/15)$
No of Video MBs/Slice	11	No System Iterations I_t	10
		Spreading Code	Walsh Code
		Spreading Factor	8

(b) Systems parameters used in the schematic of Figure 5.3

Table 5.3: Code rates and systems parameters used in the schematic of Figure 5.3

B_{sp} of the channel coded bits are mapped by SP mapper to the SP symbols S^l, and $l = 0, 1, 2....l-1$ and $b = b_0 \cdots (B_{sp} - 1) \in 0, 1$, so that we have $B_{sp} = log_2 L$, where L represents the number of modulated symbols combined into the sphere packed signalling alphabet. Then each of the four SP symbols is transmitted using DSTS via two transmit antennas during a single time slot [252]. We considered transmission over a correlated narrowband Rayleigh fading channel associated with a normalised Doppler frequency of $fd = f_d T_s = 0.01$, where f_d is the Doppler frequency and T_s is the symbol duration. The received signal is contaminated by the complex AWGN , $n = n_I + jn_Q$, where n_I and n_Q are two independent zero-mean Gaussian random variables with variance $\sigma_n^2 = \sigma_{nI}^2 = \sigma_{nQ}^2 = N_0/2$ per dimension, with $N_0/2$ representing the double sided noise power spectral density expressed in W/Hz [252]. The received signal is DSTS decoded and the soft information is passed from the SP demapper to the respective RSC decoders after de-multiplexing the LLRs into three partitions corresponding to the symbol-based partitions A, B and C. Subsequently, the bit-streams obtained after

RSC decoding are de-multiplexed into their original input order and are decoded by the H.264 decoder in order to generate the reconstructed video frame sequence, which is then used to calculate the PSNR performance of the system at the E_b/N_0 value considered.

5.2.6 Performance Results

The performance results obtained for the transceiver scheme of Figure 5.3 are presented in this section. For the sake of increasing the confidence level of our results we ran each experiment 160 times and recorded the average results. The resultant BER vs E_b/N_0 curves of the error protection scheme of Table 5.3(a) are presented in Figure 5.4, while the BER vs E_b/N_0 curves of each partition A, B and C are given in Figure 5.5, which characterises the relative performance of the different types of partitions, depending on the code-rate of the RSC code given in Table 5.3(a). Furthermore, the performance of the error protection schemes characterized in terms of the objective video quality is presented in the form of PSNR-Y vs E_b/N_0 curves in Figure 5.6. It may be observed from Figure 5.6, that the high error protection UEP2 scheme of partition A provides the best performance, followed by the UEP1 scheme, which performs better than EEP. Explicitly an E_b/N_0 gain of 4 dB is attained using UEP1 with reference to UEP2, while an E_b/N_0 gain of 2 dB is achieved by the latter with reference to the EEP scheme of Figure 5.3, at the PSNR degradation point of 2 dB. Recall that the system architecture of Section 3.10 is identical to that used in this section apart from the former dispensing with SP modulation aided DSTS transmission. Therefore, in comparing PSNR-Y vs E_b/N_0 results of Figure 5.6 to those of Figure 3.26 of Section 3.10, an E_b/N_0 gain of about 10 dB is attained. The major findings of the consider

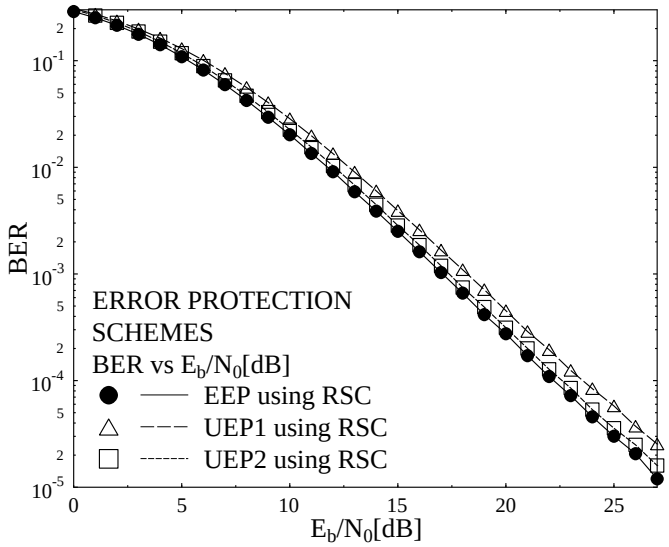

Figure 5.4: BER versus E_b/N_0 for EEP and UEP schemes of Figure 5.3 using the parameters of Table 5.3.

error protection schemes at PSNR degradation points of 1dB and 2dB are summarised in Table 5.4 In contrast to the UEP1 scheme, where partition A had the strongest protection, the rationale of this UEP2 arrangement is, that in low-motion video clips the video slices corrupted owing to the loss of partition A are concealed using the

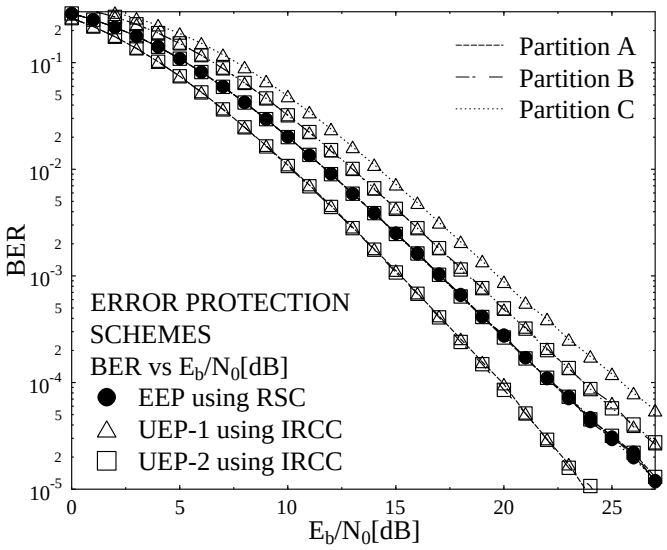

Figure 5.5: BER (A, B and C) versus E_b/N_0 for EEP and UEP schemes of Figure 5.3 using the parameters of Table 5.3.

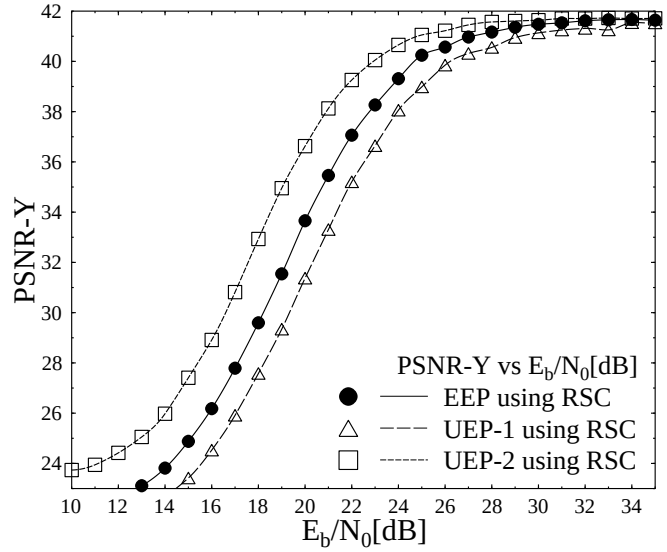

Figure 5.6: PSNR versus E_b/N_0 for EEP and UEP schemes of Figure 5.3 using the parameters of Table 5.3.

corresponding video-tile of the previously decoded frame, since this method results in less annoying artefacts owing to their high correlation with the adjacent frames. Additionally, reference to these concealed slices by future frames using motion compensated prediction results in adequate video reconstruction results. However, the loss of partition B along with a correctly received partition A of a particular slice implies that the MV and header information received in partition A will be used to decode the MBs of this slice encoded either in intra-

Table 5.4: Performance Summary Table for the system of Figure 5.3

Schematic	Parameters	Iterations	Video rate	FEC Code rate	Highest PSNR value	E_b/N_0 at $1dB$ PSNR degradation	E_b/N_0 at $2dB$ PSNR degradation
Figure 5.3	Table 5.3(a)	10	64 $kbps$	0.5	41.7582	EEP = 26 dB UEP1 = 28 dB UEP2 = 24 dB	EEP = 24 dB UEP1 = 24.5 dB UEP2 = 22 dB

or inter-frame prediction mode, while the all-important MBs of partition B represented as intra-frame MBs were completely lost. Furthermore, the future frames, and MBs referring to the lost intra-coded MBs of partition B in predictive coding will also result in a low video quality. Hence, providing high protection for partition B increases its probability to be correctly received, hence facilitating the reconstruction of every correctly received partition A. Therefore, similarly to partition A, partition B is also important and hence its corruption results in significant PSNR degradation of the decoded video sequence owing to the resultant avalanche-like error propagation, when the intra-frame coded MB updates are corrupted. Furthermore, a deficient selection of MBs for intra-frame MB update at the encoder results in poor error resilience and in an eroded coding efficiency. Therefore, the H.264/AVC codec select those specific MBs for intra-frame coding whose erroneous recovery cannot be appropriately concealed at the decoder, for example because they have a relatively high motion activity. In videophone sequences the head and shoulder areas tend to have a higher motion activity than the background, therefore the loss of intra-frame MB updates results in perceptually more annoying artefacts in the visually more important portions of the frame. Furthermore, typically partition B tends to have the lowest number of bits, therefore the employment of a strong FEC code does not result in a significant increase of the overall rate. As a result, using the remaining fraction of the total FEC-coded bit-rate budget, a reasonable protection can still be provided for partitions A and C.

5.3 RSC Coded and SP Aided DSTS for Unequal Source-Symbol Probability Aided Error Protection

In this system configuration the iterative scheme of Figure 4.6 detailed in Section 4.3.4 is employed in addition to UEP using RSC codes and the SP modulation aided DSTS transmission system of Section 5.2. This scheme has the additional capability of improving the attainable error resilience and to enhance the subjective video quality by exploiting the residual redundancy, which inherently remains in the H.264 coded bit-stream after source encoding and is used to determine the extrinsic information. The presence of residual redundancy manifests itself in the non-uniform M-ary symbol probability distribution $P(s_k)$, $s_k = [b^k(1), b^k(2), \cdots b^k(M)]$ as shown in Table 5.5, constituted by M symbols.

As in Section 4.4 this distribution is obtained by partitioning the H.264 coded bit-stream into a 3-bit/symbol sequence, yielding $[b_1(1), b_2(1), b_3(1), b_4(1), b_4(2), b_4(3), ...] = (S_1, S_2, ...S_p)$ [227]. For our simulations the H.264 video encoded bit-stream of the 300-frame *"Akiyo"* video sequence, the 150-frame *"MissAmerica"* video clip and the 300-frame *"Mother&Daughter"* video sequences were used as training sequences. The

Table 5.5: Probability of Occurance for 8-ary Symbols

Symbol	Probability
0	0.22791640
1	0.11727422
2	0.11945398
3	0.10481402
4	0.11365674
5	0.10745756
6	0.10268064
7	0.10674644

reason for selecting these specific video sequences is that all of them have relatively limited head-motion without any rapid background object motion or screen changes, which is typical in video-phone scenarios.

5.3.1 Unequal Source-Symbol Probability Aided Iterative Source Decoding

The details of the algorithm employed for determining the extrinsic information were presented in Section 4.3.4, and can also be found in [220]. For the sake of completeness, a brief review of the USSPA algorithm of Figure 5.3 is given below. In our system setup channel coding is provided by the RSC codes applied, therefore the reliability information expressed in form of the so-called Log-Likelihood values (L-values) [195] of each bit $x_n(m)$, where n denotes the discrete-time M-ary symbol index, is provided by the soft-output of the RSC channel decoder applied, which is expressed as

$$L_0(m) = ln\frac{P(x_0(m) = +1/\hat{Y})}{P(x_0(m) = -1/\hat{Y})}, \tag{5.7}$$

where $\hat{Y}$ represents the input sequence of the channel decoder and we have $m = 0, 1 \cdots, (M-1)$, with M being the number of bits per M-ary symbol. In the above equation $n = 0$ specifies the current time index.

It may be readily shown that, the corresponding bit error probability can be derived from (5.7) as

$$P_{e0}(m) = \frac{1}{1 + exp|L_0(m)|}. \tag{5.8}$$

By expressing the instantaneous bit-error probability as shown above, the M-ary symbol transition probability can be calculated by first determining the conditional probabilities of the transmitted bits $\hat{x}_0(m)$ given the received bits $x_0^{(i)}(m)$ as:

$$P[\hat{x}_0(m)|x_0^{(i)}(m)] = \begin{cases} 1 - p_{e0}(m) & \text{if } \hat{x}_0(m) = x_0^{(i)}(m) \\ p_{e0}(m) & \text{if } \hat{x}_0(m) \neq x_0^{(i)}(m). \end{cases} \tag{5.9}$$

Assuming the channel to be memoryless, the M-ary symbol transition probabilities, which are related to each $log_2 M$-bit stream M-ary symbol can be formulated as [254]:

$$P(\hat{x}_0|x_0^{(i)}) = \prod_{m=0}^{M-1} P[\hat{x}_0(m)|x_0^{(i)}(m)]. \tag{5.10}$$

This expression suggests that the M-ary symbol transition probabilities are constituted by the product of the conditional bit probabilities of the constituent bits which holds, when the bits are independent of each other. More explicitly, Equation 5.10 provides the probability of transition from any of the M possible transmitted bit combinations $x_0^{(i)}, i \in 0,1,2\cdots 2^M-1$, to the known received bit combination $\hat{x}_0$, again, provided that all these bits are independent of each other.

We can extract the extrinsic channel output information for each desired bit $x_0(\lambda)$ by excluding the λ^{th} bit of each M-ary symbol, yielding

$$P(\hat{x}_0^{ext}|x_0^{(i)ext}) = \prod_{m=0,m\neq\lambda}^{M-1} P[\hat{x}_0(m)|x_0^{(i)}(m)]. \tag{5.11}$$

Finally, the extrinsic LLR of each bit can be generated by combining its channel output information (LLR values) and the *a priori* knowledge of the k^{th} parameter, $x_0^{(k)}$, which is given by [220] [254],

$$L(x_0^{(k)}(\lambda)) = log\left(\frac{\sum_{x_0^{(k)ext}} P(x_0^{(k)ext}|x_0^{(k)}(\lambda)=+1).P(\hat{x}_0^{(k)ext}|x_0^{(k)ext})}{\sum_{x_0^{(k)ext}} P(x^{(k)ext}|x_0^{(k)}(\lambda)=-1).P(\hat{x}_0^{(k)ext}|x_0^{(k)ext})}\right), \tag{5.12}$$

$$L(x_0^{(k)}(\lambda)) = log\left(\frac{\sum_{x_0^{(k)ext}} P(x_0^{(k)ext}|x_0^{(k)}(\lambda)=+1).\prod_{m=0,m\neq\lambda}^{M-1} P(\hat{x}_0(m)|x_0^{(i)}(m))}{\sum_{x_0^{(k)ext}} P(x^{(k)ext}|x_0^{(k)}(\lambda)=-1).\prod_{m=0,m\neq\lambda}^{M-1} P(\hat{x}_0(m)|x_0^{(i)}(m))}\right). \tag{5.13}$$

5.3.2 System Overview

In this system study the performance of H.264 video encoded transmission was investigated in the context of a SP modulation aided DSTS transmission system. More specifically, the video stream of Section 3.5 was selected as the source video stream. The different code rates allocated to the various error protection schemes used for evaluating the performance of this type of setup were described in Section 4.4 and are briefly summarised here in Table 5.14(a). For convenience the schematic of the proposed system setup is presented in Figure 5.7, where we considered communication over non-dispersive correlated Rayleigh fading channels. The schematic of this system is equivalent to that of Figure 4.4 described in Section 5.2.4, with the additional inclusion of USSPA decoding. Further details concerning the transceiver structure utilised in this architecture are presented in Section 5.2.5.

5.3.3 Unequal Source-Symbol Probability Aided Iterative Source-Channel Decoding

As seen in Figure 5.7, following DSTS detection of the received signal, the soft information gleaned from the SP demapper is de-multiplexed into three streams, corresponding to the partitions A, B and C described in Section 3.8 and are passed to the respective RSC decoders in order to generate extrinsic information. The extrinsic

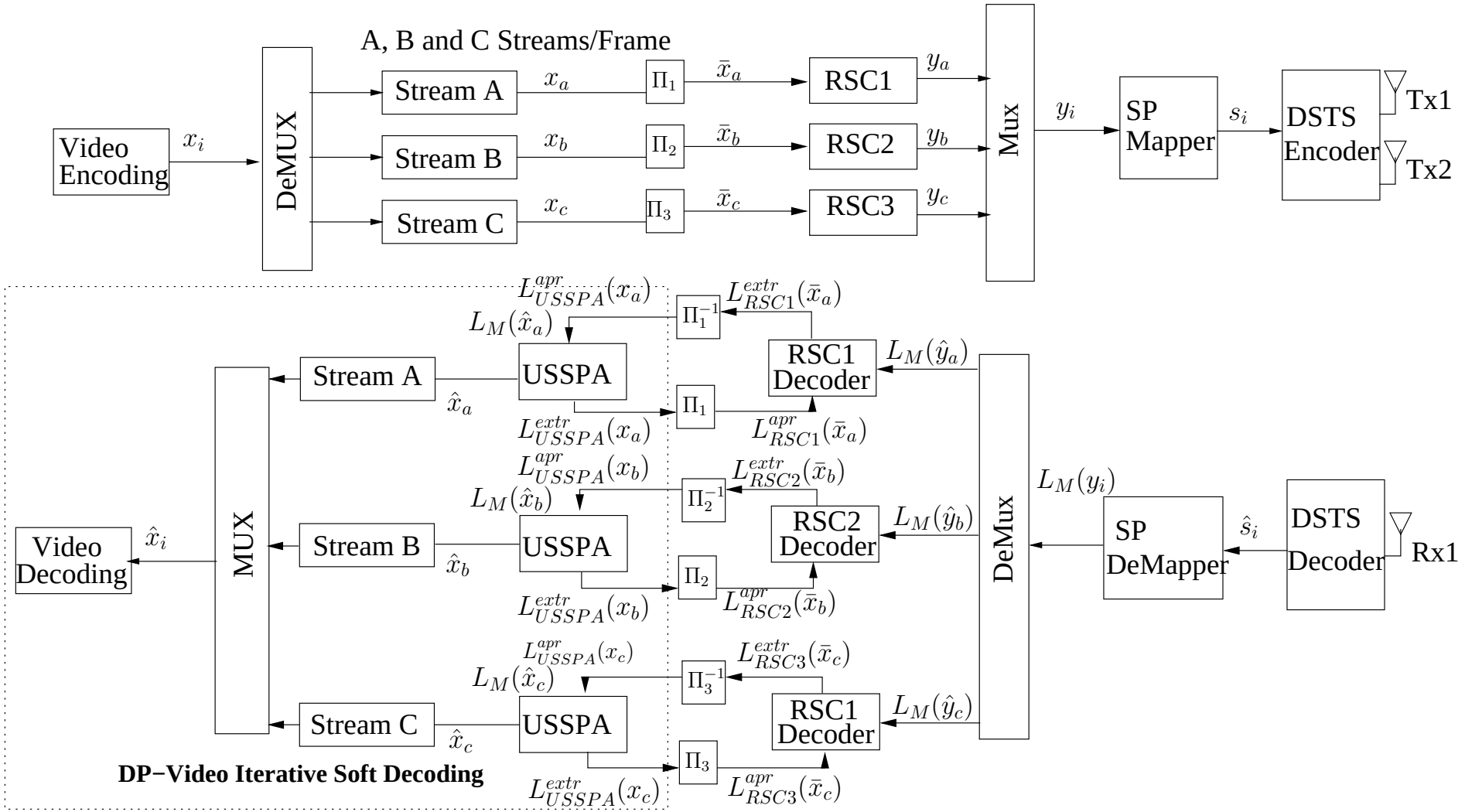

Figure 5.7: USSPA video UEP using RSC codes. All system parameters were summarised in Table 5.6. In contrast to Figure 5.3 of Section 5.2, this system employs an RSC inner code before invoking SP modulation aided DSTS transmission and USSPA based iterative decoding.

Error Protection Scheme	Code Rate				
	A	B	C	USSPA	Overall
EEP	1/2	1/2	1/2	1	1/2
UEP1	1/3	2/3	3/4	1	1/2
UEP2	1/2	1/3	2/3	1	1/2

(a) Code rates for Different Error Protection schemes

System Parameters	**Value**	**System Parameters**	**Value**
Source Coding	H.264/AVC	Video Bit Rate (Kbps)	64
Source Codec	JM 12.1	Video Frame Rate (fps)	15
Video Test Sequence	*Akiyo*	Channel Coded Rate (Kbps)	128
No of Frames	45	Baud-rate (Kbps)	64
Frame Format	QCIF	Channel Coding	RSC
		Outer Code	USSPA
No of 'P' frames between	44	Over-all Code Rate	1/2
two 'I' frames		Code Memory	3
Time Period of 'I' frame (sec)	3	Generator Polynomials	
Use of 1/4-Pixel Motion Estimation	Yes	RSC 1/2,2/3 & 3/4 (G_1, G_2)	$(5,7)_8$
Intra-frame MB update/frame	3	RSC 1/3 (G_1, G_2, G_3)	$(3,7,5)_8$
per QCIF frame		Modulation Scheme	QPSK
Use of 'B' Pictures	No	Number of Transmitters, N_t	2
Use of FMO	No	Number of Receivers, N_r	1
No of Frames Used for	1-Frame	Channel	Correlated Rayleigh Fading
Inter-Frame Motion Search		Normalised Doppler Frequency	0.01
No of Video Slices/frame	9	Interleaver Length	$\approx (64000/15)$
No of Video MBs/Slice	11	No System Iterations I_t	10

(b) Systems parameters used in the schematic of Figure 5.7

Table 5.6: Code rates and systems parameters used in the schematic of Figure 5.7

information gleaned is then exchanged between the RSC decoders and the USSPA decoder of Figure 5.7 [188]. More explicitly, the soft-information obtained after QPSK demodulation in the form of the LLR representation $L_M(\hat{y})$ is forwarded to the corresponding RSC decoder. As seen in Figure 5.7, the RSC decoder processes this input information and the *a priori* information $L^{apr}_{RSC}(\bar{x})$ gleaned from the outer decoder in order to generate the extrinsic LLR values $L^{extr}_{RSC}(\bar{x})$, which are subsequently deinterleaved by the soft-value deinterleaver of Figure 5.7, yielding $L^{apr}_{USSPA}(x)$. Then, the soft values $L^{apr}_{USSPA}(x)$ are passed to the USSPA decoder, which uses the extrinsic information generation algorithm of [188] to compute the extrinsic LLR values $L^{extr}_{USSPA}(x)$. During the iterative decoding process the RSC decoder exploits the *a priori* information for the sake of providing improved *a priori* LLR values for the outer decoder, which in turn exploits the input LLR values for providing improved *a priori* information for the RSC decoder in the subsequent iteration. Further details about iterative decoding are provided in [137].

The USSPA decoding scheme of Figure 5.7 generates extrinsic information by exploiting the residual redundancy, which inherently remains in the H.264/AVC coded bit-stream after source encoding. The pres-

ence of residual redundancy manifests itself in terms of the non-uniform *M-ary* symbol probability distribution $P[s_n(k)]$, $s_n(k)$=$[s_n(1),s_n(2),\cdots\ s_n(M)]$ shown in Table 5.5, which was generated by partitioning the H.264/AVC encoded bit-stream into an n=3-bit/symbol sequence, yielding $[b_1(1),\ b_1(2),\ b_1(3),\ b_2(1),\ b_2(2),\cdots,b_k(3)]$=$[S_1,S_2,\cdots\ S_M]$.

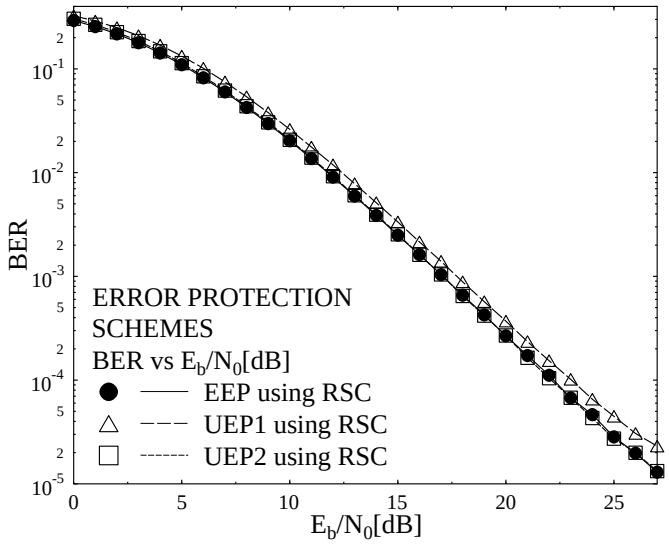

Figure 5.8: BER versus E_b/N_0 for the EEP and UEP schemes of Figure 5.7 using the parameters of Table 5.6. In contrast to Figure 4.7 of Section 4.4, which employed a similar coding scheme, but dispensed with SP aided DSTS transmissions a beneficial performance improvement of about 10 dB can be observed.

5.3.4 Performance Results

The performance results obtained for the system architecture depicted in Figure 5.7 is presented in this section. For the sake of improving the statistical reliability of our results, we ran each experiment 160 times and recorded the average results. The BER vs E_b/N_0 curves obtained using the system setup described in Section 5.3.2 and employing the various error protection schemes of Section 3.10.1 are presented in Figure 5.8. Additionally, the BER vs E_b/N_0 curves recorded for the individual partitions A, B and C are given in Figure 5.9, which characterises the relative performance of the different types of partitions, depending on the code-rate of the RSC codes given in Table 5.14(a). Furthermore, the performance of the error protection schemes applied was evaluated in terms of objective video quality and is presented in the form of PSNR-Y vs E_b/N_0 curves in Figure 5.10. It may be observed from Figure 5.10, that the UEP2 is associated with a high error protection for partition B and hence provides the best performance, while UEP1 performs better than EEP. Explicitly an E_b/N_0 gain of 3 dB is attained using UEP1 with reference to UEP2, while an E_b/N_0 gain of 1 dB is achieved with reference to EEP scheme, at the PSNR degradation point of 2 dB. Furthermore, we compared the PSNR-Y vs E_b/N_0 results obtained in this section to that of Section 4.4, which used similar error protection schemes, but dispensed with SP modulation aided DSTS transmissions. Its observed from Figure 5.10 that an E_b/N_0 gain of

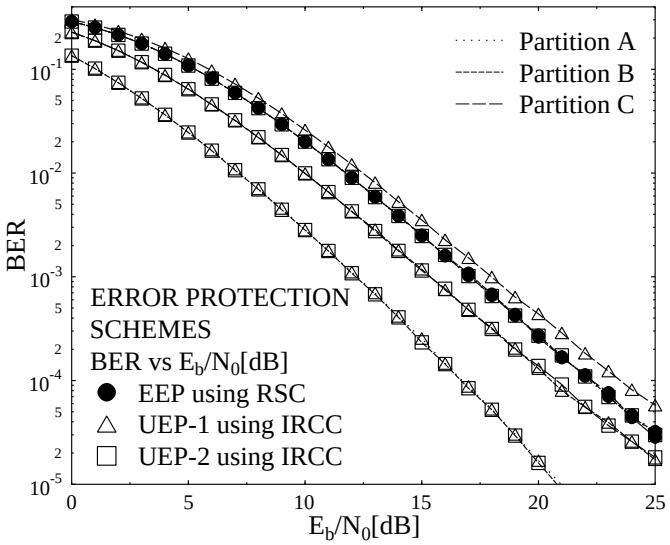

Figure 5.9: BER (A, B and C) versus E_b/N_0 for the EEP and UEP schemes of Figure 5.7 using the parameters of Table 5.6. In contrast to Figure 4.8 of Section 4.4, which employed a similar coding scheme, but dispensed with SP aided DSTS transmission a beneficial performance improvement of about 10 dB can be observed.

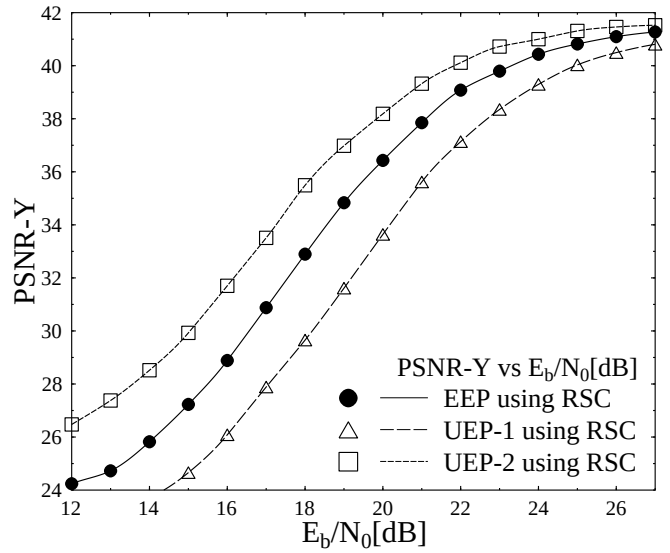

Figure 5.10: PSNR versus E_b/N_0 for the EEP and UEP schemes of Figure 5.7 using the parameters of Table 5.6. In contrast to Figure 4.9 of Section 4.4, which employed similar coding scheme, but dispensed with SP aided DSTS transmission an E_b/N_0 performance improvement of about $10dB$ can be observed.

about 10 dB can be achieved with reference to Figure 4.9 of Section 4.4, consisted of similar coding scheme but dispensing with SP aided DSTS transmission. The major findings of the consider error protection schemes at PSNR degradation points of $1dB$ and $2dB$ are summarised in Table 5.7

Table 5.7: Performance Summary Table for the system of Figure 5.7

Schematic	Parameters	Iterations	Video rate	FEC Code rate	Highest PSNR value	E_b/N_0 at $1dB$ PSNR degradation	E_b/N_0 at $2dB$ PSNR degradation
Figure 5.7	Table 5.6(a)	10	64 $kbps$	0.5	41.7582	EEP = 24 dB UEP1 = 26 dB UEP2 = 22.5 dB	EEP = 23 dB UEP1 = 24 dB UEP2 = 21 dB

5.4 SBC Assisted Unequal Error Protection Video Using RSC Codes and SP-Modulated DSTS

In this system additional SBCs are applied to the source coded stream, which are incorporated in the schematic of Section 5.3. Following RSC encoding the SBC coded stream is transmitted using a SP modulation aided DSTS transceiver. We utilised the transmitter and receiver architecture of Section 5.2.5 in order to evaluate the performance of our proposed SBCs. As described in Section 4.5.4, SBCs introduce intentional redundancy in the source coded bit-stream, which assists the outer decoder's EXIT curve to reach to the $(1, 1)$ point of EXIT chart. Clearly, SBCs may hence also be referred to as EXIT Chart Optimised Short SBCs (EOSBCs), which constitute low-complexity block codes.

5.4.1 Short Block Codes

As described in Section 4.3.2, the residual redundancy in the source coded bit-stream can be exploited for joint USSPA decoding, which has the capability of reducing the BER and of improving the subjective video quality. As described in Section 5.3.1, the performance of the USSPA decoder of Figure 5.11 is dependent on the amount of residual redundancy or correlation inherent in the source coded parameters after source coding, which may be exploited to mitigate the effects of transmission errors. However, in the presence of efficient state-of-the-art source codecs, such as the H.264 scheme, only limited redundancy is left in the source encoded bit-stream, which may result in limited extrinsic information contribution for the USSPA decoder, hence it may result in a modest further performance improvement beyond two decoding iterations. Therefore, in the SBC-aided scheme of this section we intentionally increased the amount of redundancy in the encoded bit-stream with the aid of SBC mapping. One of the advantages of using K-bit to N-bit SBC mapping associated with $N > K$ is that only 2^K out of the 2^N-bit symbols will occur in the bit-stream, which results in a non-uniform symbol probability distribution. In our system setup we used different rate SBCs as the outer code, as detailed in Section 4.5.4. However, here we initially introduce the concept using a $\frac{5}{6}$-rate SBC example. The resultant 5-bit to 6-bit mapping is shown in Table 5.8. The different code rates allocated to the various video protection classes

and used for evaluating the achievable performance of this type of architecture are given in Table 5.10(a).

Table 5.8: Input symbols and their corresponding SBC coded symbols.

Input	Output Symbol	Input	Output Symbol
00000	000000	10000	010000
00001	100001	10001	110001
00010	100010	10010	110010
00011	000011	10011	010011
00100	100100	10100	110100
00101	000101	10101	010101
00110	000110	10110	010110
00111	100111	10111	110111
01000	001000	11000	011000
01001	101001	11001	111001
01010	101010	11010	111010
01011	001011	11011	011011
01100	101100	11100	111100
01101	001101	11101	011101
01110	001110	11110	011110
01111	101111	11111	111111

The SBCs of Table 5.8 are generated using the design procedure of [2] described in Section 4.5.1. According to the SBC encoding procedure of [2], first a redundant bit r_τ is generated for the τ-*th M-ary* source word by calculating the XOR function of its N constituent bits, according to $r_\tau = [b^\tau(1) \oplus b^\tau(2)... \oplus b^\tau(N)]$; where $\oplus$ represents the XOR operation. The resultant redundant bit can be incorporated in any of the $[N+1]$ different bit positions of the resultant SBC, in order to create $[N+1]$ different SBC-encoded word combinations, as depicted in Table 5.9, each having a minimum Hamming distance of $d_{H,min}$ = 2 from all the others. In

Table 5.9: [N+1] Different SBC combinations.

Symbols	C_1	C_2	. . .	C_{N+1}
$S_{(1)}$	$r_1 b_1 b_2..b_N$	$b_1 r_1 b_2..b_N$	.	$b_1 b_2..b_N r_1$
$S_{(2)}$	$r_2 b_1 b_2..b_N$	$b_1 r_2 b_2..b_N$	.	$b_1 b_2..b_N r_2$
:	:	:	:	:
$S_{(2^N)}$	$r_{2^N} b_1 b_2..b_N$	$b_1 r_{2^N} b_2..b_N$	.	$b_1 b_2..b_N r_{2^N}$

Table 5.8 we provide the resultant output symbols for the specific case of incorporating the redundant bit r_τ at the end of the τ-*th K-bit* source symbol.

5.4.2 System Overview

The schematic of the proposed system architecture is presented in Figure 5.11, which is equivalent to that described in Section 5.3, with the additional inclusion of a SBC. We considered the H.264 coded stream of Section 3.5 as the input of the schematic of Figure 5.11. More specifically, at the transmitter side, the video sequence is compressed using the H.264/AVC video codec. Then, the output bit-stream representing a video frame consisting of B source coded bits x_i, i = 1,2, ..., B, is de-multiplexed into three different bit-streams, namely $SteamA$, $StreamB$ and $StreamC$, containing the sequentially concatenated partitions of type A, B and C of all the slices per frame, respectively. The de-multiplexer's binary output sequences x_a, x_b, and x_c, where we have a = 1,2, ..., b_a, b = 1,2, ..., b_b, c = 1,2, ..., b_c, and $B = b_a+b_b+b_c$, are then encoded to the bit-strings x'_a, x'_b, and x'_c using a SBC. Subsequently, the SBC-encoded bit-strings are interleaved using the bit-interleavers Π of Figure 5.11, into the interleaved sequences $\bar{x}_a$, $\bar{x}_b$ and $\bar{x}_c$, as shown in Figure 5.11 and are encoded by RSC codes having different code-rates, as summarised in Table 5.10(a). In the iterative decoder of Figure 5.11, interleaving and de-interleaving are necessary for ensuring that the bits are input to the constituent decoders in their expected original order, while maintaining the statistical independence of the extrinsic LLRs. Since the degree of the statistical independence guaranteed by an interleaver is always related to its length [185], concatenation of the bits generated by the MBs of a slice within a given partition results in a longer interleaver without extending the video delay and hence improves the achievable performance of iterative decoding.

5.4.3 Short Block Code Based Iterative Source-Channel Decoding

The purpose of ISCD is to extract the highest possible extrinsic information [$L^{extr}_{USSPA}(\acute{x})$ and $L^{extr}_{RSC}(\bar{x})$] from the constituent inner and outer decoders in order to assist each other in an iterative fashion. The algorithm used to find the extrinsic LLR values of a zero-order Markov model can be found in [255] and is detailed in Section 4.3.4. As mentioned above, owing to the limited amount of residual redundancy in the H.264 decoded stream, the additional performance improvements achieved by iterative joint source and channel decoding is limited. Hence, for the sake of intentionally introducing redundancy we will specifically design the SBC mapping or coding for reaching the $(1,1)$ point of the EXIT chart and hence for approaching capacity. At the receiver side, after DSTS decoding of the received signal, the soft information gleaned from the SP demapper of Figure 5.11 is de-multiplexed into three UEP video streams corresponding to the partitions A, B and C and are passed to the respective RSC decoder in order to generate extrinsic information. The extrinsic information gleaned is then exchanged between the RSC decoders and SBC decoder. More explicitly, the soft-information obtained after QPSK demodulation in the form of its LLR representation $L_M(\hat{y})$ is forwarded to the corresponding RSC decoder. The RSC decoder processes this input information and the fed back *a priori* information $L^{apr}_{RSC}(\bar{x})$ from the outer decoder in order to generate the extrinsic LLR values $L^{extr}_{RSC}(\bar{x})$, which are subsequently deinterleaved by the soft-bit interleaver of Figure 5.7, yielding $L^{apr}_{USSPA}(x)$. Then, the soft bits $L^{apr}_{USSPA}(x)$ are passed to the SBC decoder, which uses the extrinsic information generation algorithm of [188] to compute the extrinsic LLR values $L^{extr}_{USSPA}(x)$. During the iterative decoding process the RSC decoder exploits the *a priori* information supplied by the outer decoder for the sake of providing improved *a priori* LLR values for the outer channel decoder of Figure 5.11, which in turn exploits the input LLR values glean from the inner decoder for

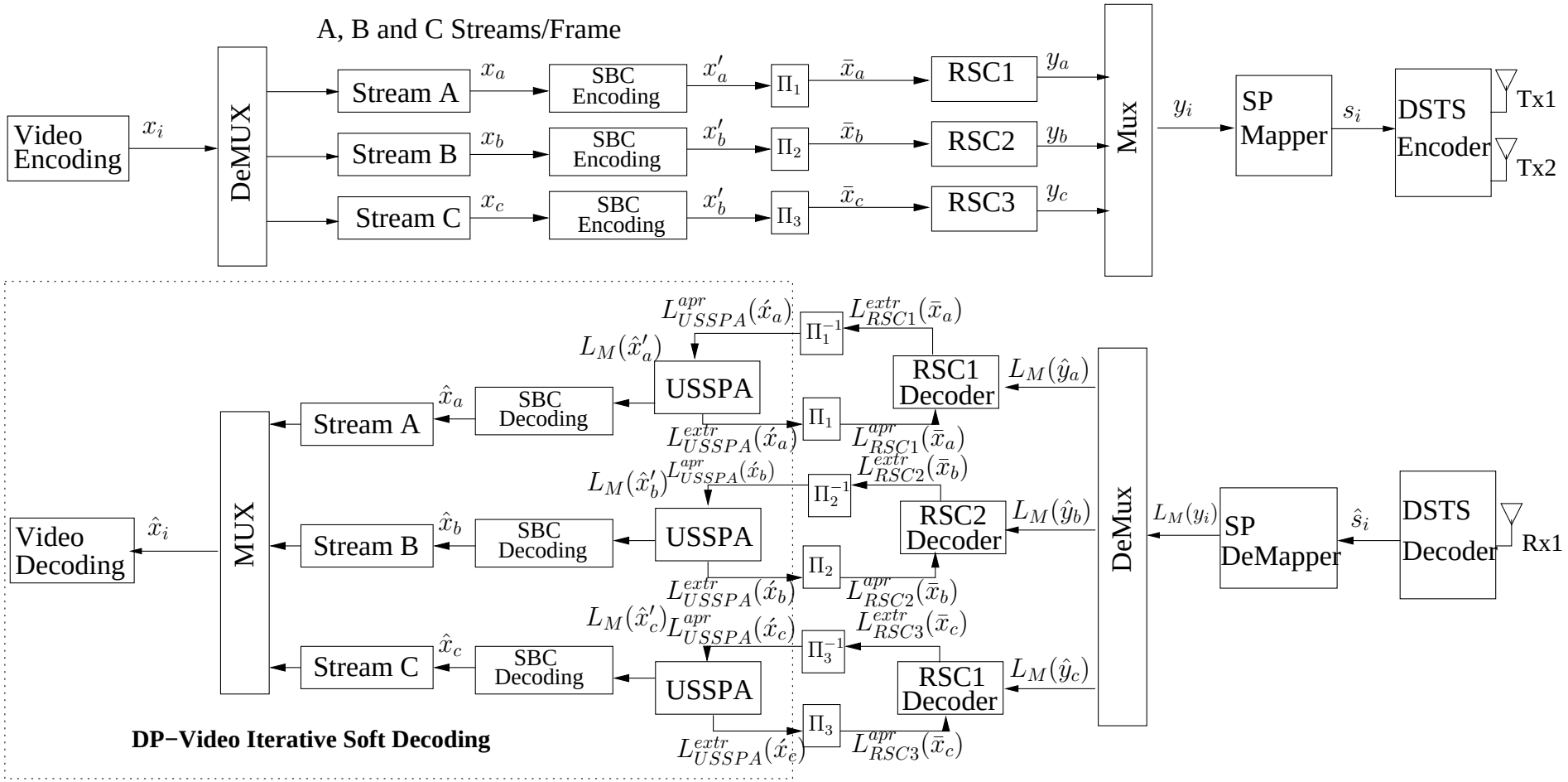

Figure 5.11: SBC assisted USSPA aided DP H.264 UEP iterative decoding with RSC codes while using SP modulation aided DSTS transmission system. All system parameters were summarized in Table 5.10. In contrast to Figure 5.7 of Section 5.3, this system employs SBC assisted USSPA based iterative decoding, while using an RSC as outer code.

Error Protection Scheme	Code Rate				
	A	B	C	SBC	Overall
EEP	3/5	3/5	3/5	5/6	1/2
UEP1	2/5	4/5	9/10	5/6	1/2
UEP2	3/5	2/5	4/5	5/6	1/2

(a) Code rates for Different Error Protection schemes

System Parameters	**Value**	**System Parameters**	**Value**
Source Coding	H.264/AVC	Video Bit Rate (Kbps)	64
Source Codec	JM 12.1	Video Frame Rate (fps)	15
Video Test Sequence	*Akiyo*	Channel Coded Rate (Kbps)	128
No of Frames	45	Baud-rate (Kbps)	64
Frame Format	QCIF	Channel Coding	RSC
No of 'P' frames between	44	Over-all Code Rate	1/2
two 'I' frames		Code Memory	3
Time Period of 'I' frame (sec)	3	Spreading Code	Walsh Code
Use of 1/4-Pixel Motion Estimation	Yes	Spreading Factor	8
Intra-frame MB update/frame	3	Generator Polynomials	
per QCIF frame		RSC 3/5,2/5,4/5, & 9/10 (G_1, G_2)	$(5,7)_8$
Use of 'B' Pictures	No	Number of Transmitters, N_t	2
Use of FMO	No	Number of Receivers, N_r	1
No of Frames Used for	1-Frame	Channel	Correlated Rayleigh Fading
Inter-Frame Motion Search		Normalised Doppler Frequency	0.01
No of Video Slices/frame	9	Interleaver Length	$\approx (64000/15)$
No of Video MBs/Slice	11	No System Iterations I_t	10
Modulation Scheme	SP(L=16)	SBC Rate	5/6

(b) Systems parameters used in the schematic of Figure 5.11

Table 5.10: Code rates and systems parameters used in the schematic of Figure 5.11

the sake of providing improved *a priori* information for the RSC decoder in the subsequent iteration. Further details about iterative decoding are provided in [137].

5.4.4 Performance Results

The performance of the schematic outlined in Figure 5.11 is evaluated using both the EEP and UEP schemes of Table 5.10(a) for the transmission of the H.264 encoded video stream of Section 3.5 by applying different-rate RSC codes to the video partitions A, B and C, as presented in Table 5.10(a). For the sake of fair comparison, the resultant overall rate of all EEP and UEP schemes is fixed to half-rate. The convolutional encoder used in our simulations is specified by the generator sequences $g_1 = [111]$ and $g_2 = [101]$, which may also be represented as $G = [1, g_2/g_1]$. Furthermore, as shown in Table 5.10(a), the fixed overall coding rate of $\frac{1}{2}$ was maintained by adjusting the puncturing rate of the RSC code applied. For the sake of maintaining a high statistical confidence level, we repeated each experiment 160 times and recorded the average results.

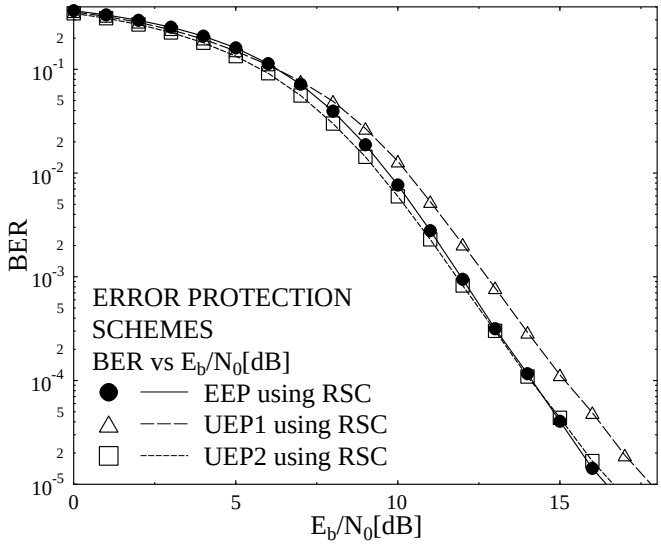

Figure 5.12: BER versus E_b/N_0 for the EEP and UEP schemes of Figure 5.11 using the parameters of Table 5.10. In this figure an improved performance can be observed in contrast to Figure 5.8 of Section 5.3, which employed a similar coding scheme but dispensed with SBC coding.

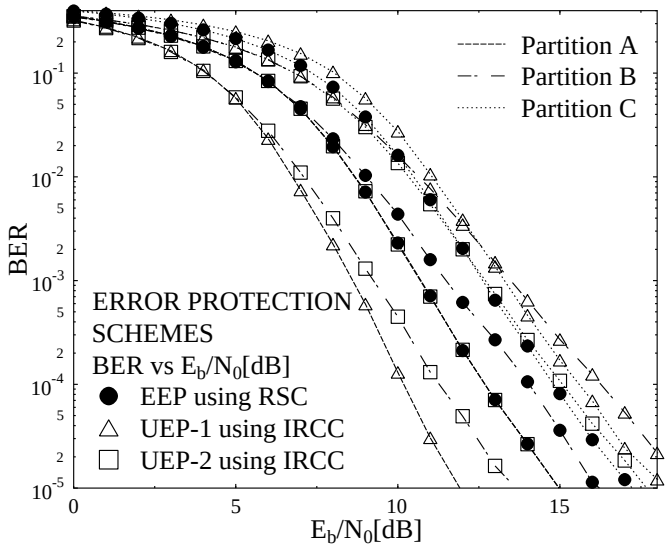

Figure 5.13: BER (A, B and C) versus E_b/N_0 for the EEP and UEP schemes of Figure 5.11 using the parameters of Table 5.10. In this figure an improved performance can be observed in contrast to Figure 5.9 of Section 5.3, which employed a similar coding scheme, but dispensed with SBC coding.

The BER vs E_b/N_0 curves obtained using the system architecture described in Section 5.4.2 for the various

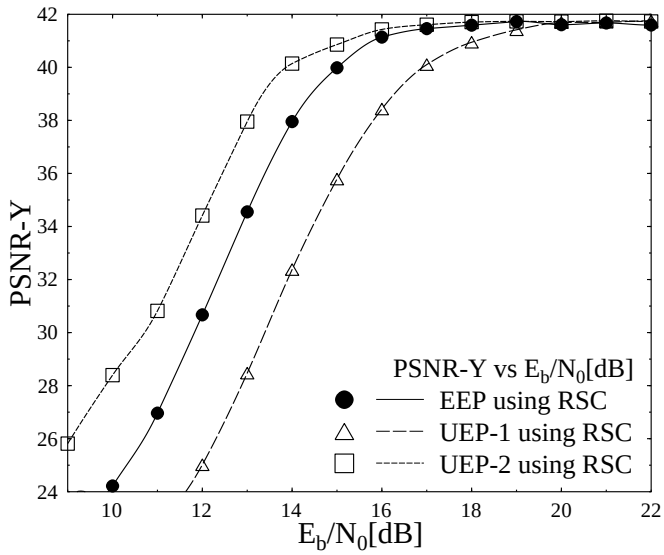

Figure 5.14: PSNR versus E_b/N_0 for the EEP and UEP schemes of Figure 5.11 using the parameters of Table 5.10. In this figure an improved performance can be observed in contrast to Figure 5.10 of Section 5.3, which employed a similar coding scheme, but dispensed with SBC coding.

error protection schemes of Table 5.10(a) are presented in Figure 5.12. Additionally, the BER vs E_b/N_0 curves recorded for the individual video partitions A, B and C are given in Figure 5.13, which characterises the relative performance of the different partitions, depending on the code-rate of the RSC codes given in Table 5.10(a). Furthermore, the performance of the various error protection schemes was evaluated in terms of the associated objective video quality, which is characterised in the form of the PSNR-Y vs E_b/N_0 curves of Figure 5.14. It may be observed from Figure 5.14 that the UEP2 scheme, which is associated with a high error protection for partition B provides the best performance, followed by the UEP1 scheme of Table 5.10(a), which performs better than EEP. Explicitly, an E_b/N_0 gain of 3 dB is attained using UEP2 with reference to UEP1, while an E_b/N_0 gain of 1 dB is achieved with reference to the EEP scheme, at the PSNR degradation point of 2 dB. Furthermore, we compared the PSNR-Y vs E_b/N_0 curves of Figure 5.14 obtained in this section to those of Figure 5.10 in Section 5.3, which were based on similar type of error protection schemes, but dispensing with SBCs. It is observed from Figure 5.14 that an E_b/N_0 gain of about 12 dB can be achieved with reference to Figure 5.10 of Section 5.3, which is an explicit benefit of our EOSBC. The major findings of the consider error protection schemes at PSNR degradation points of 1dB and 2dB are summarised in Table 5.11

5.5 Near Capacity EXIT-Chart Aided Iterative Source-Channel Decoding

In this section, we will analyse an ISCD based system design, for the transmission of H.264 coded videophone type low bit-rate video. The effect on the performance of the system due to variation in the Joint Source-Channel

Table 5.11: Performance Summary Table for the system of Figure 5.11

Schematic	Parameters	Iterations	Video rate	FEC Code rate	Highest PSNR value	E_b/N_0 at $1dB$ PSNR degradation	E_b/N_0 at $2dB$ PSNR degradation
Figure 5.11	Table 5.10(a)	10	64 *kbps*	0.5	41.7582	EEP = 14.5 *dB* UEP1 = 16.5 *dB* UEP2 = 13.5 *dB*	EEP = 15 *dB* UEP1 = 17.5 *dB* UEP2 = 14.5 *dB*

Decoding strategies is analysed with the aid of EXIT charts.

5.5.1 Iterative Source-Channel Decoding Model

The schematic of the proposed videophone type ISCD model is shown in Figure 5.11. The details about the SBC based ISCD is provided in Section 5.4.3. More specifically, at the transmitter side, the video sequence is compressed using the H.264/AVC video codec. Then, the output bit-stream representing a video frame consisting of B source coded bits x_i, i = 1,2, ..., B, is de-multiplexed into three different bit-streams, namely $SteamA$, $StreamB$ and $StreamC$, containing the sequentially concatenated partitions of type A, B and C of all the slices per frame, respectively. The de-multiplexer's binary output sequences x_a, x_b, and x_c, where we have a = 1,2, ..., b_a, b = 1,2, ..., b_b, c = 1,2, ..., b_c, and $B = b_a+b_b+b_c$, are then encoded to the bit-strings x'_a, x'_b, and x'_c using SBC. Subsequently, the SBC encoded bit-strings are interleaved using the bit-interleavers Π of Figure 5.11, into the interleaved sequences $\bar{x}_a$, $\bar{x}_b$ and $\bar{x}_c$, as shown in Figure 5.11 and are encoded by RSC codes having different code-rates, as summarised in Table 5.14(a). In the iterative decoder of Figure 5.11, interleaving and de-interleaving are necessary for ensuring that the bits are input to the constituent decoders in their expected original order, while maintaining the statistical independence of the extrinsic LLR. Since the degree of the statistical independence guaranteed by an interleaver is always related to its length [185], concatenation of the bits generated by the MBs of a slice within a given partition results in a longer interleaver without extending the video delay and hence improves the achievable performance of iterative decoding.

5.5.2 System Overview

In this improved system design study using EXIT charts, we adopt the schematic of Figure 5.11 as our underlying system model. At the transmitter side, the video sequence is compressed using the H.264/AVC video codec. More specific details about the source coding parameters considered were presented in Section 3.5. An overview of the system structure is provided in this section.

The unequal error protected and partitioned H.264/AVC bit-stream y_i of Figure 5.11 is transmitted using a SP aided DSTS assisted transmitter [172], described in Section 5.2.5. Its performance is evaluated when communicating over a temporally correlated narrowband Rayleigh fading channel. The DSTS transceiver is capable of achieving a substantial diversity gain, which results in a beneficial BER reduction and in a concommitant subjective video quality improvement.

The received signal is then DSTS decoded and the soft information is passed from the SP demapper to

the respective RSC decoder after de-multiplexing the LLRs into three partitions corresponding to the partitions A, B and C. The extrinsic information gleaned is then exchanged between the RSC decoders and USSPA decoder [188]. The presence of "residual redundancy" manifests itself in terms of the non-uniform M-ary symbol probability distribution $P(s_k)$, $s_k = [b^k(1), b^k(2), \cdots b^k(M)]$ shown in Table 5.12, which was generated by dividing the H.264/AVC encoded bit-stream into an M=3-bits/symbol sequence constituted by, i.e. $[b_1(1), b_1(2), b_1(3), b_2(1), b_2(2), b_2(3), \cdots, b_k(3)]$ = Irregular turbo-like code $[S_1, S_2, \cdots S_p]$.

In our simulations the H.264/AVC video encoded bit-stream of the 300 frame "Akiyo" sequence, the 150 frame "Miss America" clip and the 300-frame "mother & daughter" video sequence were used as training sequences. The reason for selecting these specific video sequences is that all of them are relatively low-motion 'head and shoulders' videophone sequences.

Table 5.12: Probabilities of the input symbols, obtained using the training sequences.

Symbol	Probability
0	0.2279
1	0.1172
2	0.1194
3	0.1048
4	0.1136
5	0.1074
6	0.1026
7	0.1067

As an example, our EXIT optimised rate-$\frac{3}{4}$ SBC is detailed in Table 5.13. As it becomes evident from Table 5.13, the EXIT-chart optimised SBCs ensures that the coded M-bit symbols exhibit a minimum Hamming distance of $d_{H,min}$=2. Which is the *sufficient and necessary* condition for the iterative detection convergence shown by Kliewer [163]. The conditions states that in order to guarantee that the EXIT curve of the combined source and channel coding to becomes capable of reaching the (I_A, I_E)=$(1, 1)$ point of perfect convergence, the legitimate source codewords should have a minimum Hamming distance of $d_{H,min}$=2. Additionally, only eight out of the sixteen possible 4-bit symbols are legitimate in the mapped source coded bit-stream, which exhibits a non-uniform probability of the 3-bit source symbols as shown in Table 5.13.

5.5.3 EXIT Characteristics of Short Block Codes

Figure 5.15 depicts the EXIT characteristics of using either the rate-1 or the rate-$\frac{3}{4}$ SBC scheme shown in Table 5.13. More specifically, the EXIT curve of rate-$\frac{3}{4}$ SBC does indeed reach to the top right corner of the EXIT chart at $(I_A, I_E) = (1, 1)$ and hence results in an infinitesimally low BER, provided that the *convergence tunnel* exists in combination with the considered inner code at the considered E_b/N_0 value. By contrast, the USSPA decoding scheme using rate-1 SBC fails to do so. In conclusion, our simulation results recorded for the system presented in Figure 5.11 reveal that the performance of USSPA decoding strongly depends on the presence or absence of residual source redundancy, which typically manifests itself in the form of

Table 5.13: SBC of Rate-1 and Rate-$\frac{3}{4}$

Symbol	Rate-1 SBC	Rate-$\frac{3}{4}$ SBC
000	000	0000
001	001	1001
010	010	1010
011	011	0011
100	100	1100
101	101	0101
110	110	0110
111	111	1111

less uniform distribution of the source encoded M-bit symbols. The EXIT curves for EEP, UEP1 and UEP2 error protection schemes using USSPA decoding with rate$\frac{3}{4}$ SBC as outer code and respective RSC code, mentioned in Table 5.14(a), as inner code is shown in Figure 5.16, Figure 5.17 and Figure 5.18. Observe from Figure 5.16, that the EEP scheme employing rate-$\frac{3}{4}$ SBC and rate-$\frac{4}{9}$ outer RSC code for partition A, B and C of H.264/AVC coded video results in the $convergence - tunnel$ at $E_b/N_0 = 11\ dB$. Therefore, all the three partitions will exhibit a "turbo-cliff" at $E_b/N_0 = 11\ dB$, due to the employment of equivalent inner and outer codes. Figure 5.17 shows that the UEP1 scheme employing rate-$\frac{3}{4}$ SBC and rate-$\frac{1}{3}$, rate-$\frac{8}{15}$ and rate-$\frac{2}{3}$ outer RSC codes for partition A, B and C results in the $convergence - tunnel$ at $E_b/N_0 = 10\ dB$, $E_b/N_0 = 12\ dB$ and $E_b/N_0 = 13\ dB$ respectively. While for the UEP2 scheme using rate-$\frac{3}{4}$ SBC and rate-$\frac{4}{9}$, rate-$\frac{1}{3}$ and rate-$\frac{8}{15}$ outer RSC codes for partition A, B and C, the $convergence - tunnel$ is achieved at $E_b/N_0 = 11\ dB$, $E_b/N_0 = 10\ dB$ and $E_b/N_0 = 12\ dB$ respectively.

Below we provide a brief summary of the EXIT-chart properties [161]:

- The area under the outer code's EXIT curve is given by the code-rate;
- The area in the open EXIT-tunnel is proportional to the E_b/N_0 distance from the system's capacity. Hence an infinitesimally small area coresponding to near-capacity operation; provided that the $(1, 1)$ point of the EXIT-chart is reach without having an intercept point between the two EXIT curves.
- If there is an intercept point before reaching the (1,1) point, a residual BER floor is expected.

To illustrate a bit further, Figure 5.19 present the decoding trajectory of Figure 4.10 while employing EEP scheme of Table 4.9(a), composed of $SBC_{[5,\ 15]}$ outer code and rate-3/4 inner RSC code. The BER performance of the employed iterative decoding of Figure 5.19 consisting of $I_t = 2, 4, 6$ and 10 system iterations is presented in Figure 5.20. Observe from the Figure5.20 that as expected the performance of the system improves upon increase in the number of iterations during the iterative decoding process, provided an open EXIT tunnel exists between the inner and outer EXIT curves.

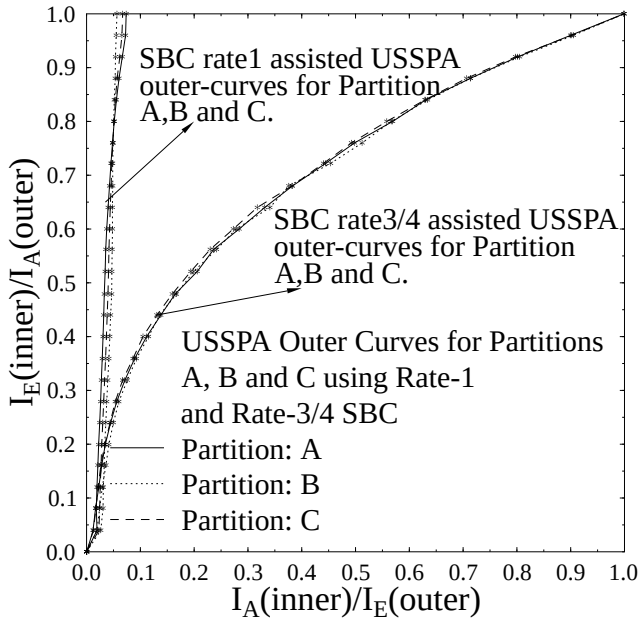

Figure 5.15: EXIT characteristics of USSPA with the aid of a rate-1 and rate-$\frac{3}{4}$ SBC of Figure 5.11 using the parameters of Table 5.14.

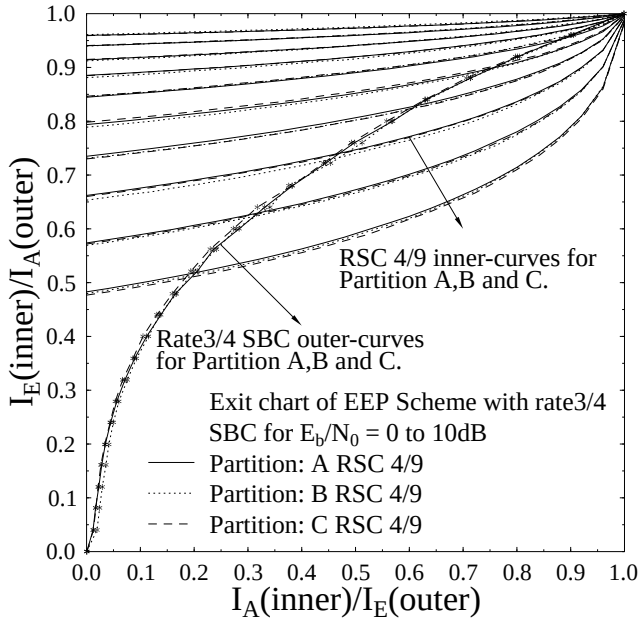

Figure 5.16: EXIT characteristics of the EEP scheme of Figure 5.11 using the parameters of Table 5.14.

5.5.4 Performance Results and Discussions

In this section, we present our performance results for the proposed ISCD system. The video source signal provided as input to our proposed system is encoded using JM 13.2 encoder of the H.264/AVC standard. We

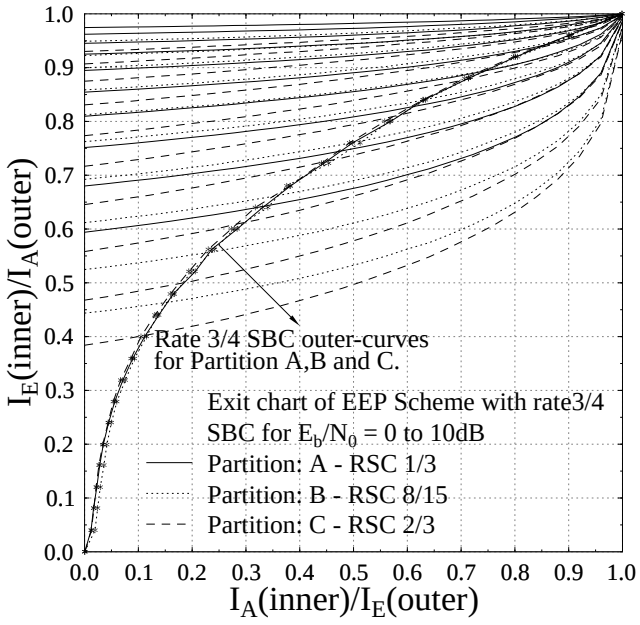

Figure 5.17: EXIT characteristics of the UEP1 scheme of Figure 4.26 using the parameters of Table 5.14.

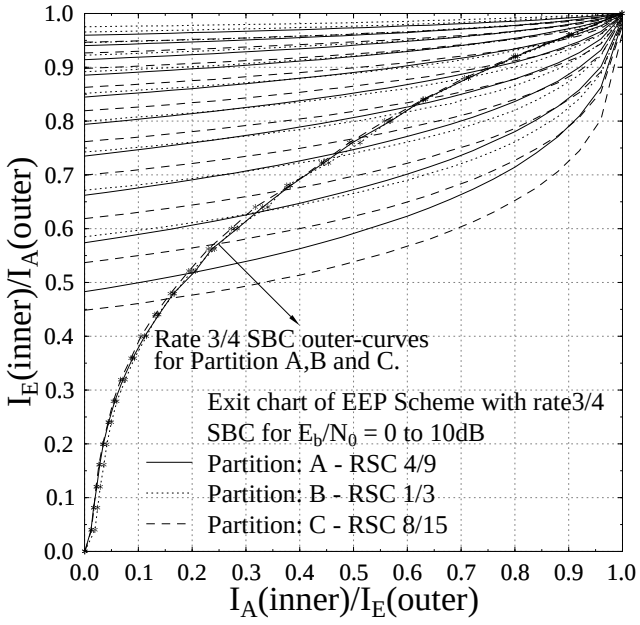

Figure 5.18: EXIT characteristics of the UEP2 scheme of Figure 5.11 using the parameters of Table 5.14.

used the video sequence of Section 3.5, consisted of an "Akiyo" video sequence [1] consisting of 45 (176x144)-pixels Quarter Common Intermediate Format (QCIF) frames at 15 frames-per-second (fps) at the target bit-rate of $64kbps$.

The remaining parameters of the proposed system model are listed in Table 5.14(b).

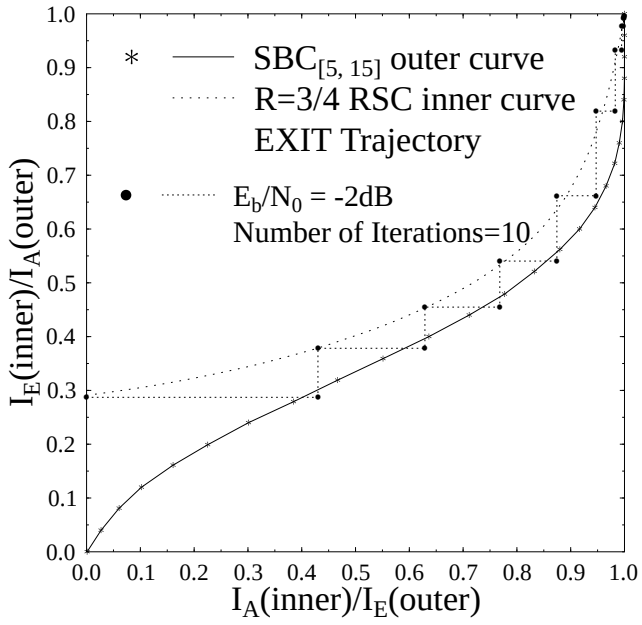

Figure 5.19: EXIT characteristics of the EEP scheme of Figure 5.11 using the parameters of Table 5.14, composed of the $SBC_{[5, 15]}$ outer code and rate-3/4 inner RSC code.

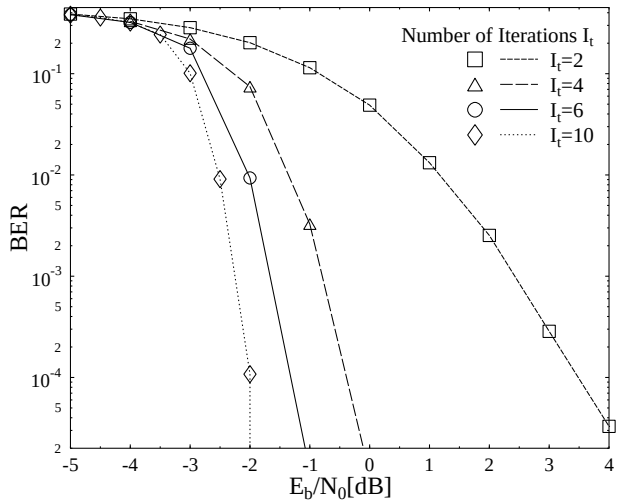

Figure 5.20: BER performance of the various error protection schemes of Table 5.14(a), composed of the $SBC_{[5, 15]}$ outer code and rate-3/4 inner RSC code, using various number of system iterations.

The error resilient encoding techniques, such as Flexible Macro-block Ordering (FMO) [74] and the employment of multiple reference frames for inter-frame motion compensation typically results in modest video performance improvements in low-motion head-and-shoulders video sequences, such as the ”Akiyo” clip, despite their considerably increased complexity. Therefore they were turned off. Additionally, only the immediately preceding frame was used for motion search, which results in a reduced computational complexity compared to using multiple reference frames. Furthermore, due to the limited residual redundancy inherent in

the source coded bit-stream characterised in Table 5.13 and for the sake of reducing the computational complexity imposed, we limited the number of iterations between the RSC and USSPA decoders to $I_t = 3$ iterations, when using rate-1 SBC, and to $I_t = 5$ iterations, when employing rate-$\frac{3}{4}$ SBC. For the sake of increasing the confidence in our results, we repeated each 45-frame experiment 160 times and averaged the results. The RSC codes applied to the three different-sensitivity interleaved bit-streams scheme had different code-rates and were combined with the SBC rates listed in Table 5.14(a). At the receiver seen in Figure 5.11, iterative soft-bit source

Error Protection Scheme		Code Rate				
		Type A	Type B	Type C	SBC	Overall
Rate-1 SBC	EEP	1/3	1/3	1/3	1	1/3
	UEP1	1/4	2/5	1/2	1	1/3
	UEP2	1/3	1/4	2/5	1	1/3
Rate-$\frac{3}{4}$ SBC	EEP	4/9	4/9	4/9	3/4	1/3
	UEP1	1/3	8/15	2/3	3/4	1/3
	UEP2	4/9	1/3	8/15	3/4	1/3

(a) Code rates for different Error Protection schemes of Figure 5.11. In contrast to Table 5.10(a) of Section 5.4.2, in this table an over-all coding rate of 1/3 is considered. Additionally, a system setup consisted of Rate-1 SBC is also presented.

System Parameters	**Value**	**System Parameters**	**Value**
Source Coding	H.264/AVC	Video Bit Rate (Kbps)	64
Source Codec	JM 12.1	Video Frame Rate (fps)	15
Video Test Sequence	*Akiyo*	Channel Coded Rate (Kbps)	192
No of Frames	45	Baud-rate (Kbps)	96
Frame Format	QCIF	Channel Coding	RSC
No of 'P' frames between	44	Over-all Code Rate	1/3
two 'I' frames		Code Memory	3
Time Period of 'I' frame (sec)	3	Spreading Code	Walsh Code
Use of 1/4-Pixel Motion Estimation	Yes	Spreading Factor	8
Intra-frame MB update/frame	3	Generator Polynomials (G_1, G_2, G_3)	$(11, 13, 15)_8$
per QCIF frame		Modulation Scheme	SP(L=16)
Use of 'B' Pictures	No	Number of Transmitters, N_t	2
Use of FMO	No	Number of Receivers, N_r	1
No of Frames Used for	1-Frame	Channel	Correlated Rayleigh Fading
Inter-Frame Motion Search		Normalised Doppler Frequency	0.01
No of Video Slices/frame	9	Interleaver Length	$\approx (64000/15)$
No of Video MBs/Slice	11	No System Iterations I_t	10

(b) Systems parameters used in the schematic of Figure 5.11

Table 5.14: Code rates and systems parameters used in the schematic of Figure 5.11

and channel decoding is applied by exchanging extrinsic information between the receiver blocks, which has the capability of improving the achievable subjective video quality. The performance of the system was evaluated

by keeping the same overall code rate as well as video rate for both the Equal Error Protection (EEP) and UEP coding schemes. In the UEP1 scheme of Table 5.14(a) the highest error protection was provided for partition A, while partition B was more strongly protected than C. By contrast, the UEP2 arrangement of Table 5.14(a) represents another error protection scheme, in which the highest level of error protection is provided for partition B, followed by partition A and C. As outlined in Section 3.3 that the actual decoding trajectories of the inner and outer constituent decoders is used to verify the EXIT chart based convergence prediction. Therefore, in order to verify the convergence prediction of our proposed EEP, UEP1 and UEP2 schemes, the actual decoding trajectories are recorded at E_b/N_0 = 9 dB and are presented in Figures 5.21, 5.22 and 5.23, respectively for partitions A, B and C having the respective inner code-rates shown in Table 5.14(a), while using rate-$\frac{3}{4}$ SBC. These trajectories were obtained by recording the Mutual Information (MI) at the input and output of both the inner and outer decoder during the bit-by-bit Monte-Carlo simulation of the iterative soft-bit source and channel decoding algorithm. The steps shown in the Figures 5.21, 5.22 and 5.23 represent the actual extrinsic information transfer between the USSPA decoder and the corresponding inner RSC channel decoder. As stated in Section 5.5, that the bits generated by the MBs of a slice within a given partition were concatenated in order to employ a longer interleaver, therefore the assumptions outlined at the beginning of Section 5.5.3 are justified and hence the EXIT chart based convergence prediction becomes accurate.

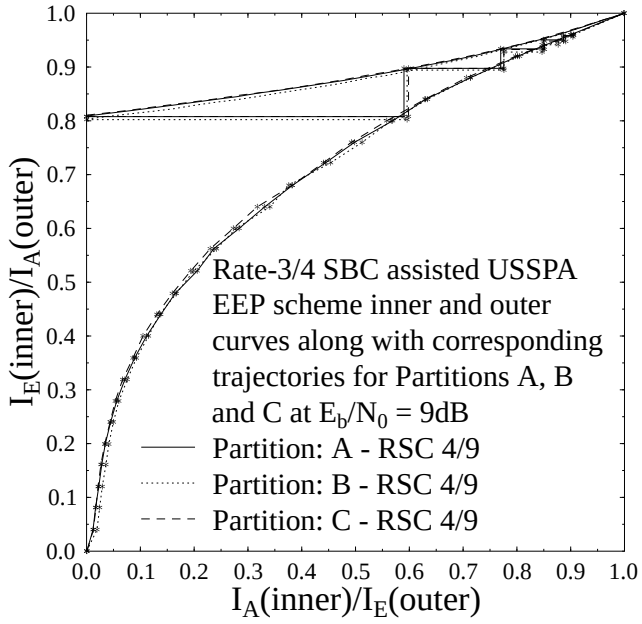

Figure 5.21: The EXIT chart and simulated decoding trajectories of the EEP scheme of Figure 5.11 using the parameters of Table 5.14 at E_b/N_0 = 9 dB.

Figures 5.24 and 5.25 show the performance of both the EEP and UEP schemes of Table 5.14(a) in terms of the attainable BER, when using the rate-1 and rate-$\frac{3}{4}$ SBC. Naturally, the three partitions of the EEP scheme experience a similar BER, since the same code rate was applied to them. Partitions B and C of the UEP1 scheme have a higher BER than partition A, which is a consequence of their higher code rate relative to partition A. Accordingly, the BER performance of partition B is better than that of partition C, due to its lower code rate.

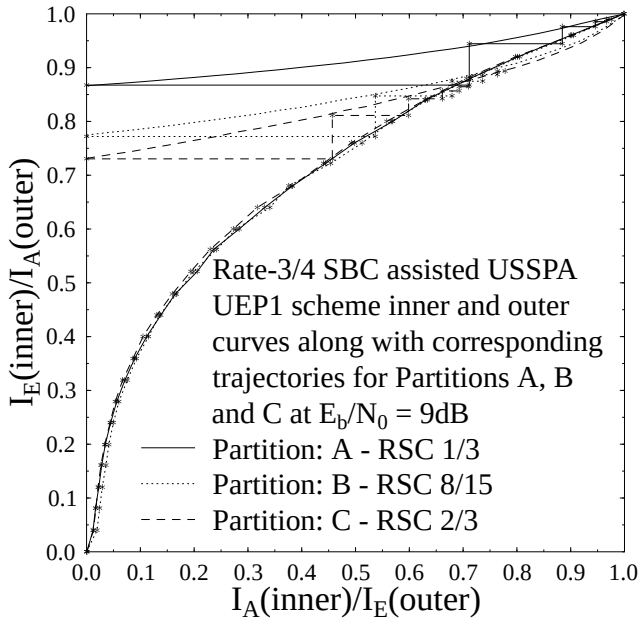

Figure 5.22: The EXIT chart and simulated decoding trajectories of the UEP1 scheme of Figure 5.11 using the parameters of Table 5.14 at E_b/N_0 = 9 dB.

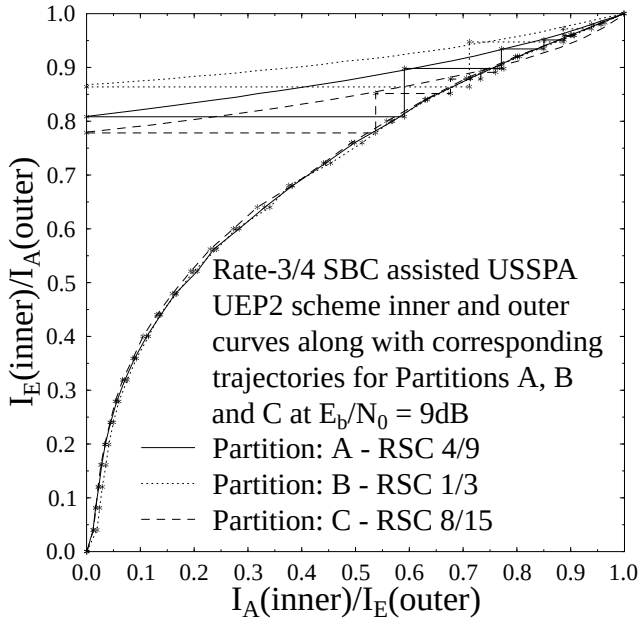

Figure 5.23: The EXIT chart and simulated decoding trajectories of the UEP2 scheme of Figure 5.11 using the parameters of Table 5.14 at E_b/N_0 = 9 dB.

Similarly, partition C of the UEP2 scheme has a higher error rate than partition A, which in turn has a higher error rate than partition B.

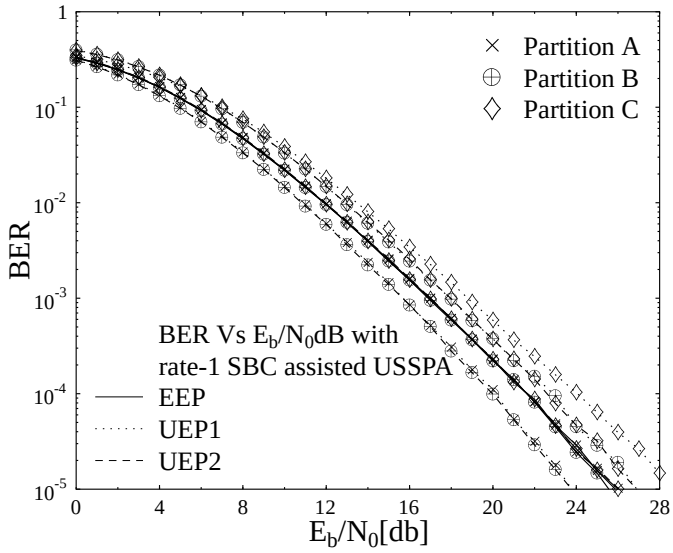

Figure 5.24: BER versus E_b/N_0 performance of the various error protection schemes of Figure 5.11 using the parameters of Table 5.14.

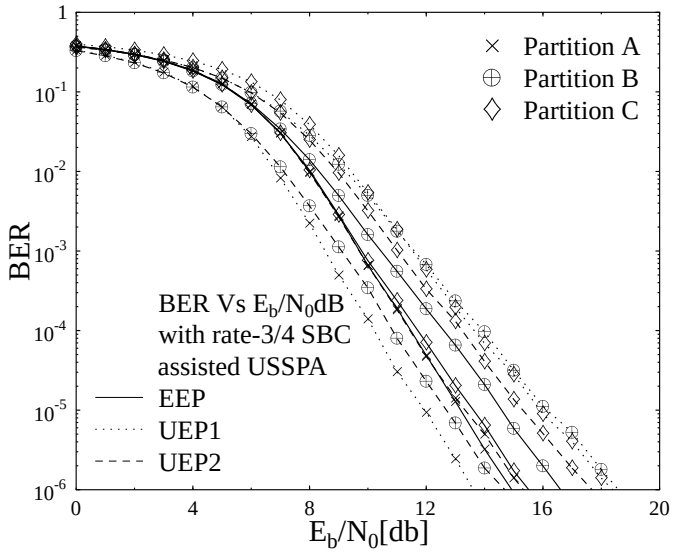

Figure 5.25: BER versus E_b/N_0 performance of the various error protection schemes of Figure 5.11 using the parameters of Table 5.14.

Observe in Figure 5.26 that while using USSPA in conjunction with rate-1 SBC, the UEP1 scheme result in a worse PSNR performance than the EEP and UEP2 arrangements, the UEP2 scheme provides the best PSNR among the three schemes for the entire E_b/N_0 region considered. Similar performance trends were observed for the different error protection schemes in Figure 5.27, while using rate-$\frac{3}{4}$ SBC, with an additional E_b/N_0 gain of 12 dB, relative to the rate-1 SBC. Additionally, the PSNR vs E_b/N_0 curves for the chrominance component of the video sequence is presented in Figure 5.30, 5.31, 5.32 and 5.33. Finally, the performance trends expressed in terms of the PSNR versus BER curves for the rate-1 SBC and rate-$\frac{3}{4}$ SBC are portrayed in Figures 5.28

and 5.29, respectively. More specific explanation about the reason behind the resultant performance trends due to applying the different error protection schemes is presented in Section 5.2.6. The major findings of the consider error protection schemes at PSNR degradation points of $1dB$ and $2dB$ are summarised in Table 5.15

Table 5.15: Performance Summary Table for the system of Figure 5.11

Schematic	Parameters	Iterations	Video rate	FEC Code rate	Highest PSNR value	E_b/N_0 at $1dB$ PSNR degradation	E_b/N_0 at $2dB$ PSNR degradation
Figure 5.11	Table 5.14(a)	10	64 $kbps$	0.333	41.7582	EEP = 13.5 dB UEP1 = 15.2 dB UEP2 = 13.0 dB	EEP = 13 dB UEP1 = 14.5 dB UEP2 = 12.5 dB

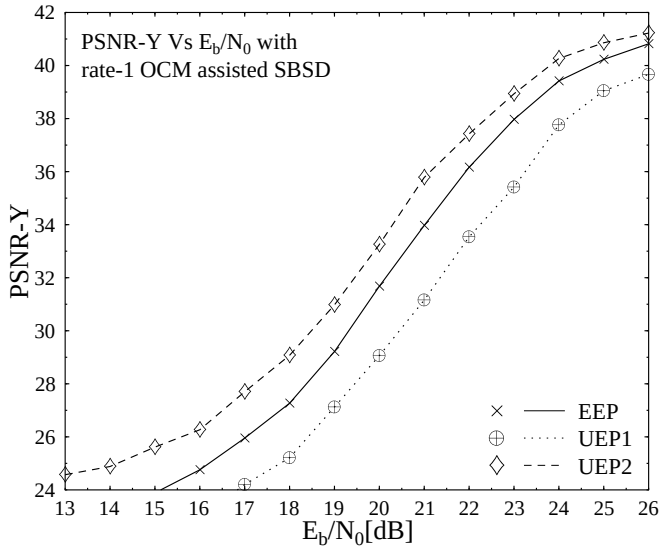

Figure 5.26: PSNR-Y versus E_b/N_0 performance of the various error protection schemes of Figure 5.11 using the parameters of Table 5.14.

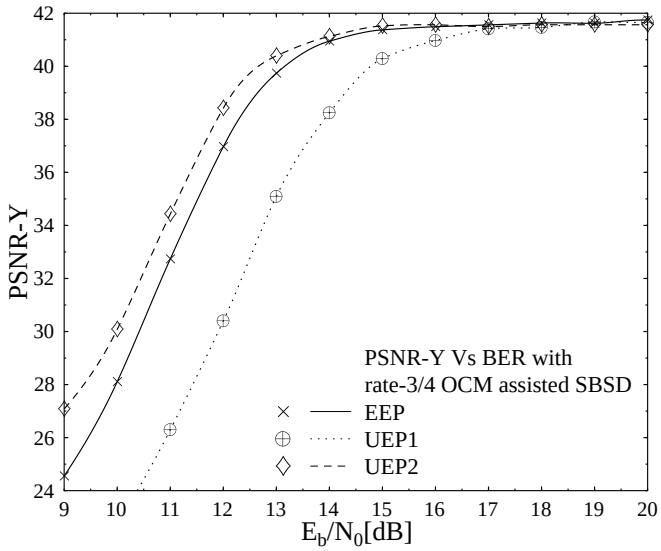

Figure 5.27: PSNR-Y versus E_b/N_0 performance of the various error protection schemes of Figure 5.11 using the parameters of Table 5.14.

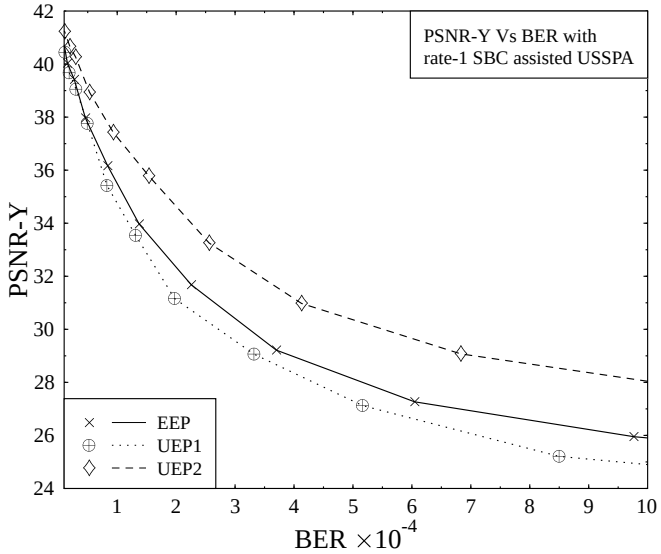

Figure 5.28: PSNR-Y versus BER performance of the various error protection schemes of Figure 5.11 using the parameters of Table 5.14.

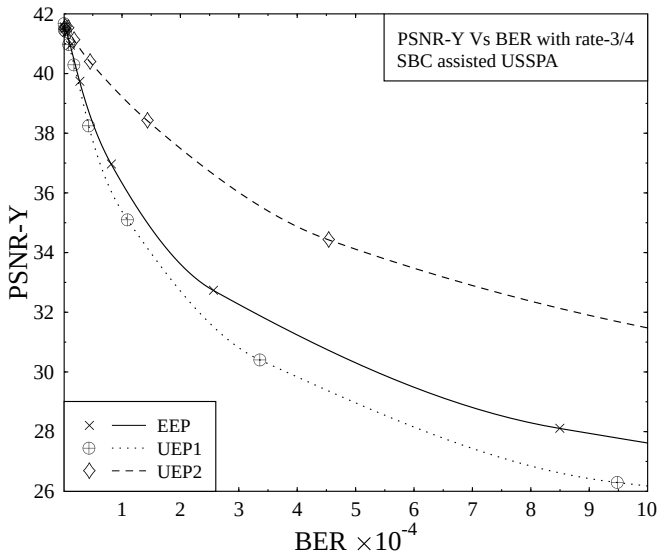

Figure 5.29: PSNR-Y versus BER performance of the various error protection schemes of Figure 5.11 using the parameters of Table 5.14.

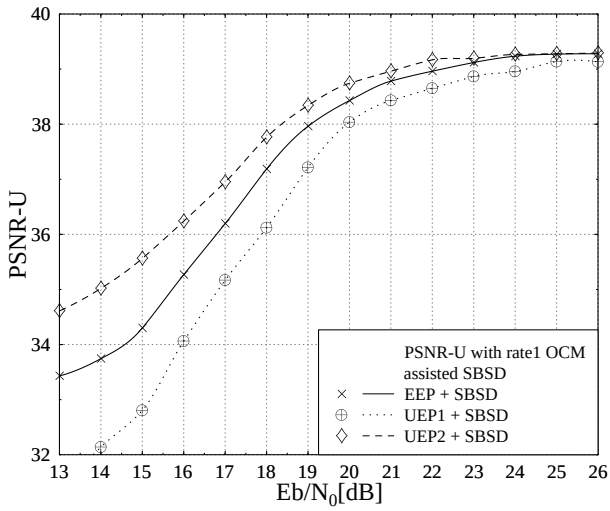

Figure 5.30: PSNR-U versus E_b/N_0 performance of the various error protection schemes of Figure 5.11 using the parameters of Table 5.14.

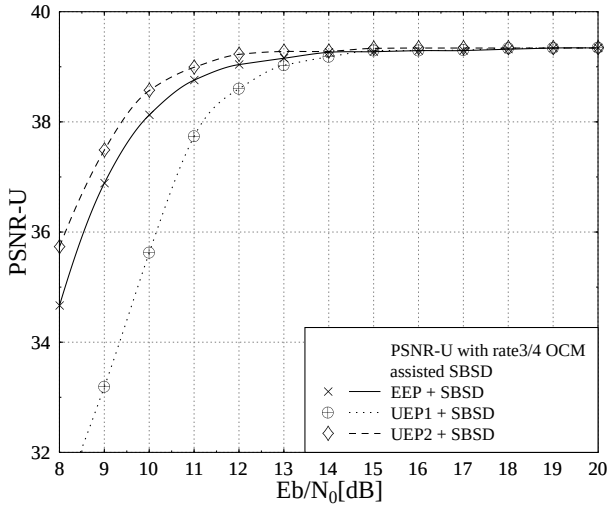

Figure 5.31: PSNR-U versus E_b/N_0 performance of the various error protection schemes of Figure 5.11 using the parameters of Table 5.14.

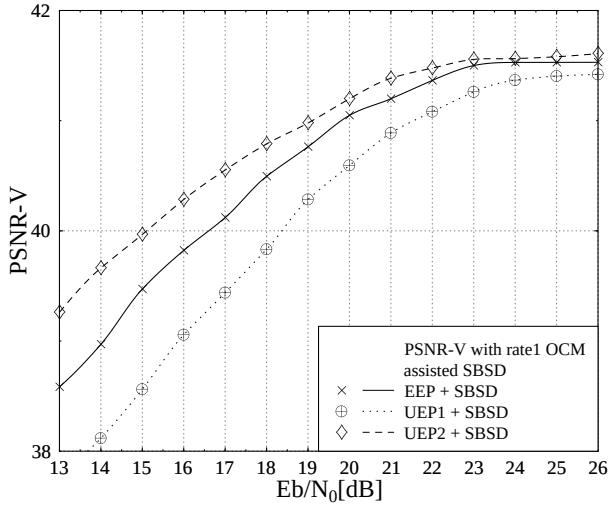

Figure 5.32: PSNR-V versus E_b/N_0 performance of the various error protection schemes of Figure 5.11 using the parameters of Table 5.14.

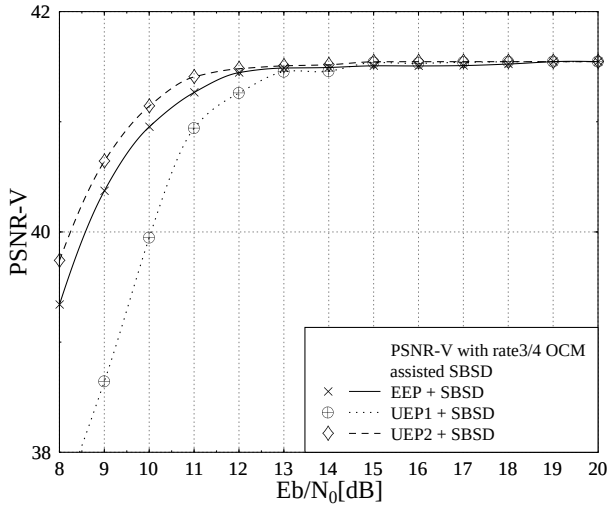

Figure 5.33: PSNR-V versus E_b/N_0 performance of the various error protection schemes of Figure 5.11 using the parameters of Table 5.14.

5.6 Iteratively Detected H.264 Wireless Video Telephony Using Three-Stage System Design

In this section, we advocate a three-stage serially concatenated scheme, for near-capacity operation. Contrary to the two-stage system design consisted of one iterative loop, the three-stage system employs two iterations between the inner and the intermediate codes decoder, which we refer to as inner iterations, as well as between the outer codes decoder and the intermediate codes decoder, which are referred to as outer iterations. A particular combination of inner iterations followed by outer iterations is referred to as one System iteration.

5.6.1 Three-Stage System Design Example

In order to characterise the advantage of the Three-Stage system design for the performance improvement of joint source-channel decoding in videophone type video coding and transmission scenario, we present our design example. Our proposed design example consists of inner sphere peaking modulator, intermediate rate-1 precoder and outer short block code. Therefore, our focus is to analyse the effect on performance improvement of the traditional two-stage turbo-coding, if an intermediate unity rate precoder is invoked between the inner and outer iterative components, which result in a three-stage serially concatenated system.

5.6.2 Three-Stage System Overview

The schematic of the proposed system is shown in Figure 5.34, where we employ rate-$\frac{1}{3}$ SBC coding, provided by partitioning the k^{th} video frame into N source codec symbols, where each symbol $v_{n,k}$ consists of M source coded bits $v_{n,k}(m)$, $m = 1, \cdots M$ although this is not shown in Figure 5.34. In our case $M =$ 2-bit input symbols are encoded by the rate-$\frac{1}{3}$ SBCs of Figure 5.34, resulting $M' = 6$ SBC coded symbols. The SBC encoded bits s are then interleaved by a random bit interleaver Π_{out} and then the interleaved bits s' are encoded by the URC. The URC-encoded bits r are interleaved by the second random bit interleaver Π_{in} of Figure 5.34 into r' and passed to the SP modulator. The benefit of the SP modulator is that it allows us to jointly consider the space-time symbols of the DSTS scheme's two antennas, while maximising the Euclidean distance of the resultant symbols. The SP modulator maps B number of coded bits $b = b_0, ...b_{B-1} \in 0, 1$ to a SP symbol $v \in V$, so that we have $v = map_{sp}(b)$, where $B = log_2(L)$, and L represents the set of legitimate SP constellation points, as detailed in [251]. More explicitly, we used $B = log_2(16) = 4$ channel coded bits per SP symbol. The resultant set of SP symbols are transmitted with the aid of DSTS within two time slots using two transmit antennas. In this study, we consider transmission over a temporally correlated narrowband Rayleigh fading channel, associated with a normalised Doppler frequency of $f_D = f_d T_s = 0.01$, where f_d is Doppler frequency and T_s is the symbol duration. As detailed in Section 5.3.1 the residual redundancy in the source coded bitstream is exploited by the USSPA decoder as *a priori* information for computing the extrinsic LLRs [176]. The extrinsic LLR computation is provided in detail in Section 4.3.4 and can be found in [195, 220], and is briefly reviewed in Section 5.3.1.

The schemes considered in this section differ in the choice of the outer SBC codec. Specifically, we consid-

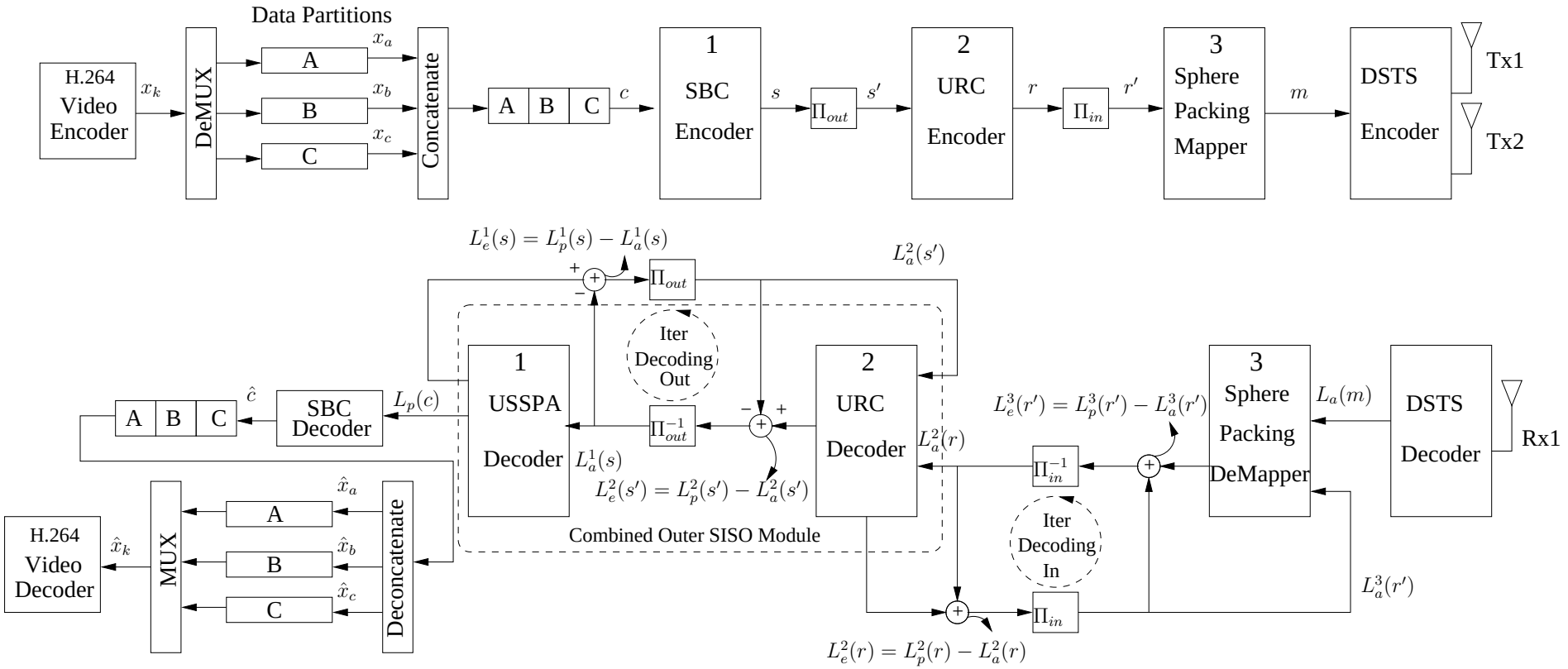

Figure 5.34: The Proposed three-stage ISCD System Model. All system parameters were summarized in Table 5.18. In contrast to Figure 4.10 of Section 4.5, in this system in addition to iterative decoding exchanging extrinsic information between the URC-decoder and USSPA source decoder an extra iteration is performed between the SP-demapper and URC decoder.

ered EXIT-chart optimised SBCs and an equivalent-rate arbitrary SBC based benchmarker having a minimum Hamming distance of $d_{H,min} = 3$ and 1 respectively, as given in Table 5.16. We refer to these two schemes as the DSTS-SP-URC-SBC and DSTS-SP-URC-SBC* arrangements.

Table 5.16: SBCs with corresponding symbols and $d_{H,min}$, obtained from Table 4.8 of Section4.5.4.

SBC Type	Symbols in Decimal	$d_{H,min}$
Arbitrary Rate-$\frac{1}{3}$ SBC$^\star_{[2,\ 6]}$	{0,16,32,48}	1
Optimised Rate-$\frac{1}{3}$ SBC$_{[2,\ 6]}$	{0,22,41,63}	3

5.6.3 Three-Stage Iterative Decoding

At the receiver, the APP SISO decoder of the URC-decoder and the USSPA decoder iteratively exchange extrinsic information represented in the form of LLRs, to assist each other in approaching the point of perfect convergence at (1,1) of the EXIT-chart, as shown in Figure 5.34. The variable $L(.)$ represents the respective bit-LLRs, where the LLRs of the corresponding decoder in our three-stage system design are differentiated by the subscript 1, indicating the outer SBC decoder. By contrast, subscript 2 is used to represent our intermediate URC decoder, while 3 corresponds to the inner SP decoder. Additionally, the specific type of the LLRs is indicated by the subscript *a*, *p* and *e*, corresponding to *a priori*, *a posteriori* and *extrinsic* information, respectively.

5.6.4 Inner Iterations

The received complex-valued symbols corresponding to each $B = 4$ number of URC-coded bits per DSTS-SP symbol are demapped to their LLR [252] representations. The extrinsic LLR values $L_e^3(r)$ are generated by subtracting the *a priori* information $L_a^3(r)$ provided by the URC decoder from the *a posteriori* LLR values $L_p^3(r)$ at the output of the SP-demapper, as shown in Figure 5.34. The LLRs $L_p^3(r)$ are deinterleaved by the softbit interleaver Π_{in} of Figure 5.34 and are passed to the URC-decoder in order to produce the *a posteriori* LLR values $L_p^2(r)$ using the MAP algorithm [168] for the URC-encoded bits. The extrinsic LLR values $L_e^2(r)$ of the URC-encoded bits *r* are obtained by subtracting the *a priori* LLRs $L_a^2(r)$ input to the URC-encoder from the URC generated *a posteriori* LLRs $L_p^2(r)$. Following interleaving the resultant extrinsic LLRs $L_e^3(r)$ are fed back to the SP-demapper as the *a priori* information $L_a^3(r)$. This *a priori* information is exploited by the SP demapper for the sake of providing improved extrinsic information for the URC decoder in the successive iterations.

5.6.5 Outer Iterations

The outer iterations are comprised of exchanging extrinsic information between the USSPA and URC decoder of Figure 5.34. First, the extrinsic LLRs $L_e^2(s)$ produced by subtracting the *a priori* information $L_a^2(s)$ from the *a posteriori* LLRs $L_p^2(s)$ generated by the URC decoder are deinterleaved using the soft-bit interleaver Π_{out} of

Figure 5.34 and passed to the USSPA decoder as *a priori* LLRs $L_a^1(s)$. The USSPA computes the *a posteriori* LLRs $L_p^1(s)$ and subtracts the input *a priori* LLRs $L_a^1(s)$ from it in order to produce the extrinsic LLRs $L_e^1(s)$. These extrinsic LLRs are interleaved and fed back as *a priori* information to the URC decoder for the sake of generating improved extrinsic LLRs in the subsequent iterations.

In our proposed system design we define, a system iteration, I_{system} as that composed of one inner iteration followed by two outer iterations.

5.6.6 EXIT Chart Analysis

EXIT charts [255] are used widely for the analysis of iterative systems by providing an insight into the convergence behaviour of the system based on the exchange of MI amongst the constituent receiver components.

As portrayed in Figure 5.34, the URC decoder receives its input from and provides output to both the SP and the USSPA. More explicitely, for the symbol x having the *a priori* LLR value $L_{\dot{a}}(x)$, the MI is denoted by $I_{\dot{,}A}(x)$, while the MI between the extrinsic LLR $L_{\dot{e}}(x)$ and the corresponding symbol x is denoted by $I_{\dot{,}E}(x)$. The MI associated with one of the three constituent decoders is differentiated by replacing the subscript (.) with the corresponding subscript 1, 2 and 3 for the SBC, URC and SP-mapper, respectively, as shown in Figure 5.34. Thus the URC decoder is provided with two *a priori* inputs. The first *a priori* input $L_a^2(r)$ corresponding to the coded bits r is provided by the SP demapper. The second *a priori* input $L_a^2(s')$ corresponding to the data bits s' is provided by the USSPA and fed back to the URC-decoder. Similarly, the URC decoder generates two extrinsic outputs $L_e^2(r)$ and $L_e^2(s')$, representing the data bits r and s', respectively, where the corresponding EXIT functions of the URC decoder are $F_r[L_a^2(r), L_a^2(s')]$ and $F_{s'}[L_a^2(r), L_a^2(s')]$, respectively. However, the USSPA and the SP demapper only receives input from and provides output for the URC decoder. Therefore, their corresponding EXIT functions are $F_s[L_a^1(s)]$ and $F_{r'}[L_a^3(r'), E_b/N_0]$, respectively.

According to the EXIT chart analysis of iterative decoding, an infinitesimally low BER can only be achieved by an iterative receiver, if there exists an open EXIT tunnel between the two constituent decoders, so that during the process of iterative decoding, the outer decoder results in the highest possible extrinsic information $I_E(outer) = 1$, for a given input *a priori* information as shown in Figure 5.36.

For the EXIT chart analysis of our proposed system the intermediate URC decoder and the USSPA are viewed as a single combined outer SISO module. The EXIT chart of the proposed benchmarker DSTS-SP-URC-SBC* scheme is shown in Figure 5.35 along with the EXIT curves of the SP demapper recorded for the E_b/N_0 values of 8 to 13 dB. As seen from Figure 5.35, the EXIT cure of the combined outer SISO module constituted by the DSTS-SP-URC-SBC* scheme cannot reach the (1,1) point of perfect convergence in the EXIT chart, since it intersects with the EXIT curve of the inner SP demapper, which implies that an infinitesimally low BER cannot be achieved. Additionally, the outer EXIT curve of the combined SISO module recorded for the DSTS-SP-URC-SBC scheme is shown in Figure 5.36 along with the EXIT curves of the SP demapper for various E_b/N_0 values. Figure 5.36 shows that the joint EXIT curve of the USSPA and URC decoder in the DSTS-SP-URC-SBC arrangement reaches the (1,1) point of the EXIT chart. Figure 5.35 and 5.36 also provide the Monte-carlo simulation based decoding trajectories of the proposed system at the E_b/N_0 values considered. These trajectories were recorded by acquiring the MI at the input and output of

both the inner SP demapper and the joint outer SISO module during the bit-by-bit Monte-Carlo simulation of the iterative decoding algorithm. Observe from the decoding trajectories of Figure 5.36 that for E_b/N_0 values higher than 8 dB the DSTS-SP-URC-SBC scheme becomes capable of achieve the highest possible extrinsic information of $I_E(outer) = 1$ during the iterative decoding process. However the DSTS-SP-URC-SBC* is unable to achieve this goal due to the intersection of the inner and outer EXIT curves.

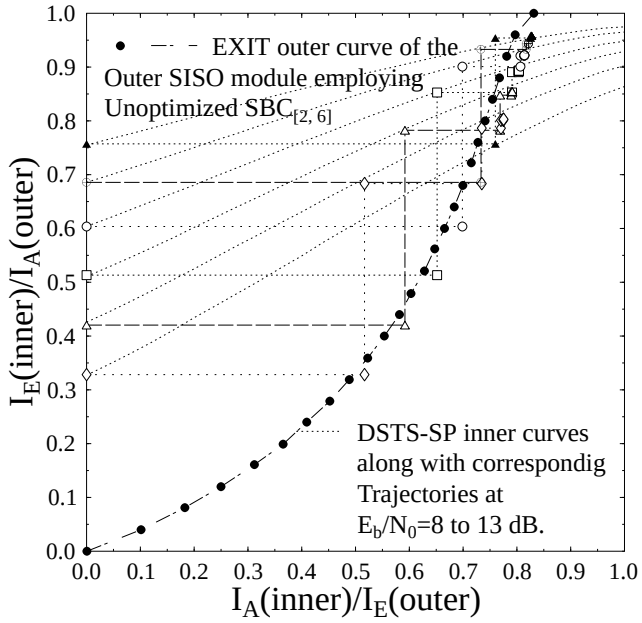

Figure 5.35: The EXIT chart and simulated decoding trajectories of the DSTS-SP-SBC* scheme of Figure 5.34 using the parameters of Table 5.18 at E_b/N_0 = 8 to 13 dB. The outer EXIT curve employing the SBC$^*_{[2,6]}$ scheme of Table 5.16 having $d_{H,min} = 1$, is unable to reach to the $(1,1)$ EXIT-chart point.

5.6.7 System Performance Results

In this section, we present our overall performance results for the proposed system model. For the performance analyses of our proposed three-stage system model, we employed the H.264/AVC video source codec with *Akiyo* video test sequence of 45 frames, as detailed in Section 3.5. We consider the SP modulation scheme [251] of Section 5.2.1 associated with $L = 16$ sphere-packing modulated symbols, while employing Anti-Gray Mapping (AGM)[2] for source bits-to-SP symbol mapping. Our system design consisted a two-antenna-aided DSTS and a single receiver antenna arrangement. The performance of the system was evaluated, while considering various combinations of the system iterations I_{system} and of the iterations within the outer joint USSPA-URC system module I_{out}. For the sake of increasing the confidence in our results, we repeated each 45-frame experiment 160 times and averaged the generated results.

[2] Any bit-to-symbol mapping, which is different from the gray-mapping is referred to as an AGM. The best AGM has to be found also by EXIT-chart optimisation.

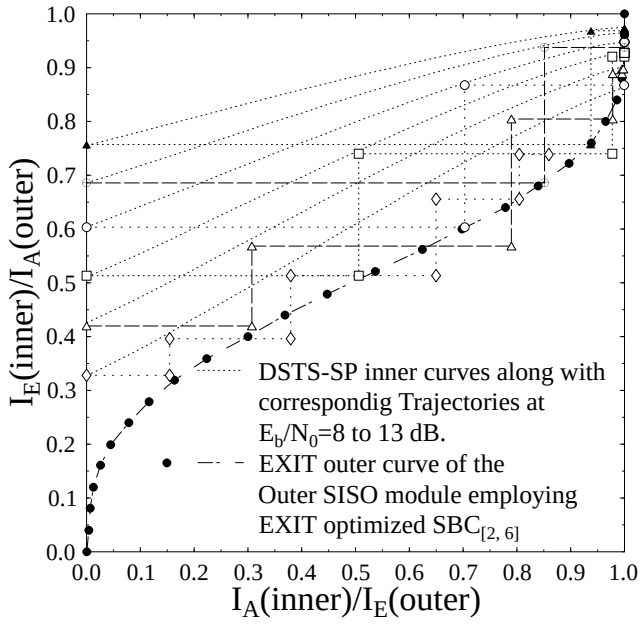

Figure 5.36: The EXIT chart and simulated decoding trajectories of the DSTS-SP-SBC scheme of Figure 5.34 using the parameters of Table 5.18 at E_b/N_0 = 8 to 13 dB. As expected, the outer decoder's EXIT curve employing the $\text{SBC}_{[2,6]}$ scheme of Table 5.16 having $d_{H,min} = 3$ approaches to $(1,1)$ EXIT point.

The BER performance of the error protection scheme employed is shown in Figure 5.37. It can be observed from Figure 5.37 as expected, that the DSTS-SP-URC-SBC scheme using $I_{system} = 5$ system iterations results in the best BER performance, when compared to $I_{system} = 4$ and 3 for the same error protection scheme. Additionally, it can be seen that the the DSTS-SP-URC-SBC* scheme results in the worst BER performance due to its inability to reach the (1,1) point of perfect convergence.

Furthermore, the $PSNR$ versus E_b/N_0 curve of the proposed error protection scheme is portrayed in Figure 5.38. It may be observed in Figure 5.38 that the DSTS-SP-URC-SBC scheme employing rate-$\frac{1}{3}$ SBCs having $d_{H,min} = 3$ and $I_{system=5}$ system iterations results in the best PSNR performance across the entire E_b/N_0 region considered. It is also observed in Figure 5.38 that when performing iterative decoding, while employing an arbitrary rate-$\frac{1}{3}$ SBC results in the worst $PSNR$ performance at the same overall code rate of $\frac{1}{3}$, as given in Table 5.18(a). Quantitatively, when using the DSTS-SP-URC-SBC scheme of Table 5.18(a), an E_b/N_0 gain of upto 22 dB may be achieved relative to the DSTS-SP-URC-SBC* scheme at the PSNR degradation point of 2 dB, as shown in Figure 5.38.

Finally, the subjective video quality of the error protection schemes employed is presented in Figure 5.39. The video frames presented in Figure 5.39 were obtained by repeated retransmission of the received video sequence using the same system with $I_{system} = 5$ system iterations 30 times, in order to have a pertinent subjective video quality comparison. Observe from Figure 5.39 that an unimpaired video quality is attained by the DSTS-SP-URC-SBC scheme at an E_b/N_0 value of 10 dB. However, video impairment persist for the

DSTS-SP-URC-SBC* scheme even at the high E_b/N_0 values of 28.5 dB, 29 dB, 29.5 dB and 30 dB, as shown in Figure 5.39. The major findings of the consider error protection schemes at PSNR degradation points of 1dB and 2dB are summarisedin Table 5.17

Table 5.17: Performance Summary Table for the system of Figure 5.34

Schematic	Parameters	Iterations	Video rate	FEC Code rate	Highest PSNR value	E_b/N_0 at 1dB PSNR degradation	E_b/N_0 at 2dB PSNR degradation
Figure 5.34	Table 5.18(a)	10	64 *kbps*	0.333	41.7582	$I_t(3)$ = 10.0 dB $I_t(4)$ = 9.5 dB $I_t(5)$ = 8.6 dB	$I_t(3)$ = 9.6 dB $I_t(4)$ = 9.0 dB $I_t(5)$ = 8.5 dB

Error Protection Scheme	Code Rate		
	SBC type	Intermediate code	Overall
DSTS-SP-URC-SBC*	Rate-$\frac{1}{3}$ $SBC^\star_{[2,\,6]}$	Rate-1 Precoder	1/3
DSTS-SP-URC-SBC	Rate-$\frac{1}{3}$ $SBC_{[2,\,6]}$	Rate-1 Precoder	1/3

(a) Code rates for different Error Protection schemes.

System Parameters	**Value**	**System Parameters**	**Value**
Source Coding	H.264/AVC	Video Bit Rate (Kbps)	64
Source Codec	JM 12.1	Video Frame Rate (fps)	15
Video Test Sequence	*Akiyo*	Channel Coded Rate (Kbps)	192
No of Frames	45	Baud-rate (Kbps)	96
Frame Format	QCIF	Intermediate Code	URC
No of 'P' frames between	44	Over-all Code Rate	1/3
two 'I' frames		Code Memory	1
Time Period of 'I' frame (sec)	3	Generator Polynomials	
Use of 1/4-Pixel Motion Estimation	Yes	URC (G_1, G_2)	$(3,2)_8$
Intra-frame MB update/frame	3		
per QCIF frame		Modulation Scheme	SP
Use of 'B' Pictures	No	Number of Transmitters, N_t	2
Use of FMO	No	Number of Receivers, N_r	1
No of Frames Used for	1-Frame	Channel	Correlated Rayleigh Fading
Inter-Frame Motion Search		Normalised Doppler Frequency	0.01
No of Video Slices/frame	9	Interleaver Length	$\approx (64000/15)$
No of Video MBs/Slice	11	No System Iterations I_t	5, 4, 3

(b) Systems parameters used in the schematic of Figure 5.34

Table 5.18: Code rates and systems parameters used in the schematic of Figure 5.34

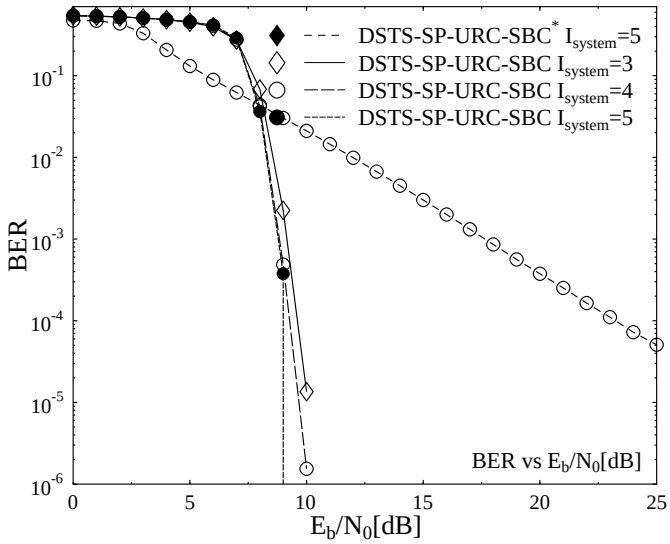

Figure 5.37: BER versus $E_b/N0_0$ performance of the various error protection schemes of Figure 5.34 using the parameters of Table 5.18.

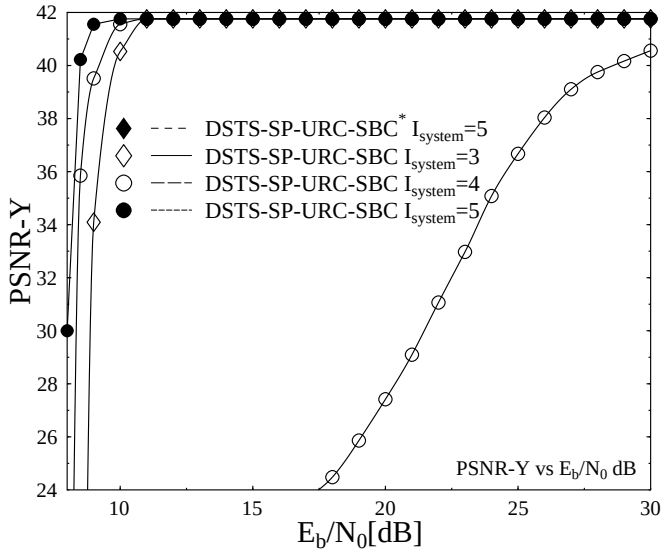

Figure 5.38: PSNR-Y versus $E_b/N0_0$ performance of the various error protection schemes of Figure 5.34 using the parameters of Table 5.18.

Figure 5.39: Subjective video quality of the 45^{th} *"Akiyo"* video sequence frame using (from top) DSTS-SP-RSC-SBC and DSTS-SP-RSC-SBC* schemes of Figure 5.34 using the parameters of Table 5.18 at E_b/N_0 values of (from left) 8.5 dB, 9 dB, 9.5 dB and 10 dB for DSTS-SP-RSC-SBC and 28.5 dB, 29 dB, 29.5 dB and 30 dB for DSTS-SP-RSC-SBC*.

5.7 Chapter Conclusions

This chapter has the following findings,

- In Section 5.2 we evaluated the attainable performance of the UEP schemes of Figure 5.2 proposed in Section 3.10, which employed a sophisticated low-complexity SP modulation aided MIMO DSTS scheme, relying on two transmit and one receive antenna, dispensing with CIR estimation. Explicitly an E_b/N_0 gain of 4 dB is attained in Figure 5.6 using UEP1 with reference to UEP2, while an E_b/N_0 gain of 2 dB is achieved by the latter with reference to the EEP scheme at the PSNR degradation point of 2 dB. Furthermore, with reference to the system setup of Section 3.10, which utilised an identical coding scheme, but dispensed with SP modulation aided DSTS transmission, in Figure 5.6 an E_b/N_0 gain of about 10 dB is attained relative to Figure 3.26 of Section 3.10.

- In Section 5.3 we contrived the novel USSPA decoding scheme of Figure 5.7 for the transmission of an H.264/AVC coded video bit-stream, in order to exploit the residual redundancy, which inherently remains in the coded bit-stream after source and channel encoding and for the sake of reliable extrinsic informantion generation. The resultant USSPA assisted UEP aided bit-stream was transmitted using the DSTS aided SP modulation scheme of Figure 5.11 for attaining a diversity gain without the need for any high-complexity MIMO channel estimation. Quantitatively, an E_b/N_0 gain of 3 dB is attained using UEP1 with reference to UEP2, while an E_b/N_0 gain of 2 dB is achieved by the latter with reference to the EEP scheme, at the PSNR degradation point of 2 dB. Furthermore, an E_b/N_0 gain of about 10 dB was achieved compared to the PSNR-Y vs E_b/N_0 curves of Figure 5.10 in Section 5.4.3 to that of Figure 4.9 in Section 4.4.1. Both schemes consisted of similar type of error protection schemes, but the former dispensed with SP modulation aided DSTS transmission.

- Additionally, in Section 5.4 we incorporated the novel EOSBC arrangement of Figure 4.26 described in Section 5.4.1, by appropriately apportioning the total available bit-rate budget between the source and channel codecs, which results in an improved performance, when ISCD was employed. The SBC coded stream after RSC encoding was transmitted using the SP modulation aided DSTS transceiver shown in Figure 5.11. Explicitly, an E_b/N_0 gain of 3 dB is attained in Figure 5.14 using the UEP1 with reference to UEP2, while an E_b/N_0 gain of 1 dB is achieved by the latter with reference to the EEP scheme of Table 5.14(a) scheme, at the PSNR degradation point of 2 dB. Furthermore, we compared the PSNR-Y vs E_b/N_0 curves obtained in Figure 5.14 of Section 5.4.4 to that of Figure 3.26 of Section 3.10, both of which consisted of similar error protection schemes, but the former dispensed with SP modulation aided DSTS transmission. It is observed from Figure 5.10 that an E_b/N_0 gain of about 10 dB can be achieved with reference to Figure 4.9 of Section 4.4.

- In Section 5.5, the performance of SBCs was analysed using the system setup of Figure 5.11. Additionally, in Section 5.5.3 the performance of the UEP SBC schemes employed was analysed with the aid of EXIT charts. Explicitly, using the code rates of Table 5.14(a) for the different error protection schemes, an E_b/N_0 gain of about 12 dB was attained in Figure 5.27 using SBCs relative to Figure 5.26 dispensing with SBCs, at the PSNR degradation point of 1 dB.

- Furthermore, in Figure 5.34 of Section 5.6.2 a three-stage system design was presented, which was constituted by the serially concatenated and iteratively decoded EOSBCs and the precoded DSTS aided multi-dimensional SP modulation scheme of Figure 5.3 designed for near-capacity joint source and channel coding. It was detailed in Section 5.6.7 that the employment of EXIT optimised SBCs, which deliberately imposed further artificial channel-coding redundancy on the source coded bit stream provided significant improvements in terms of the $PSNR$ versus E_b/N_0 performance, when compared to the benchmarker scheme of Figure 5.34 employing equivalent-rate SBCs, which were not optimised using EXIT-charts for achieving the best possible iterative convergence behaviour. Additionally, the convergence behaviour of the proposed system was analysed with the aid of EXIT charts in Section 5.6.6. It was pointed out in the context of Figure 5.35 and 5.36 that although the joint EXIT function of the non-optimised inner SBC encoder and unity rate precoder cannot reach the point of perfect convergence at (1,1), the joint EXIT function of the optimised inner SBC encoder in conjunction with rate-1 precoder can. The H.264/ SBC/ URC/DSTS-SP based three-stage design example of Figure 5.34 and using SBCs having $d_{H,min} = 3$ exhibited an E_b/N_0 gain of 22 dB in Figure 5.38 at the PSNR degradation point of 2 dB relative to the identical-rate benchmarker employing equivalent rate SBCs having $d_{H,min} = 1$, as detailed in Section 5.6.7.

5.8 Chapter Summary

In Section 5.2 we evaluated the performance of the data-partitioned H.264 coded video transmission system of Figure 5.3 using different UEP schemes. RSC codes were used for the UEP of the video stream, while employing a SP modulation aided DSTS transmission scheme. When using UEP, the perceptually more important bits were provided with more strong protection relative to less important video bits. Explicitly an E_b/N_0 gain of 4 dB is attained in Figure 5.6 using UEP1 with reference to UEP2, while an E_b/N_0 gain of 2 dB is achieved with reference to EEP scheme, at the PSNR degradation point of 2 dB.

Additionally, in Section 5.3 the USSPA was incorporated which has the additional capability of improving error correction capability, hence enhancing the subjective video quality by exploiting the residual redundancy that remains in the coded stream after encoding. The resultant USSPA coded bit-stream is transmitted using a DSTS aided SP modulation scheme for attaining a diversity gain without the need for any high-complexity MIMO channel estimation. Quantitatively, an E_b/N_0 gain of 3 dB is attained using UEP1 with reference to UEP2, while an E_b/N_0 gain of 2 dB is achieved with reference to EEP scheme, at the PSNR degradation point of 2 dB.

In Section 5.4 an iterative detection aided combination of RSC and a SBC was used to improve the overall BER performance, which enhanced the objective video quality expressed in terms of PSNR. The SBCs of Section 5.4 result in introducing intentional redundancy in the source coded bit-stream, which assists the outer decoder's EXIT curve to reach to the $(1,1)$ point of the EXIT chart. The effect of different error protection schemes of Table 5.10(a) on the attainable system performance was demonstrated in Figure 5.14, while keeping the overall bit-rate budget constant for the transmission of DP H.264 source coded video over correlated narrowband Rayleigh fading channels. Quantitatively, an E_b/N_0 gain of 3 dB is attained in Figure 5.14 using UEP1

with reference to UEP2, while an E_b/N_0 gain of 1 dB is achieved with reference to EEP scheme, at the PSNR degradation point of 2 dB. Furthermore we compared the PSNR-Y vs E_b/N_0 of obtained in Section 5.4.4 to that of Section 3.10, consisted of similar type of error protection schemes, but dispensing with SP modulation aided DSTS transmission. Its observed from Figure 5.10 that an E_b/N_0 gain of about 10 dB can be achieved with reference to Figure 4.9 of Section 4.4.

Additionally, in Section 5.5 EXIT charts were utilised in ordered to analyse the effect of the SBC coding rate on the achievable performance of the UEP iterative JSCD strategies of Table 5.14(a). Explicitly using the code rates of Table 5.14(a) for different Error Protection schemes an E_b/N_0 gain of about 12 dB was attained in Figure 5.27 using SBC relative to Figure 5.26 dispensing with SBC, at the PSNR degradation point of 1 dB.

Furthermore, in Figure 5.34 of Section 5.6.2 we considered jointly optimised three-stage source and channel decoding, while employing serially concatenated and iteratively decoded SBCs combined with a URC and multi-dimensional SP modulation. The resultant coded signal was transmitted over non-coherently detected DSTS MIMOs transceiver designed for near-capacity JSCD, as shown in Figure 5.34. The performance of the system was evaluated in Figure 5.34 by considering interactive video telephony using the H.264/AVC source codec. The output bit-stream generated by the state-of-the-art H.264/AVC video codec typically contain limited natural residual redundancy. Therefore, to improve the error robustness of ISCD in Figure 5.34, SBCs were incorporated in order to impose artificial redundancy on the source coded parameters. The natural residual redundancy of source coding and the artificial redundancy imposed by SBCs was iteratively exploited in the architecture of Figure 5.34 in a turbo process to improve the overall BER in Figure 5.37 and the objective video quality performance of Figure 5.38 quantified in terms of the PSNR. In Section 5.6.6 the convergence behaviour of the MIMO transceiver advocated was investigated with the aid of EXIT charts. Explicitly, the proposed system of Figure 5.34 exhibits an E_b/N_0 gain of about 22 dB in Figure 5.38 at the PSNR degradation point of 2 dB in comparison to the benchmarker scheme of Figure 5.38 carrying out DSTS aided SP-demodulation as well as iterative source and channel decoding, when using $I_{system} = 5$ system iterations, while communicating over correlated narrow-band Rayleigh fading channels, as detailed in Section 5.6.7.

Chapter 6

Conclusions and Future Research

In this concluding chapter, we will first provide our overall summary and conclusions in Section 6.1. The major findings of the thesis are summarised in Table 6.1 and Table 6.2. Then a range of topics concerning potential future research ideas will be presented in Section 6.2.

6.1 Thesis Summary and Conclusions

In this thesis, we have provided detailed transceiver designs employing novel channel coding schemes designed for enhancing the achievable performance of H.264 video transmission systems. After a brief historical perspective on the H.264 video codec in Section 2.2, we have commenced with the characterisation of an UEP Recursive Systematic Convolutional (RSC) coded video transmission system in Section 3.10, as the first step towards more sophisticated source-channel coded video transmission systems. Our best performing near-capacity design was presented in Section 5.6.2.

The findings of this thesis are summarised below,

- **Chapter 1:**
  Chapter 1 provided a basic introduction to the different commonly used concepts of video coding. As detailed in Section 1.2, the video signal is generated by the video camera in analogue format by scanning a two-dimensional scene. Each video clip is comprised of a number of scanned scenes constituting a frame. The number of lines per frame has a direct relation to its bandwidth and resolution. Additionally, the number of frames per unit of time is another factor affecting the temporal resolution of the video, which should always be kept above some specified value to avoid video flicker. Additionally, in order to facilitate video storage and compression, the analogue video is converted into digital format, as described in Section 1.3, using a sequence of operations, such as filtering, sampling and quantisation. Additionally, in Section 1.6 we have detailed the various image formats obtained after the digitisation process, such as the Source Input Format (SIF), Common Intermediate Format (CIF), Quarter Source Input Format (QSIF), Quarter Common Intermediate Format (QCIF) etc.

From the subjective and objective video quality evaluation methods presented in Section 1.7 it was noted that the PSNR may be unable to adequately characterise the subjective human perception, since the humans exhibit different sensitivity to distortions at different parts of the image. For example, a small block of severely distorted pixels in the subjectively most eye catching areas of the frame, such as the human face of a speaker may attract the viewer's attention and hence it may be perceived as a large distortion in the frame, although it may hardly affect the objective PSNR quality metric owing to its limited size.

After digitisation, the video sequence may be compressed by following the sequence of operations described in Section 1.8, including motion compensation, intra-frame coding, inter-frame coding, transform coding, quantisation and entropy coding. The concept of entropy coding was detailed in Section 1.8.3, which is used to achieve further compression by reducing the redundancy among the symbols, by employing various VLC techniques. The idea of variable length coding is to assign a lower number of bits to high-probability source symbols, while long code words are assigned to symbols associated with a low probability. Furthermore, the concept of Motion Compensated Temporal Interpolation (MCTI) is presented in Section 1.8.5.4 and from the results obtained it may be observed that despite the high compression efficiency achieved by the employment of MCTI, hardly any perceptual difference is observed among the interpolated frame of Figure 1.17 in comparison to the original frame presented in Figure 1.11. Finally, the chapter was concluded by providing information related to the building blocks of video coding systems in Section 1.8.5.

- **Chapter 2:**

  It was stated in Section 2.1 that the focus of the H.264 AVC standardisation activity was to design an efficient and network-friendly video codec capable of supporting a variety of applications. The design of H.264/AVC was defined in form of two layers known as the Video Coding Layer (VCL) and Network Abstraction Layer (NAL) [76], as stated in Section 2.2. The core compression technique of H.264/AVC is based on the VCL, while the NAL formats the stream output by the VCL, generating a specific format suitable for various transport layers in multi-stream multiplexing environments, in order to be as independent of the transport network as possible. It may be concluded from Section 2.3 that just like the standard video codecs, such as H.261, H.263, H.264/AVC, MPEG-1, 2 and 4, the H.264 scheme is also based on the hybrid video coding concept, consisting of a combination of video coding techniques, such as motion compensation and transform coding.

  Additionally, Section 2.4 provided further details about the different features contained in the H.264 coded stream, such as its network-related adaptation capability, its flexible bit stream structure and error resilience. In order to improve its motion compensation capability and to minimise the appearance of blocking artifacts in highly motion-active areas of the frame, the H.264/AVC codec supports the INTRA_4x4 intra-frame prediction mode used for the prediction of the luminance component Y. In the INTRA_4x4 mode a macro-block of 16x16 picture elements is partitioned into sixteen 4x4 sub-blocks and motion prediction is applied separately for each 4x4 sub-block, as described in Section 2.5.

  The advantages of using the small transform block sizes described in Section 2.5.3 were;

  – The smaller block size of 4x4 instead of 8x8 pixels facilitates better motion estimation for small

objects of both in the inter- and intra-frame coded mode. As a result, the residual signal has a lower spatial correlation and results in an efficient reduction of correlation.

- The smaller transform size results in reduced artifacts around the block edges, such as mosquito noise or ringing artifacts.
- The smaller transform size also results in reduced computational complexity owing to its smaller processing word-length.

Furthermore, in comparison to the previous standards the H.264 codec provides significant improvements, including;

- Enhancement in motion compensation by using high-accuracy motion vectors and multiple reference frames, as described in Section 2.5.2.
- Employment of an integer transform, instead of the classic DCT, as presented in Section 2.5.3.
- Adaptive in-loop de-blocking filter which was detailed in Section 2.5.4.
- Enhanced adaptive entropy coding, as highlighted in Section 2.6.

- **Chapter 3:**

In Section 3.6 we elaborated on the hierarchical structure of the H.264 coded video stream and its bit-stream syntax. As depicted in Figure 3.12, the hierarchical structure of the H.264 video stream consists GOPs, which are composed of a number of frames, while the frames are sub-divided into slices. The slices are composed macroblocks, which in turn are composed of four 8x8 pixel blocks. Furthermore, it was observed from the bit-stream syntax of the H.264 video codec pictorally presented in Figure 3.13 of Section 3.7 that various coding elements, such as the Start Code Prefix, Information Bytes, Sequence Parameter Sets and Picture Parameter Sets were encapsulated in the NAL unit. In Section 3.8 we observed that the H.264 Data-Partitioning (DP) may generate three bit-strings per slice referred to as partitions, hosting different-sensitivity classes and this provides us with the ability to protect the different sensitivity classes based on their relative importance. Moreover, in Section 3.9 we performed a detailed error sensitivity study for the partitions A, B and C of the H.264/AVC coded video stream. It was observed that in the H.264 video stream, partition A is the most important type of partition, followed by Partition B and then Partition C. Furthermore, in Section 3.10, we proposed UEP H.264/AVC coded video transmissions using RSC codes. Based on the sensitivity study of Section 3.9, the performance of the RSC coded system setup was analysed for the transmission of the DP-aided H.264/AVC coded bit-stream. We have demonstrated in Section 3.10 that by using UEP employing an appropriate bit-rate budget allocation to the different partitions of the H.264/AVC coded video based on their relative importance resulted in useful PSNR improvements, as depicted in Figure 3.26. Explicitly, an E_b/N_0 gain of 4 dB was attained using UEP1 over UEP2, while an E_b/N_0 gain of 3 dB was achieved with reference to the EEP scheme, at the PSNR degradation point of 2 dB.

The performance of a DP H.264 coded video transmission system using the UEP IRCCs of Table 3.9(a) was evaluated in Section 3.11. We considered the various error protection schemes designed for the transmission of H.264/AVC coded video using the serially concatenated turbo transceiver of Figure 3.27, consisting of an EXIT-chart-optimised outer IRCC and rate-1 inner precoder. When using the UEP scheme of

Table 3.9(a), the perceptually more important bits were provided with more strong protection relative to less important bits. The iterative detection aided combination of the IRCC and the rate-1 decoders seen in Figure 3.27 were used to improve the overall BER performance and to enhance the objective video quality expressed in terms of the PSNR. The effect of the different error protection schemes of Table 3.9(a) on the attainable system performance was demonstrated, while keeping the overall bit-rate budget constant for the transmission of DP aided H.264 source coded video for transmission over correlated narrowband Rayleigh fading channels. In Section 3.11.3 we exploited the high design flexibility of IRCCs, which is constituted by a family of different-rate subcodes, while maintaining an excellent iterative decoding convergence performance. Additionally, as a benefit of using different-rate subcodes, IRCCs have the capability of providing UEP for the H.264 coded video stream, as detailed in Section 3.11.3. The EXIT chart matching procedure of Section 3.11.4 was used for the design of our specific IRCC.

Finally, in Section 3.11.4 EXIT charts were used for analysing the attainable system performance of the various IRCC-based error protection schemes of Table 3.9(a). Explicitly, our experimental results of Section 3.11.5 demonstrated that the proposed UEP scheme using IRCCs outperforms its EEP counterpart employing regular convolutional codes by about 0.5 dB at the PSNR degradation point of 1 dB, as shown in Figure 3.33. Moreover, the convergence behaviour of the system of Figure 3.27 was analysed using EXIT charts in Section 3.11.4.

- **Chapter 4:**

  It was observed in Section 4.3 that in ISCD both the source and channel decoder exploit the residual redundancy for error correction and to determine, where the errors occurred and how to correct them. Furthermore, in Section 4.4 we analysed the beneficial effects of soft-bit source decoding on the achievable performance of the iterative video decoding scheme of Figure 4.6. The source video stream was encoded using the sophisticated state-of-the-art H.264 video codec operated at a low target bit-rate of 64 $kbps$. Explicitly, an E_b/N_0 gain of 5 dB was attained in Figure 4.9 using the UEP1 scheme of Table 4.4(a) in comparison to UEP2, while an E_b/N_0 gain of 3 dB was achieved by the latter with reference to the corresponding EEP scheme, at the PSNR degradation point of 2 dB. Additionally, it was observed from the results of Figure 3.26 and 4.9 that the USSPA aided iterative decoder using RSC codes resulted in a modest performance improvement relative to the identical setup of Figure 3.26 dispensing with the iterative USSPA scheme, as described in Section 3.10.1. The reason for this modest performance improvement was the lack of sufficient residual redundancy in the bit-stream, which was encoded using the sophisticated state-of-the-art H.264 video codec operated at a target bit-rate of 64 $kbps$, for example.

  In Section 4.5 we imposed further artifical redundancy on the coded video stream in order to improve the attainable iterative decoding performance. We proposed a family of EOSBCs in Section 4.5.1, which was designed for guaranteed convergence in soft-bit assisted iterative joint source and channel decoding, and hence facilitated improved iterative USSPA operations. The DP H.264 source coded video was employed to evaluate the attainable performance of our system using EOSBC assisted iterative USSPA decoding in conjunction with RSC codes for transmission over correlated narrowband Rayleigh fading channels. The effect of different SBC schemes having diverse minimum Hamming distances ($d_{H,min}$) and code rates on the attainable system performance was characterised, when using iterative USSPA channel decoding,

while keeping the overall bit-rate budget constant by appropriately partitioning the total available bit rate budget between the source and channel codecs. Naturally, our goal was to improve the overall BER performance and to enhance the objective video quality expressed in terms of the PSNR.

Additionally, in order to analyse the effects of different motion activities on the performance of the video coding setup of Figure 4.10, we considered diverse video test sequences, such as the ”Akiyo”, ”News” and the ”Foreman” clips. The test sequences considered were encoded using the H.264/AVC codec, while considering the system parameters of Table 4.9(b). We utilised the system architecture of Figure 4.10, while considering the parameters of the unity-rate scheme from Table 4.9. It can be observed by viewing these video sequences that the ”News” video sequence has a higher motion activity than our original ”Akiyo” video sequence. Similarly, the ”Foreman” video sequence has a higher motion activity than the ”News” video clip. The BER versus E_b/N_0 performance of the video test sequences considered is presented in Figure 6.1. It can be observed from Figure 6.1 that the resultant BER versus E_b/N_0 performance is the same, regardless of the video test sequences considered. Similarly, the PSNR-Y versus E_b/N_0 performance of the ”Akiyo”, ”News” and the ”Foreman” video test sequences is shown in Figure 6.2. It can be observed from Figure 6.2 that the objective video quality of the low bit-rate video coding configuration of Table 4.9 is dependent on the amount of motion activity within the video scene. The ”Akiyo” video sequence associated with the lowest motion activity has the best PSNR performance relative to the ”News” video sequence. Similarly, due to the high motion activity of the ”Foreman” video sequence, the ”News” video sequence has a better objective video quality than the ”Foreman” sequence in the E_b/N_0 range considered in Figure 6.2.

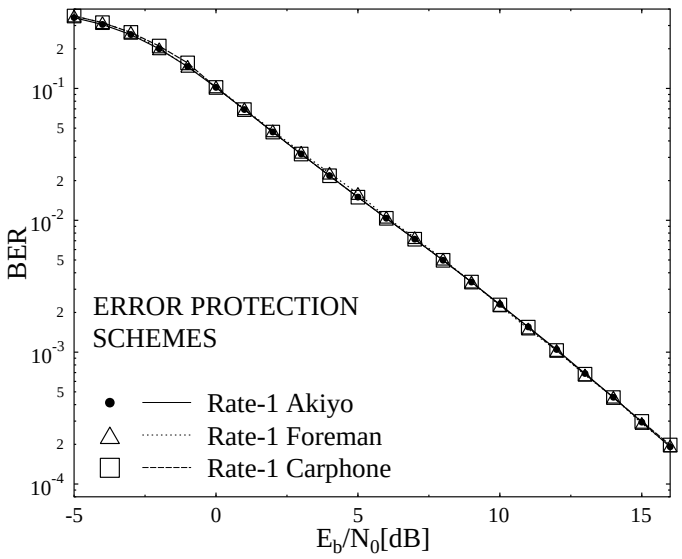

Figure 6.1: BER versus E_b/N_0 performance of the ”Akiyo”, ”News” and the ”Foreman” video test sequencies using Rate1 error protection schemes of Table 4.9(a), while utilizing the system architecture of Figure 4.10

.

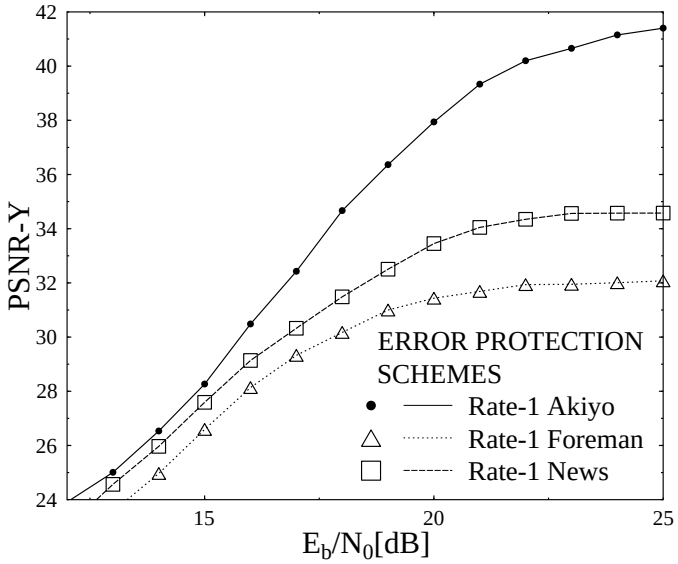

Figure 6.2: PSNR-Y versus E_b/N_0 performance of the "Akiyo", "News" and the "Foreman" video test sequencies using Rate1 error protection schemes of Table 4.9(a), while utilizing the system architecture of Figure 4.10

In Section 4.5.5 EXIT charts were used for analysing the attainable system performance. We observed from the decoding trajectories of Figures 4.12, 4.13, 4.14 and 4.15 that the convergence behaviour of the EOSBCs considered degrades upon increasing their coding rate. Explicitly, our experimental results of Figure 4.21 show that the proposed error protection scheme using rate-$\frac{1}{3}$ EOSBCs having $d_{H,min} = 6$ outperforms the identical-rate EOSBCs having $d_{H,min} = 3$ by about 2.5 dB at the $PSNR$ degradation point of 1 dB. Additionally, an E_b/N_0 gain of 8 dB was achieved in Figure 4.21 compared to the rate-$\frac{5}{6}$ EOSBC having $d_{H,min} = 2$ and an identical overall code-rate. Furthermore, an E_b/N_0 gain of 25 dB was attained in Figure 4.23 at the PSNR degradation point of 1 dB, while using iterative soft-bit source and channel decoding with the aid of rate-$\frac{1}{3}$ EOSBCs relative to the identical-rate benchmarker.

Furthermore, as seen in Figure 4.26 of Section 4.6, we analysed the performance of the proposed EOSBC algorithms using a Rate-1 precoder as our inner code. In Section 4.6.3 EXIT charts were used to record the Monte-Carlo simulation based decoding trajectories of the various error protection schemes employing the different rate-$\frac{1}{3}$ outer EOSBCs as well as using the rate-1 inner precoder detailed in Table 4.11(a). It was observed in Section 4.6.3 that as expected, the convergence behaviour of the EOSBCs improves upon increasing $d_{H,min}$. Explicitly, the H.264/SBC/rate-1 precoder design example exhibited an E_b/N_0 gain of 3 dB in Figure 4.32 at the $PSNR$ degradation point of 1 dB, when using SBCs having $d_{H,min} = 6$ compared to the identical-rate EOSBCs having $d_{H,min} = 3$. Moreover, an E_b/N_0 gain of 27 dB was attained in Figure 4.34, when using iterative USSPA source and channel decoding with the aid of rate-$\frac{1}{3}$ EOSBCs relative to the identical-rate benchmarker of Figure 4.26.

Moreover, in Section 4.7 further redundancy was introduced in the source coded streams by transforming the EOSBC algorithms of Section 4.5 in a systematic way, which resulted in a further increase in

the $d_{H,min}$ of the generated symbols. It was observed from Figures 4.37, 4.38, 4.39, and 4.40 that the convergence behaviour of the EOSBC coding improves upon increasing $d_{H,min}$.

Our design of Figure 4.35, which was based on the H.264, RSM and RSC constituent components, exhibited an E_b/N_0 gain of 5 dB in Figure 4.42 at the $PSNR$ degradation point of 2 dB, when using the RSM_2^6 scheme of Table 4.14 associated with $d_{H,min} = 4$ compared to the employment of the RSM_2^3 having $d_{H,min} = 2$, which in turn outperformed the RSM_5^6 having both an identical $d_{H,min}$ and an identical overall system code-rate by about 2 dB at the $PSNR$ degradation point of 2 dB. Moreover, an E_b/N_0 gain of 20 dB was attained in Figure 4.44 using iterative USSPA source and channel decoding with the aid of the RSM_2^3 of Table 4.14 relative to the identical-rate benchmarker of Figure 4.35 dispensing with RSM.

- **Chapter 5:**

  In Section 5.2 we evaluated the performance of the UEP schemes of Figure 5.3 proposed in Section 3.10, which is a sophisticated low-complexity SP modulation aided MIMO DSTS scheme, consisting of two transmit and one receive antenna, dispensing with CIR estimation. Explicitly, an E_b/N_0 gain of 4 dB was attained in Figure 5.6 using the UEP1 scheme of Table 5.3(a) with reference to UEP2, while an E_b/N_0 gain of 2 dB was achieved with reference to the EEP scheme, at the PSNR degradation point of 2 dB. Furthermore, with reference to the system setup of Section 3.10, which utilised the identical coding scheme of Figure 5.3 but dispensed with SP modulation aided DSTS transmission, in Figure 5.6 an E_b/N_0 gain of about 10 dB was attained relative to Figure 3.26 of Section 3.10.

  Additionally, in Section 5.3 we contrived the novel USSPA source and channel decoding arrangement of Figure 5.7 for the transmission of an H.264/AVC coded video bit-stream, in order to exploit the residual redundancy, which inherently remained in the coded bit-stream after source and channel encoding. The resultant arrangement contributed towards the reliable generation of extrinsic informantion. The resultant USSPA UEP aided bit-stream was transmitted using the DSTS aided SP modulation scheme of Figure 5.7 for attaining a diversity gain without the need for any high-complexity MIMO channel estimation. Quantitatively, an E_b/N_0 gain of 3 dB was attained in Figure 5.10, when using the UEP1 scheme of Table 5.3(a) in comparison to UEP2, while an E_b/N_0 gain of 2 dB was achieved by the latter with reference to the EEP scheme at the PSNR degradation point of 2 dB. Furthermore, an E_b/N_0 gain of about 10 dB was achieved, when comparing the PSNR-Y vs E_b/N_0 curves of Figure 5.10 to those of Figure 4.9, where the latter relied on similar error protection schemes, but dispensing with SP modulation aided DSTS transmission.

  Furthermore, in Section 5.4 we incorporated a novel SBC scheme by partitioning the total available bit-rate budget between the source and channel codecs, which resulted in an improved performance, when the ISCD scheme of Figure 5.11 was employed. After RSC encoding the SBC coded stream was transmitted using the SP modulation aided DSTS transceiver of Figure 5.11. Explicitly, an E_b/N_0 gain of 3 dB was attained in Figure 5.14 using the UEP1 scheme of Table 5.10(a) over the UEP2 scheme, while an E_b/N_0 gain of 1 dB was achieved by the latter with reference to the EEP scheme at the PSNR degradation point of 2 dB. Furthermore, we compared the PSNR-Y vs E_b/N_0 curves of Figure 5.14 obtained in Section 5.4.4 to those of Figure 3.26 in Section 3.10, which employed similar type of error protection schemes, but

dispensed with SP modulation aided DSTS transmission. It was observed from Figure 5.10 that an E_b/N_0 gain of about 10 dB can be achieved with reference to Figure 4.9 of Section 4.4.

Moreover, in Section 5.5, the performance of EOSBCs was analysed using the system architecture of Figure 5.11. Additionally, in Section 5.5.3 the performance of the UEP SBC scheme employed was analysed with the aid of EXIT charts. Explicitly, when using the code rates of Table 5.14(a) for the different error protection schemes, an E_b/N_0 gain of about 12 dB was attained in Figure 5.27 using EOSBCs relative to the results of Figure 5.26 dispensing with SBC, when observed at the PSNR degradation point of 1 dB.

Finally, in Figure 5.34 of Section 5.6.2 a three-stage system design was presented, which was constituted by serially concatenated and iteratively decoded EOSBCs and a precoded DSTS aided multi-dimensional SP modulation scheme designed for near-capacity joint source and channel coding. As detailed in Section 5.6.7, the employment of EOSBCs, which deliberately imposed further artificial redundancy on the source coded bit stream provided significant improvements in terms of the $PSNR$ versus E_b/N_0 performance, when compared to the benchmarker scheme employing equivalent-rate EOSBCs, which were not optimised for achieving the best possible iterative convergence behaviour. Additionally, the convergence behaviour of the system of Figure 5.34 was analysed with the aid of EXIT charts in Figure 5.35 and 5.36 of Section 5.6.6. It has been pointed out in the context of Figures 5.35 and 5.36 that although the joint EXIT function of the non-optimised inner EOSBC encoder and unity rate precoder cannot reach the point of perfect convergence at (1,1), the joint EXIT function of the optimised inner EOSBC encoder operating in conjunction with the rate-1 precoder can. The H.264/ SBC/ URC/DSTS-SP based three-stage design example of Figure 5.34 using SBCs having $d_{H,min} = 3$ exhibited an E_b/N_0 gain of 22 dB in Figure 5.38 at the PSNR degradation point of 2 dB relative to the identical-rate benchmarker employing equivalent-rate EOSBCs having $d_{H,min} = 1$, as detailed in Section 5.6.7.

Table 6.1: Thesis Summary Table

Chapter #	Schematic	Parameters	Iterations	Video rate	FEC code rate	Highest PSNR value	Error protection scheme	$E_b/N_0[dB]$ at $1dB$ PSNR degradation	$E_b/N_0[dB]$ at $2dB$ PSNR degradation
Chapter 3	Figure 3.27	Table 3.9	10	64 $kbps$	0.7	41.7	EEP	5.0	4.8
							UEP1	5.5	5.1
							UEP2	4.8	4.6
Chapter 4	Figure 4.6	Table 4.4	10	64 $kbps$	0.5	41.7	EEP	37.0	33.5
							UEP1	39.0	35.0
							UEP2	35.0	31.0
	Figure 4.10	Table 4.9	10	64 $kbps$	0.25	41.7	$\frac{2}{3}$ $SBC_{[2,\ 3]}$	3.0	1.5
							R=$\frac{3}{4}$ $SBC_{[3,\ 4]}$	3.5	2.5
							R=$\frac{4}{5}$ $SBC_{[4,\ 5]}$	4.0	3.0
							R=$\frac{5}{6}$ $SBC_{[5,\ 6]}$	5.0	3.5
							R=$\frac{1}{3}$ $SBC_{[2,\ 6]}$	-1.7	-2.0
							R=$\frac{1}{3}$ $SBC_{[3,\ 9]}$	-3.2	-3.5
							R=$\frac{1}{3}$ $SBC_{[4,\ 12]}$	-3.7	-3.8
							R=$\frac{1}{3}$ $SBC_{[5,\ 15]}$	-4.0	-4.1
	Figure 4.26	Table 4.11	10	64 $kbps$	0.33	41.7	$\frac{1}{3}$ $SBC_{[2,\ 6]}$	-2.0	0.5
							R=$\frac{1}{3}$ $SBC_{[3,\ 9]}$	-1.7	-1.6
							R=$\frac{1}{3}$ $SBC_{[4,\ 12]}$	-1.5	-1.8
							R=$\frac{1}{3}$ $SBC_{[5,\ 15]}$	1.0	-2.2
	Figure 4.35	Table 4.15	10	64 $kbps$	0.25	41.7	$\frac{2}{3}$ SBC_2^3	3.0	1.5
							R=$\frac{3}{4}$ SBC_3^4	3.5	2.5
							R=$\frac{4}{5}$ SBC_4^5	3.7	3.0
							R=$\frac{5}{6}$ SBC_5^6	4.5	3.2
							R=$\frac{1}{3}$ SBC_2^6	-3.1	-3.2
							R=$\frac{1}{3}$ SBC_3^8	-2.2	-2.2
							R=$\frac{1}{3}$ SBC_4^{10}	-2.3	-2.8
							R=$\frac{1}{3}$ SBC_5^{12}	-2.7	-2.9

Table 6.2: Thesis Summary Table

Chapter #	Schematic	Parameters	Iterations	Video rate	FEC code rate	Highest PSNR value	Error protection scheme	$E_b/N_0[dB]$ at $1dB$ PSNR degradation	$E_b/N_0[dB]$ at $2dB$ PSNR degradation
Chapter 5	Figure 5.3	Table 5.3	10	64 $kbps$	0.5	41.7	EEP	26.0	24.0
							UEP1	28.0	24.5
							UEP2	24.0	22.0
	Figure 5.7	Table 5.6	10	64 $kbps$	0.5	41.7	EEP	24.0	23.0
							UEP1	26.0	24.0
							UEP2	22.5	21.0
	Figure 5.11	Table 5.10	10	64 $kbps$	0.5	41.7	EEP	14.5	15.0
							UEP1	13.5	14.5
							UEP2	16.5	17.5
	Figure 5.11	Table 5.14	10	64 $kbps$	0.33	41.7	EEP	13.5	13.0
							UEP1	13.0	12.5
							UEP2	15.2	14.5
	Figure 5.34	Table 5.18	10	64 $kbps$	0.33	41.7	$I_t(3)$	8.6	9.6
							$I_t(4)$	9.5	9.0
							$I_t(5)$	10.0	8.5

6.2 Future Work

The main features of wireless video and multimedia applications include a high compression efficiency, convenient integration of the coded video into heterogeneous communication networks and enhanced error-resilience features. The support of all these features makes the H.264/AVC codec an attractive candidate for all wireless applications, including Multimedia Messaging Services (MMS), Packet-Switched Streaming Services (PSS) and conversational applications. Therefore, given the beneficial properties of the H.264 coding standard, we will focus our future research suggestions on utilising the H.264 coding standard for different video coding and transmission scenarios.

6.2.1 Efficient Coding/Transmission of High Bit-Rate Video

High bit-rate distributive broadcast video applications such as High Definition Television (HDTV) have different requirements compared to low-bit rate video. Typical HDTV systems employ 1280x720- and 1920x1080-pixels standard resolution images scanned at 60 frames/second. HDTV overcomes most of the limitations of Standard Digital TV (SDTV), which is unable to provide high-quality results for large-screen TVs and projectors. It seems likely that in the future all European HDTV applications may be H.264 based. The limiting factors of low bit-rate video coding applications such as low processing power and low transmission bandwidth requirements are typically less stringent for high bit-rate video. Therefore, more sophisticated source and channel coding techniques can be utilised for the sake of achieving an improved performance.

The H.264/AVC FRExt amendment described in Section 2.8.4 specifies three further nested sets of profiles in addition to the main profile. These profiles extend the capabilities of the standard to provide services for more advanced applications, such as high-definition consumer applications, including HDTV and high-quality computer monitors. Hence it is beneficial to extend our current experience with using the H.264 main profile described in Section 2.8.2 to the H.264/AVC FRExt profile and analyse the attainable performance improvements using JSCD techniques.

6.2.2 Scalable Video Coding/Transmission

Scalable video coding may be designed for graceful degradation in error-infested transmission environments along with its capability to adapt different formats and bit-rates along with power adaptation [113, 256–258]. It allows the decoding of partially received bit-streams to provide video services at reduced temporal or spatial resolutions. As the achievable reconstructed video quality is closely related to the partial bit-stream rate, this allows us to support both low- and high-bit rate video applications. As part of our future work, we will extend our current video coding investigations to scalable video, in order to exploit the redundancy among the base-layer and the enhancement layers of the SVC coded stream using the H.264 SVC codec.

Our investigations were based on a fixed modulation/coding schemes. However, it is unrealistic to expect that a fixed-rate scheme remains capable of delivering a constant QoS in the face of channel-quality fluctuations reaching 40 dB. Fortunately, scalable video coding is capable of delivering the video at the specific bit-rate

facilitated by the instantaneous channel quality, when using for example High-Speed Packet Access (HSPA) style near-instantaneously adaptive transceivers [203, 259–261].

6.2.3 3D Video Coding/Transmission

Current 3D video research activities include 3D content creation, coding/compression, transmission and display. Further research activities in this area may focus on 3D video coding/compression and error-resilient transmission, since 3D video based on multi-view coding is receiving significant attention among researchers [262–266]. The idea is to efficiently encode multiple camera views. The community's interest is in the generation of highly compressed multi-view streams based on inter-view prediction in order to exploit the redundancy among multiple views.

The successful implementation of H.264/AVC substantially contributed towards stimulating the recent interest in 3D and multi-view coding. The coding gain achieved by the H.264/AVC scheme can be utilised to create 3D and multi-view television. H.264/AVC based video coding is efficient, and the coding artefacts are relatively low, hence H.264/AVC based 3D services may be expected to emerge in the near-future. H.264/AVC is also used for the compression of different stereo-scopic views, where the inherent redundancy among the stereoscopic views is exploited using inter-view prediction [267]. In the future we will invoke our existing coding techniques along with a range of other sophisticated strategies in order to exploit this inter-layer redundancy for the robust encoding of low-bit rate multiview video.

6.2.4 Video Over IP

Video over IP has found its way into SKYPE, which combines a packet switched network with streaming video. Unlike data transfer over IP, the video-streaming quality may be evaluated in real time at the end point. Furthermore, the quality is not solely a function of the video codec or of the network bandwidth, but of a combination of both. Unlike data integrity, the video quality depends on a combination of receiving reliable packets at a high throughput, while bearing in mind that different types of packets may have different importance. Since severe video quality degradation may occur during network congestion, layered video transmission associated with Differentiated Services (DiffServ) [268] may be used, where different video layers are mapped to different priority levels. Similarly, different IP packets associated with different priorities receive different treatment within the network, which results in a graceful degradation of the attainable video quality. Various layering mechanisms, such as DP , temporal scalability and SNR scalability can be utilised to maximise the perceived video quality for transmission in certain network conditions. Scalable video coding associated with built-in scalability features provides the best solution for this type of environment.

This application requires the additional employment of packet-layer FEC schemes, such as Fountain Codes [269] or Luby Transform (LT) codes [270] for the sake of recovering the packets that may have been dropped owing statistical multiplexing induced packet-loss events at the routers.

6.2.5 Digital Video Broadcasting

Digital Video Broadcast (DVB) employs a suite of open standards, including DVB over satellites (DVB-S) [271], DVB over Terrestrial (DVB-T) [272], DVB over cable (DVB-C) [273] and for transmission to hand-held mobile devices (DVB-H) [274, 275]. These distribution systems differ mainly in terms of their modulation scheme used. DVB provides different features, including HDTV over broadcast networks, in order to enable video broadcast to mobile pocket TV receivers, TV receivers in vehicles, while supporting the flexible choice of image, audio and HDTV quality, which may be achieved via an interactive channel between the viewer and network operator. The capabilities provided in the form DVB-H for hand-held devices include the support of mobility, smaller screens and antennas, indoor coverage and the reliance on battery power [179] [275].

Our existing designs, which relied on using H.264 video coding and transmission are also applicable to DVB type systems, noting that further performance improvements may be achieved, since DVB signals can tolerate longer interleavers and hence exhibit an increased error resilience.

6.2.6 Exploiting Redundancy in The Video Domain:

We analysed different ISCD strategies in order to exploit the residual redundancy of the source. The extrinsic information obtained after exploiting the residual redundancy was exchanged iteratively between the constituent decoders in order to assist each other in subsequent iterations. The philosophy of ISCD can be extended to the video domain in order to more efficiently exploit the high correlation among the consecutive frames.

6.2.7 Cooperative Source and Channel Coding For Wireless Video Transmission:

In our future work we will also consider the joint optimisation of the proposed video source and channel coding strategies in combination with cooperative communication [276, 277] to improve the decoded video quality.

6.2.8 3D EXIT Chart Based Three-Stage System Design Using Iterative Detection Aided H.264 Wireless Video Telephony

In Figure 5.34 of Section 5.6.2 we advocated a three-stage serially concatenated scheme, which was designed for near-capacity operation employing two iterations between the inner decoder and the intermediate code's decoder, which were referred to as inner iterations, as well as between the outer code's decoder and the intermediate code's decoder, which were termed as outer iterations. The performance of the system was analysed in Section 5.6.6 using 2D EXIT charts, which considers the intermediate URC decoder and the USSPA source decoder as a single combined outer Soft-Input Soft-Output (SISO) module. However, this figure does not characterise the convergence behaviour of the URC decoder and of the USSPA source decoder alone. Therefore, in our future work we will perform 3D EXIT chart analysis of this system in order to provide more close insights for this sophisticated three-stage system design.

6.2.9 Distributed Video Coding

The recent video coding strategy encouraged by the information-theoratic results published by Slepian and Wolf [278] for lossless source coding as well as by Wyner and Ziv [279] for lossy source coding resulted in a new video paradigm for video coding, known as distributed video coding. The conventional video compression standards, such as ISO MPEG [280] schemes or ITU-T recommendations H.263 [281] and H.264 [84] exploit the source redundancy at the encoder. These encoders are based on the assumption that the video is encoded once and decoded several times. However, the development of rich uplink media applications such as multimedia sensor networks, mobile video phones, PDAs, wireless laptop cameras etc., imposed the requirement of low-complexity encoders, shifting the processing complexity to the decoder. In distributed video coding the computationally complex task of motion estimation is shifted to the decoder. A similar video coding paradigm based on the principles of coding with side information has been developed by Puri and Ramchandran, which was presented under the acronym PRISM (power-efficient, robust, high-compression, syndrome-based multimedia coding) in [217,282,283]. In recent years considerable research efforts have been dedicated to the design of distributed video coding systems [284–286].

As part of our future work we will consider video quality optimisation in distributed video coding scenarios, while employing different cooperative and iterative joint source and channel coding strategies.

Bibliography

[1] L. Hanzo, P. Cherriman, and J. Streit, *Video Compression and Communications: From Basics to H.261, H.263, H.264, MPEG2, MPEG4 for DVB and HSDPA-Style Adaptive Turbo-Transceivers.* Wiley-IEEE Press, September 2007.

[2] Nasruminallah and L. Hanzo, "Short block codes for guaranteed convergence in soft-bit assisted iterative joint source and channel decoding," *IEEE Electronics Letters*, vol. 44, pp. 1315–1316, 23 2008.

[3] Nasruminallah and L. Hanzo, "EXIT-chart optimized short block codes for iterative joint source and channel decoding in H.264 video telephony," *IEEE Transactions on Vehicular Technology*, vol. 58, pp. 4306 –4315, October 2009.

[4] Nasruminallah, M. El-Hajjar, N. S. Othman, A. P. Quang, and L. Hanzo, "Over-complete mapping aided, soft-bit assisted iterative unequal error protection H.264 joint source and channel decoding," *IEEE 68th Vehicular Technology Conference (VTC 2008-Fall), (Calgary, CANADA)*, pp. 1 –5, 21 to 24 September 2008.

[5] Nasruminallah and L. Hanzo, "Convergence behaviour of iteratively decoded short block-codes in H.264 joint source and channel decoding," *IEEE 69th Vehicular Technology Conference (VTC Spring-2009), (Barcelona, SPAIN)*, pp. 1 –5, 26 to 29 April 2009.

[6] D. Yang, Nasruminallah, L.-L. Yang, and L. Hanzo, "SVD-aided unequal-protection spatial multiplexing for wireless video telephony," *IEEE 69th Vehicular Technology Conference (VTC Spring 2009), (Barcelona, SPAIN)*, pp. 1 –5, 26 to 29 April 2009.

[7] Nasruminallah, R. G. Maunder, and L. Hanzo, "Iterative detection aided H.264 wireless video telephony using irregular convolutional codes," *IEEE 70th Vehicular Technology Conference (VTC 2009-Fall), (Alaska, USA)*, pp. 1 –5, 20 to 23 September 2009.

[8] Nasruminallah, M. El-Hajjar, and L. Hanzo, "Robust transmission of H.264 coded video using three-stage iterative joint source and channel decoding," *IEEE Global Telecommunications Conference (GLOBECOM-2009), (Hawaii, USA)*, pp. 1 –5, 30 November - 4 December 2009.

[9] Nasruminallah and L. Hanzo, "Exploiting redundancy in iterative H.264 joint source and channel decoding for robust video transmission," *IEEE Vehicular Technology Conference Fall (VTC 2009-Fall), (Taipei, TAIWAN)*, pp. 1 –5, 16 to 19 May 2010.

[10] Nasruminallah, M. F. U. Butt, S. X. Ng, and L. Hanzo, "H.264 wireless video telephony using iteratively-detected binary self-concatenated coding," *IEEE 71st Vehicular Technology Conference Fall (VTC 2010-Spring), (Taipei, TAIWAN)*, pp. 1 –5, 16 to 19 May 2010.

[11] Nasruminallah, M. El-Hajjar, and L. Hanzo, "Iterative H.264 source and channel decoding using sphere packing modulation aided layered steered space-time codes," *IEEE International Conference on Communications (ICC 2010), (Cape Town, SOUTH AFRICA)*, pp. 1 –5, 23 to 27 May 2010.

[12] M. Miyahara, "Improvements of television picture quality by eliminating the disturbances caused by interlaced scanning," *IEEE Transactions on Communications*, vol. 31, pp. 902 – 906, July 1983.

[13] P. H. Saul, "Analogue-digital conversion handbook," *Electronics and Power*, vol. 32, p. 396, May 1986.

[14] Z. He, W. Zeng, and C. W. Chen, "Low-pass filtering of rate-distortion functions for quality smoothing in real-time video communication," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, pp. 973 – 981, August 2005.

[15] A. M. Bock, "Design criteria for video sampling rate conversion filters," *IEEE Electronics Letters*, vol. 26, pp. 1259 –1260, August 1990.

[16] R. Feghali, F. Speranza, D. Wang, and A. Vincent, "Video quality metric for bit rate control via joint adjustment of quantization and frame rate," *IEEE Transactions on Broadcasting*, vol. 53, pp. 441 –446, March 2007.

[17] Y.-H. Kim, B. Choi, and J. Paik, "High-fidelity RGB video coding using adaptive inter-plane weighted prediction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, pp. 1051 –1056, July 2009.

[18] R. Mancuso, S. Smorfa, and M. Olivieri, "A novel high-quality yuv-based image coding technique for efficient image storage in portable electronic appliances," *IEEE Transactions on Consumer Electronics*, vol. 54, pp. 695 –702, May 2008.

[19] *CCIR Recommendation 601: 'Digital methods of transmitting television information'*. Recommendation 601, encoding parameters of digital television for studios., September.

[20] G. M. Cortelazzo, E. Malavasi, M. Segato, A. Baschirotto, and G. Biaggioni, "SC realization of antialiasing CCIR-601 video filters," *IEEE Transactions on Consumer Electronics*, vol. 40, pp. 274 –281, August 1994.

[21] E. Dumic, M. Mustra, S. Grgic, and G. Gvozden, "Image quality of 4: 2: 2 and 4:2:0 chroma subsampling formats," *International Symposium ELectronic MARketing (ELMAR-2009)*, pp. 19 –24, September 2009.

[22] K. Roa, Z. S. Bojkovic, and D. A. Milanvanovic, *Multimedia Communication Systems: Techniques, Standards, and Networks*. Prentice Hall, March 2002.

[23] M. Ghanbari, *Standard Codecs: Image compression to advanced Video Coding*. IEE Telecommunications Series, 2003.

[24] K. W. Sum and R. D. Murch, "Very low bit rate segmented video coding for visual telephony," *IEEE International Symposium on Circuits and Systems (ISCAS-1994)*, vol. 3, pp. 197 –200 vol.3, June 1994.

[25] P. L. Correia and F. Pereira, "Objective evaluation of video segmentation quality," *IEEE Transactions on Image Processing*, vol. 12, pp. 186 – 200, February 2003.

[26] A. Leontaris, P. C. Cosman, and A. R. Reibman, "Quality evaluation of motion-compensated edge artifacts in compressed video," *IEEE Transactions on Image Processing*, vol. 16, pp. 943 –956, April 2007.

[27] K. Seshadrinathan and A. C. Bovik, "Motion tuned spatio-temporal quality assessment of natural videos," *IEEE Transactions on Image Processing*, vol. 19, pp. 335 –350, February 2010.

[28] M. Vranjes, S. Rimac-Drlje, and D. Zagar, "Subjective and objective quality evaluation of the H.264/AVC coded video," *15th International Conference on Systems, Signals and Image Processing (IWSSIP-2008)*, pp. 287 –290, June 2008.

[29] P. K. Aeluri, V. Bo, S. Richie, and A. Weeks, "Objective quality analysis of MPEG-1, MPEG-2 windows media video," *IEEE 6th Southwest Symposium on Image Analysis and Interpretation*, pp. 221 – 225, March 2004.

[30] Q. Huynh-Thu and M. Ghanbari, "Scope and validity of PSNR in image/video quality assessment," *IEEE Electronics Letters*, vol. 44, pp. 800 –801, September 2008.

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

[32] J. Jain and A. Jain, "Displacement measurement and its application in interframe image coding," *IEEE Transactions on Communications*, vol. 29, no. 12, pp. 1799–1808, 1981.

[33] L. C. Perez and K. Sayood, "Joint source/channel coding for variable length codes using a precoder," *IEEE Wireless Communications and Networking Conference (WCNC-1999)*, vol. 2, pp. 983 –987, 1999.

[34] A. Saxena and K. Rose, "Distributed predictive coding for spatio-temporally correlated sources," *IEEE Transactions on Signal Processing*, vol. 57, pp. 4066 –4075, October 2009.

[35] S. A. Rizvi and N. M. Nasrabadi, "Lossless image compression using modular differential pulse code modulation," *Proceedings of International Conference on Image Processing (ICIP-1999)*, vol. 1, pp. 440 –443 vol.1, 1999.

[36] S. R. Subramanya and C. Sabharwal, "Performance evaluation of hybrid coding of images using wavelet transform and predictive coding," *Proceedings of Fourth International Conference on Computational Intelligence and Multimedia Applications (ICCIMA 2001)*, pp. 426 –431, 2001.

[37] K. W. Sum and R. D. Murch, "Generalized region based transform coding for video compression," *Proceedings of Data Compression Conference (DCC-1995)*, p. 478, March 1995.

[38] S. Rhee, Y. Kim, H. M. Jung, and K. T. Park, "A uniform reconstruction level quantizer with improved performance at low bit rates," *IEEE Asia Pacific Conference on Circuits and Systems (APC-1996)*, pp. 149–152, November 1996.

[39] R. C. Gonzalez and R. E. Woods, *Digital Image Processing.* Published by Prentice Hall, March 2007.

[40] S. So and K. K. Paliwal, "Multi-frame GMM-based block quantisation for distributed speech recognition under noisy conditions," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2006)*, vol. 1, pp. I –I, May 2006.

[41] Y.-L. Chan and W.-C. Siu, "Fast interframe transform coding based on characteristics of transform coefficients and frame difference," *IEEE International Symposium on Circuits and Systems (ISCAS-1995)*, vol. 1, pp. 449 –452 vol.1, May 1995.

[42] C.-K. Wong, O. C. Au, and C.-W. Tang, "Motion compensated temporal interpolation with overlapping," in *IEEE International Symposium on Circuits and Systems (ISCAS-1996)*, vol. 2, (Atlanta, GA, USA), pp. 608–611, May 1996.

[43] Y. Bin and D. Hui-chuan, "Image stabilization by combining gray-scale projection and block matching algorithm," *IEEE International Symposium on IT in Medicine Education (ITIME 2009)*, vol. 1, pp. 1262 –1266, August 2009.

[44] V. Seferidis and M. Ghanbari, "General approach to block-matching motion estimation," *Optical Engineering*, vol. 32, no. 7, pp. 1464–1474, 1993.

[45] A.-M. Huang and T. Nguyen, "Correlation-based motion vector processing with adaptive interpolation scheme for motion-compensated frame interpolation," *IEEE Transactions on Image Processing*, vol. 18, pp. 740 –752, April 2009.

[46] A. Tamaki, K. Kato, and T. Yanaru, "On the analysis of contour figures by using cross correlation function," *Proceedings of the 34th SICE Annual Conference (SICE-95)*, pp. 1195 –1200, July 1995.

[47] E. Linzer, P. Tiwari, and M. Zubair, "High performance algorithms for MPEG motion estimation," *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP-1996)*, vol. 4, pp. 1934 –1937 vol. 4, May 1996.

[48] M. C. Chen and A. N. Willson, "Rate-distortion optimal motion estimation algorithms for motion-compensated transform video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, pp. 147 –158, April 1998.

[49] A. Erturk and S. Erturk, "Two-bit transform for binary block motion estimation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, pp. 938 – 946, July 2005.

[50] K. B. Kim, Y. Jeon, and M.-C. Hong, "Variable step search fast motion estimation for H.264/AVC video coder," *IEEE Transactions on Consumer Electronics*, vol. 54, pp. 1281 –1286, August 2008.

[51] *Applications of Digital Image Processing*. Published by Society of Photo-optical Instrumentation Engineers, IEEE Computer Society, December 2007.

[52] S. Zhu and K.-K. Ma, "A new diamond search algorithm for fast block-matching motionestimation," *IEEE Transactions on Image Processing*, vol. 9, pp. 287–290, February 2000.

[53] S. Jin, S.-J. Park, and J. Jeong, "Adaptive fast full search algorithm using partitioned region and optimized search order," *IEEE Transactions on Consumer Electronics*, vol. 53, pp. 1703 –1711, November 2007.

[54] H. Jia and L. Zhang, "A new cross diamond search algorithm for block motion estimation," *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2004. Proceedings (ICASSP-2004)*, vol. 3, pp. iii–(357–360), May 2004.

[55] H. A. Ilgin and L. F. Chaparro, "Low bit rate video coding using dct-based fast decimation/interpolation and embedded zerotree coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, pp. 833 –844, July 2007.

[56] R. Talluri, K. Oehler, T. Barmon, J. D. Ourtney, A. Das, and J. Liao, "A robust, scalable, object-based video compression technique for very low bit-rate coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, pp. 221 –233, February 1997.

[57] G. K. Wallace, "The JPEG still picture compression standard," *IEEE Transactions on Consumer Electronics*, vol. 38, pp. xviii –xxxiv, February, 1992.

[58] T. Wiegand, M. Lightstone, D. Mukherjee, T. G. Campbell, and S. K. Mitra, "Rate-distortion optimized mode selection for very low bit rate video coding and the emerging H.263 standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, pp. 182 –190, April 1996.

[59] D. Lauzon, A. Vincent, and L. Wang, "Performance evaluation of MPEG-2 video coding for HDTV," *IEEE Transactions on Broadcasting*, vol. 42, pp. 88 –94, June 1996.

[60] N. Kamaci and Y. Altunbasak, "Performance comparison of the emerging H.264 video coding standard with the existing standards," *International Conference on Multimedia and Expo (ICME-2003)*, vol. 1, pp. I – 345–8 vol.1, July 2003.

[61] F. Halsall, *Multimedia Communications: Applications, Networks, Protocols and Standards*. Addison-Wesley, March 2001.

[62] R. Schafer and T. Sikora, "Digital video coding standards and their role in video communications," *Proceedings of the IEEE*, vol. 83, pp. 907 –924, June 1995.

[63] K. R. Rao and P. Y. Huifang, *Discrete Cosine Transform: Algorithms, Advantages, Applications.* Published by Academic Press, December 2007.

[64] C.-K. Wong and O. C. Au, "Fast motion compensated temporal interpolation for video," in *Proceedings of SPIE Symposium of Visual Communications and Image Processing, (Taipei, Taiwan)*, vol. 2, pp. 1108–1118, June 1995.

[65] C.-W. Tang and O. C. Au, "Unidirectional motion compensated temporal interpolation," in *IEEE International Symposium on Circuits and Systems (ISCAS-1997)*, vol. 2, pp. 1444–1447, June 1997.

[66] O. A. Ojo and H. Schoemaker, "Adaptive global concealment of video up-conversion artefacts," *IEEE Transactions on Consumer Electronics*, vol. 47, pp. 40 –46, February 2001.

[67] X. Chen, C. N. Canagarajah, and J. L. Nunez-Yanez, "Backward adaptive pixel-based fast predictive motion estimation," *IEEE Signal Processing Letters*, vol. 16, pp. 370 –373, May 2009.

[68] G. J. Sullivan and T. Wiegand, "Video compression - from concepts to the H.264/AVC standard," *Proceedings of the IEEE*, vol. 93, pp. 18 –31, January 2005.

[69] L. Hanzo, T. H. Liew, and B. L. Yeap, *Turbo Coding, Turbo Equalisation and Space-Time Coding for Transmission over Fading Channels*. New York, USA: John Wiley & Sons, 2002.

[70] A. Ashikhmin, G. Kramer, and S. ten Brink, "Extrinsic information transfer functions: model and erasure channel properties," *IEEE Transactions on Information Theory*, vol. 50, pp. 2657 – 2673, November 2004.

[71] S. Yatawatta and A. Petropulu, "Blind channel estimation in MIMO OFDM systems with multiuser interference," *IEEE Transactions on Signal Processing [see also IEEE Transactions on Acoustics, Speech, and Signal Processing]*, vol. 54, pp. 1054–1068, March 2006.

[72] M. El-Hajjar, O. Alamri, S. X. Ng, and L. Hanzo, "Turbo detection of precoded sphere packing modulation using four transmit antennas for differential space-time spreading," *IEEE Transactions on Wireless Communications*, vol. 7, pp. 943 –952, March 2008.

[73] ISO-IEC/JTC1/SC29/WG11, "Information technology-coding of audio visual objects-part 10: Advanced video coding," *Final Draft International Standard, ISO/IEC FDIS 14 496-10*, December 2003.

[74] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, and T. Wedi, "Video coding with H.264/AVC: tools, performance, and complexity," *IEEE Circuits and Systems Magazine*, vol. 4, pp. 7–28, January 2004.

[75] N. Kamnoonwatana, D. Agrafiotis, and C. N. Canagarajah, "Exploiting MPEG-7 texture descriptors for fast H.264 mode decision," *IEEE 15th International Conference on Image Processing (ICIP-2008), (California, USA)*, pp. 2796 –2799, October 2008.

[76] T. Stockhammer, M. M. Hannuksela, and T. Wiegand, "H.264/AVC in wireless environments," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, pp. 657–673, July 2003.

[77] T. Stockhammer and M. Bystrom, "H.264/AVC data partitioning for mobile video communication," *IEEE International Conference on Image Processing (ICIP-2004 Singapore)*, vol. 1, pp. 545–548, October 2004.

[78] D. Lefol, D. Bull, and C. N. Canagarajah, "Performance evaluation of transcoding algorithms for H.264," *IEEE Transactions on Consumer Electronics*, vol. 52, pp. 215 –222, February 2006.

[79] O. Nemethova, J. C. Rodriguez, and M. Rupp, "Improved detection for H.264 encoded video sequences over mobile networks," *IEEE 8th International Symposium on Communication Theory and Applications, (Ambleside, Lake District, UK)*, pp. 343–348, July 2005.

[80] G. Thomas, "A comparison of motion-compensated interlace-to-progressive conversion methods," *Signal Processing: Image Communication*, vol. 12, pp. 209–229, June 1998.

[81] S. Wenger, "H.264/AVC over IP," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, pp. 645–656, July 2003.

[82] P. Ferre, J. Chung-How, D. Bull, and A. Nix, "Distortion-based link adaptation for wireless video transmission," *EURASIP Journal on Applied Signal Processing*, vol. 2008, pp. 113–130, March 2008.

[83] P. Ferre, D. Agrafiotis, T. K. Chiew, A. R. Nix, and D. R. Bull, "Multimedia transmission over IEEE 802.11g wlans: Practical issues and considerations," *International Conference on Consumer Electronics (ICCE-2007), (Las Vegas, United States)*, pp. 1 –2, January 2007.

[84] D. Marpe, T. Wiegand, and G. J. Sullivan, "The H.264/MPEG4 advanced video coding standard and its applications," *IEEE Communications Magazine*, vol. 44, pp. 134–143, August 2006.

[85] T. Stockhammer, M. M. Hannuksela, and T. Wiegand, "H.264/AVC in wireless environments," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, pp. 657–673, July 2003.

[86] B. Intihar, "Video-conversion techniques ensure a sharper image," *Electronics Design, Strategy, News (EDN), Rogers publishing company Englewood, US*, vol. 43, pp. 97–98, 13 March 1998.

[87] M. Karczewicz and R. Kurceren, "The SP- and SI-frames design for H.264/AVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, pp. 637–644, July 2003.

[88] T. H. Vu and S. Aramvith, "An error resilience technique based on FMO and error propagation for H.264 video coding in error-prone channels," pp. 205 –208, July 2009.

[89] P. Raibroycharoen, M. M. Ghandi, E. V. Jones, and M. Ghanbari, "Performance analysis of H.264/AVC video transmission with unequal error protected turbo codes," *IEEE 61st Vehicular Technology Conference (VTC 2005-Spring)*, vol. 3, pp. 1580–1584, May 2005.

[90] R. Zhang, S. L. Regunathan, and K. Rose, "Video coding with optimal inter/intra-mode switching for packetloss resilience," *IEEE Journal on Selected Areas in Communications*, vol. 18, pp. 966–976, June 2000.

[91] S. Wenger, "Video redundancy coding in H.263+," *International Workshop on Audio-Visual Services, (Aberdeen, UK)*, September 1997.

[92] Y.-K. Wang, M. M. Hannuksela, and M. Gabbouj, *Error Resilient Video Coding Using Unequally Protected Key Pictures*. Springer Berlin / Heidelberg, September 2003.

[93] P. Ferre, D. Agrafiotis, and D. Bull, "Macroblock selection algorithms for error resilient H.264 video wireless transmission using redundant slices," *Proceedings of the International Society for Optical Engineering Visual Communications and Image Processing (SPIE-2008), (San Jose, California, USA)*, vol. 6822, pp. 68220S–68220S–8, 29 to 31 January 2008.

[94] A. Puri, X. Chen, and A. Luthra, "Video coding using the H.264/MPEG-4 AVC compression standard," *Signal Processing: Image Communication (Elsevier Publishers)*, vol. 19, no. 9, pp. 793 – 849, 2004.

[95] S. Saponara, K. Denolf, G. Lafruit, C. Blanch, and J. Bormans, "Performance and complexity coevaluation of the advanced video coding standard for cost-effective multimedia communications," *EURASIP Journal on Applied Signal Processing*, vol. 2004, pp. 220–235, January 2004.

[96] T. Wedi and H. G. Musmann, "Motion- and aliasing-compensated prediction for hybrid video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, pp. 577–586, July 2003.

[97] I. G. Richardson, *H.264 and MPEG-4 Video Compression: Video Coding for Next-generation Multimedia*. Chichester, West Sussex, England: John Wiley & Sons Ltd, October 2003.

[98] T. Wiegand, X. Zhang, and B. Girod, "Long-term memory motion-compensated prediction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, pp. 70–84, February 1999.

[99] T. Wiegand and B. Girod, *Multi-Frame Motion-Compensated Prediction for Video Transmission*. Norwell, Massachusetts, USA: Kluwer Accademy Publishers, September 2001.

[100] B. Girod, "Efficiency analysis of multihypothesis motion-compensatedprediction for video coding," *IEEE Transactions on Image Processing*, vol. 9, pp. 173–183, February 2000.

[101] M. Flierl, T. Wiegand, and B. Girod, "Rate-constrained multi-hypothesis motion-compensated prediction for video coding," in *IEEE 2000 International Conference on Image Processing, (ICIP-2000)*, vol. 3, (Vancouver, BC, Canada), pp. 150–153, 10 to 13 September 2000.

[102] T. W. et al, "Joint draft 5: Scalable video coding," in *Joint Video Team of ISO/IEC MPEG and ITU-T VCEG, Doc. JVT-R201*, (Bangkok, Thailand), January 2006.

[103] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, pp. 1103–1120, September 2007.

[104] N. Ahmed, T. Natara, and K. R. Rao, "Discrete cosine transfom," *IEEE Transactions on Computers*, vol. 23, pp. 90–93, January 1974.

[105] H. S. Malvar, A. Hallapuro, M. Karczewicz, and L. Kerofsky, "Low-complexity transform and quantization in H.264/AVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, pp. 598–603, July 2003.

[106] D. Marpe, H. Schwarz, and T. Wiegand, "Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, pp. 620–636, July 2003.

[107] J. Wen and J. D. Villasenor, "Reversible variable length codes for efficient and robust image and video coding," *IEEE Data Compression Conference (DCC-1998), (Snowbird, UT , USA)*, pp. 471 – 480, 30 March to 01 April 1998.

[108] W. Yu and Y. He, "A high performance CABAC decoding architecture," *IEEE Transactions on Consumer Electronics*, vol. 51, pp. 1352–1359, November 2005.

[109] A. Ortega and K. Ramchandran, "Rate-distortion methods for image and video compression," *IEEE Signal Processing Magazine*, vol. 15, pp. 23–50, November 1998.

[110] G. J. Sullivan and T. Wiegand, "Rate-distortion methods for video compression," *IEEE Signal Processing Magazine*, vol. 15, pp. 74–90, November 1998.

[111] T. Wiegand and B. Girod, "Lagrange multiplier selection in hybrid video coder control," in *International Conference on Image Processing*, vol. 3, (Thessaloniki, Greece), pp. 542–545, 2001.

[112] ITU-T: H.264 Recommendation, "H.264 : Advanced video coding for generic audiovisual services," *ITU-T Recommendation*, p. 282, May 2003.

[113] V. Sgardoni, M. Sarafianou, P. Ferre, A. Nix, and D. Bull, "Robust video broadcasting over 802.11a/g in time-correlated fading channels," *IEEE Transactions on Consumer Electronics*, vol. 55, pp. 69 –76, February 2009.

[114] M. M. Ghandi, B. Barmada, E. V. Jones, and M. Ghanbari, "H.264 layered coded video over wireless networks:. channel coding and modulation constraints.," in *Hindawi Publishing Corporation EURASIP Journal on Applied Signal Processing*, pp. 1–8, February 2006.

[115] N. Thomos, S. Argyropoulos, N. V. Boulgouris, and M. G. Strintzis, "Robust transmission of H.264/AVC video using adaptive slice grouping and unequal error protection," in *IEEE International Conference on Multimedia and Expo (ICME-2006), (Toronto, ON, Canada)*, pp. 593–596, 09 to 12 July 2006.

[116] Y. C. Chang, S. W. Lee, and R. Komiya, "A low-complexity unequal error protection of H.264/AVC video using adaptive hierarchical QAM," *IEEE Transactions on Consumer Electronics*, vol. 52, pp. 1153–1158, November 2006.

[117] M. M. Ghandi and M. Ghanbari, "Layered H.264 video transmission with hierarchical QAM," *Journal of Visual Communication and Image Representation*, vol. 17, no. 2, pp. 451 – 466, 2006.

[118] A. Ksentini, M. Naimi, and A. Gueroui, "Toward an improvement of H.264 video transmission over IEEE 802.11e through a cross-layer architecture," *IEEE Communications Magazine*, vol. 44, pp. 107–114, January 2006.

[119] H.-B. Yu, C. Wang, and S. Yu, "A novel error recovery scheme for H.264 video and its application in conversational services," *IEEE Transactions on Consumer Electronics*, vol. 50, pp. 329–334, February 2004.

[120] Y. Xu and Y. Zhou, "H.264 video communication based refined error concealment schemes," *IEEE Transactions on Consumer Electronics*, vol. 50, pp. 1135–1141, November 2004.

[121] A. Q. Pham, J. Wang, L.-L. Yang, and L. Hanzo, "An iterative detection aided irregular convolutional coded wavelet videophone scheme using reversible variable-length codes and MAP equalization.," in *IEEE Vehicular Technology Conference 65th (VTC-2007 Spring)*, (Dublin), pp. 2404–2408, April 2007.

[122] S. X. Ng, J. Y. Chung, P. Cherriman, and L. Hanzo, "Burst-by-burst adaptive decision feedback equalized TCM, TTCM, and BICM for H.263-assisted wireless video telephony," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, pp. 363–374, March 2006.

[123] Y. Wang and S. Yu, "Joint source-channel decoding for H.264 coded video stream," *IEEE Transactions on Consumer Electronics*, vol. 51, pp. 1273–1276, November 2005.

[124] M. Grangetto, E. Magli, and G. Olmo, "Robust video transmission over error-prone channels via error correcting arithmetic codes," *IEEE Communications Letters*, vol. 7, pp. 596–598, December 2003.

[125] B. Barmada, M. M. Ghandi, E. V. Jones, and M. Ghanbari, "Prioritized transmission of data partitioned H.264 video with hierarchical QAM," *IEEE Signal Processing Letters*, vol. 12, pp. 577–580, August 2005.

[126] Q. Qu, Y. Pei, and J. W. Modestino, "An adaptive motion-based unequal error protection approach for real-time video transport over wireless IP networks," *IEEE Transactions on Multimedia*, vol. 8, pp. 1033–1044, October 2006.

[127] Y. Lee, S. Lee, Y. H. Kim, S. I. Lee, Z.-K. Yim, B. Choi, S. Kim, and J.-S. Seo, "Field trials for terrestrial digital multimedia broadcasting system," *IEEE Transactions on Broadcasting*, vol. 53, pp. 425–433, March 2007.

[128] P. Ferre, A. Doufexi, J. Chung-How, A. R. Nix, and D. R. Bull, "Robust video transmission over wireless LANs," *IEEE Transactions on Vehicular Technology*, vol. 57, pp. 2596 –2602, July 2008.

[129] V. Sgardoni, M. Sarafianou, P. Ferre, A. Nix, and D. Bull, "Robust video broadcasting over 802.11a/g in time-correlated fading channels," *IEEE Transactions on Consumer Electronics*, vol. 55, pp. 69–76, February 2009.

[130] S. Ahmad, R. Hamzaoui, and M. Al-Akaidi, "Adaptive unicast video streaming with rateless codes and feedback," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, pp. 275 –285, February 2010.

[131] R. A. Farrugia and C. J. Debono, "A hybrid error control and artifact detection mechanism for robust decoding of H.264/AVC video sequences," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. PP, pp. 2941 –2945, March 2010.

[132] S. Gao and K.-K. Ma, "Error-resilient H.264/AVC video transmission using two-way decodable variable length data block," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, pp. 340 –350, March 2010.

[133] G. Forney, "Concatenated Codes," *Cambridge: MIT Press*, 1966.

[134] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near shannon limit error-correcting coding and decoding:turbo-codes. 1," in *IEEE International Conference on Communications (ICC-1993)*, vol. 2, (Geneva, Switzerland), pp. 1064–1070, May 1993.

[135] D. Divsalar and F. Pollara, "Multiple turbo codes for deep-space communications," *Telecommunications and Data Acquisition Progress Report, Jet Propulsion Laboratory, (Pasadena, CA)*, pp. 42–121, 15 May 1995. http://hdl.handle.net/2014/3115.

[136] S. Benedetto and G. Montorsi, "Iterative decoding of serially concatenated convolutional codes," *Electronics Letters*, vol. 32, pp. 1186–1188, June 1996.

[137] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "Serial concatenation of interleaved codes: performance analysis, design and iterative decoding," *IEEE Transactions on Information Theory*, vol. 44, pp. 909–926, May 1998.

[138] D. Raphaeli and Y. Zarai, "Combined turbo equalization and turbo decoding," in *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM-1997)*, vol. 2, (Phoenix, AZ), pp. 639–643, November 1997.

[139] D. Raphaeli and Y. Zarai, "Combined turbo equalization and turbo decoding," *IEEE Communications Letters*, vol. 2, pp. 107–109, April 1998.

[140] M. Toegel, W. Pusch, and H. Weinrichter, "Combined serially concatenated codes and turbo-equalization," in *2nd International Symposium on Turbo Codes*, (Brest, France), pp. 375–378, September 2000.

[141] R. Ramamurthy and W. E. Ryan, "Convolutional double accumulate codes (or double turbo DPSK)," *IEEE Communications Letters*, vol. 5, pp. 157–159, April 2001.

[142] C. Douillard, "Iterative correction of intersymbol interference: turbo equalization," *European Transaction on Telecommunications*, vol. 6, pp. 507–511, October 1995.

[143] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "A soft-input soft-output APP module for iterative decoding of concatenated codes," *IEEE Communications Letters*, vol. 1, pp. 22–24, January 1997.

[144] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," in *Proceedings of IEEE International Symposium on Information Theory (ISIT-1997)*, (Ulm, Germany), p. 96, June 1997.

[145] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Transactions on Information Theory*, vol. 44, pp. 927–946, May 1998.

[146] S. ten Brink, J. Speidel, and R.-H. Yan, "Iterative demapping and decoding for multilevel modulation," in *IEEE Global Telecommunications Conference (GLOBCOM-1998)*, vol. 1, (Sydney,NSW), pp. 579–584, 1998.

[147] X. Li and J. A. Ritcey, "Bit-interleaved coded modulation with iterative decoding," *IEEE Communications Letters*, vol. 1, pp. 169–171, November 1997.

[148] X. Li and J. A. Ritcey, "Bit-interleaved coded modulation with iterative decoding using soft feedback," *IEE Electronics Letters*, vol. 34, pp. 942–943, May 1998.

[149] X. Li and J. A. Ritcey, "Trellis-coded modulation with bit interleaving and iterative decoding," *IEEE Journal on Selected Areas in Communications*, vol. 17, pp. 715–724, April 1999.

[150] X. Wang and H. V. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Transactions on Communications*, vol. 47, pp. 1046–1061, July 1999.

[151] A. Sezgin, D. Wuebben, and V. Kuehn, "Analysis of mapping strategies for turbo-coded space-time block codes," in *Proceedings of IEEE Information Theory Workshop*, (Paris, France), pp. 103–106, April 2003.

[152] D. Divsalar, S. Dolinar, and F. Pollara, "Low complexity turbo-like codes," in *2nd International Symposium on Turbo Codes and Related Topics*, (Brest, France), pp. 73–80, September 2000.

[153] D. Divsalar, S. Dolinar, and F. Pollara, "Serial concatenated trellis coded modulation with rate-1 inner code," in *IEEE Global Telecommunications Conference (GLOBECOM-2000)*, vol. 2, (San Francisco, CA), pp. 777–782, 29 November 2000.

[154] K. R. Narayanan, "Effect of precoding on the convergence of turbo equalization for partial response channels," *IEEE Journal on Selected Areas in Communications*, vol. 19, pp. 686–698, April 2001.

[155] I. Lee, "The effect of a precoder on serially concatenated coding systems with an ISI channel," *IEEE Transactions on Communications*, vol. 49, pp. 1168–1175, July 2001.

[156] L. Lifang, D. Divsalar, and S. Dolinar, "Iterative demodulation, demapping and decoding of coded non-square QAM," in *IEEE Transactions on Communications*, vol. 53, pp. 16–19, January 2005.

[157] H. E. Gamal and A. R. Hammons, "Analyzing the turbo decoder using the Gaussian approximation," *IEEE Journal on Selected Areas in Communications*, vol. 47, pp. 671–686, February 2001.

[158] S. ten Brink, "Designing iterative decoding schemes with the extrinsic information transfer chart," in *AEU International Journal of Electronics and Communications*, vol. 54, pp. 389–398, November 2000.

[159] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenatedcodes," *IEEE Transactions on Communications*, vol. 49, pp. 1727–1737, Oct 2001.

[160] M. Tuchler and J. Hagenauer, "EXIT charts of irregular codes," *in Proceedings of the Conference on Information Sciences and Systems, Princeton, NJ, USA*, pp. 748–753, May 2002.

[161] J. Hagenauer, "The EXIT chart - Introduction to extrinsic information transfer in iterative processing," in *European Signal Processing Conference*, (Vienna, Austria), pp. 1541–1548, September 2004.

[162] M. Tüchler, S. ten Brink, and J. Hagenauer, "Measures for tracing convergence of iterative decoding algorithms," in *Proceedings of the 4th International ITG Conference on Source and Channel Coding*, (Berlin, Germany), pp. 53–60, January 2002.

[163] J. Kliewer, N. Goertz, and A. Mertins, "Iterative source-channel decoding with Markov random field source models," *IEEE Transactions on Signal Processing*, vol. 54, pp. 3688–3701, October 2006.

[164] F. Brannstrom, L. K. Rasmussen, and A. J. Grant, "Convergence analysis and optimal scheduling for multiple concatenated codes," *IEEE Transactions on Information Theory*, vol. 51, no. 9, pp. 3354–3364, 2005.

[165] S. ten Brink, "Convergence of multidimensional iterative decoding schemes," in *Conference Record of the Thirty-Fifth Asilomar Conference on Signals, Systems and Computers*, vol. 1, (Pacific Grove, CA, USA), pp. 270–274, 7 August 2002.

[166] M. Tüchler, "Convergence prediction for iterative decoding of threefold concatenated systems," in *IEEE Global Telecommunications Conference (GLOBECOM-2002)*, vol. 2, pp. 1358–1362, November 2002.

[167] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate (corresp.)," *IEEE Transactions on Information Theory*, vol. 20, pp. 284–287, March 1974.

[168] P. Robertson, E. Villebrun, and P. Hoeher, "A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain," in *IEEE International Conference on Communications (ICC-1995)*, vol. 2, (Seattle, WA, USA), pp. 1009–1013, June 1995.

[169] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "Serial concatenation of interleaved codes: performance analysis, design and iterative decoding," *IEEE Transactions on Information Theory*, vol. 44, pp. 909–926, May 1998.

[170] M. Tuchler, "Design of serially concatenated systems depending on the block length," *IEEE Transactions on Communications*, vol. 52, pp. 209–218, February 2004.

[171] M. Adrat, T. Clevorn, J. Brauers, and P. Vary, "Minimum terms of residual redundancy for successful iterative source-channel decoding," *IEEE Communications Letters*, vol. 10, pp. 778 –780, November 2006.

[172] O. R. Alamri, B. L. Yeap, and L. Hanzo, "A turbo detection and sphere-packing-modulation-aided space-time coding scheme," *IEEE Transactions on Vehicular Technology*, vol. 56, pp. 575–582, March 2007.

[173] R. G. Maunder, J. Wang, S. X. Ng, L.-L. Yang, and L. Hanzo, "On the performance and complexity of irregular variable length codes for near-capacity joint source and channel coding," *IEEE Transactions on Wireless Communications*, vol. 7, pp. 1338–1347, April 2008.

[174] O. Alamri, J. Wang, S. X. Ng, L.-L. Yang, and L. Hanzo, "Near-capacity three-stage turbo detection of irregular convolutional coded joint sphere-packing modulation and space-time coding," *IEEE Transactions on Communications*, vol. 57, pp. 1486 –1495, May 2009.

[175] L. Kong, S. X. Ng, R. G. Maunder, and L. Hanzo, "Maximum-throughput irregular distributed space-time code for near-capacity cooperative communications," *IEEE Transactions on Vehicular Technology*, vol. 59, pp. 1511 –1517, march 2010.

[176] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Transactions on Information Theory*, vol. 42, pp. 429–445, March 1996.

[177] S. ten Brink, "Designing iterative decoding schemes with the extrinsic information transfer chart," *AEU International Journal of Electronics and Communications*, vol. 54, pp. 389–398, November 2000.

[178] R. Johannesson and K. S. Zigangirov, *Fundamentals of Convolutional Coding*. Uk: Wiley-IEEE Press, 1999.

[179] M. Tüchler and J. Hagenauer, "EXIT charts of irregular codes," *in Proceedings of the Conference on Information Sciences and Systems (ISS-2002), (Princeton, NJ, USA)*, pp. 748–753, May 2002.

[180] C. E. Shannon, "The mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423, July 1948.

[181] G. D. Forney and D. J. Costello, "Channel Coding: The Road to Channel Capacity," *Proceedings of the IEEE*, vol. 95, pp. 1150–1177, June 2007.

[182] B. J. Frey and D. J. C. MacKay, "Irregular turbo-like codes," *in Proceedings of the International Symposium on Turbo Codes*,, pp. 67–72, September 2000.

[183] M. M. Ghandi, B. Barmada, E. V. Jones, and M. Ghanbari, "Wireless video transmission using feedback-controlled adaptive H.264 source and channel coding," *IET Communications*, vol. 3, pp. 172 –184, February 2009.

[184] T.-C. Wang, H.-C. Fang, W.-M. hao, H.-H. Chen, and L.-G. Chen, "An UVLC encoder architecture for H.26l," in *IEEE International Symposium on Circuits and Systems (ISCAS-2002)*, vol. 2, (Phoenix-Scottsdale, AZ, USA), pp. 308–311, 07 August 2002.

[185] R. G. Maunder, J. Kliewer, S. X. Ng, J. Wang, L.-L. Yang, and L. Hanzo, "Joint iterative decoding of trellis-based VQ and TCM," *IEEE Transactions on Wireless Communications*, vol. 6, pp. 1327–1336, April 2007.

[186] S. X. Ng, J. Y. Chung, and L. Hanzo, "Turbo-detected unequal protection MPEG-4 wireless video telephony using multi-level coding, trellis coded modulation and space-time trellis coding," in *IEE Proceedings - Communications*, vol. 152, pp. 1116–1124, December 2005.

[187] L. Hanzo, C. Somerville, and J. Woodard, *Voice and Audio Compression for Wireless Communications, 2nd Edition.* Wiley-IEEE Press, August 2007.

[188] M. Adrat and P. Vary, "Iterative source-channel decoding: improved system design using EXIT charts," *EURASIP Journal on Applied Signal Processing*, vol. 2005, no. 1, pp. 928–941, 2005.

[189] R. Stedman, H. Gharavi, L. Hanzo, and R. Steele, "Transmission of subband-coded images via mobile channels," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 3, pp. 15–26, February 1993.

[190] L. Hanzo and J. Streit, "Adaptive low-rate wireless videophone schemes," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 5, pp. 305–318, August 1995.

[191] J. Streit and L. Hanzo, "Dual-mode vector-quantized low-rate cordless videophone systems forindoors and outdoors applications," *IEEE Transactions on Vehicular Technology*, vol. 46, pp. 340–357, May 1997.

[192] J. Streit and L. Hanzo, "Quadtree-based reconfigurable cordless videophone systems," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, pp. 225–237, April 1996.

[193] L. Hanzo, P. Cherriman, and E.-L. Kuan, "Interactive cellular and cordless video telephony: State-of-the-art system design principles and expected performance," *Proceedings of the IEEE*, vol. 88, pp. 1388–1413, September, 2000.

[194] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: turbo-codes," *IEEE Transactions on Communications*, vol. 44, pp. 1261–1271, October 1996.

[195] T. Fingscheidt and P. Vary, "Softbit speech decoding: a new approach to error concealment," *IEEE Transactions on Speech and Audio Processing*, vol. 9, pp. 240–251, 07 August 2001.

[196] T. Fingscheidt, S. Heinen, and P. Vary, "Joint speech codec parameter and channel decoding of parameter individual block codes (PIBC)," in *IEEE Workshop on Speech Coding Proceedings*, (Porvoo, Finland), pp. 75–77, 06 August 2002.

[197] R. Perkert, M. Kaindl, and T. Hindelang, "Iterative source and channel decoding for GSM," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP-2001)*, vol. 4, (Salt Lake City, UT, USA), pp. 2649–2652, 07 to 11 May 2001.

[198] A. Guyader, E. Fabre, C. Guillemot, and M. Robert, "Joint source-channel turbo decoding of entropy-coded sources," *IEEE Journal on Selected Areas in Communications*, vol. 19, pp. 1680–1696, September 2001.

[199] J. Kliewer and R. Thobaben, "Iterative joint source-channel decoding of variable-length codes using residual source redundancy," *IEEE Transactions on Wireless Communications*, vol. 4, pp. 919–929, May 2005.

[200] J. Liu, G. Tu, C. Zhang, and Y. Yang, "Joint source and channel decoding for variable length encoded turbo codes," *EURASIP Journal on Applied Signal Processing*, vol. 2008, pp. 1–10, January 2008.

[201] P. J. Cherriman, T. Keller, and L. Hanzo, "Subband-adaptive turbo-coded OFDM-based interactive video telephony," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, pp. 829–839, October 2002.

[202] L. Hanzo, P. Cherriman, and E.-L. Kuan, "Interactive cellular and cordless video telephony: State-of-the-art system design principles and expected performance," *Proceedings of the IEEE*, vol. 88, pp. 1388–1413, September 2000.

[203] L. Hanzo, J. Blogh, and S. Ni, *3G, HSPA and FDD versus TDD Networking: Smart Antennas and Adaptive Modulation, 2nd Edition*. Wiley-IEEE Press, February 2008.

[204] T. Clevorn, P. Vary, and M. Adrat, "Iterative source-channel decoding using short block codes," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-2006)*, vol. 4, pp. 221–224, May 2006.

[205] R. Thobaben, "A new transmitter concept for iteratively-decoded source-channel coding schemes," in *IEEE 8th Workshop on Signal Processing Advances in Wireless Communications (SPAWC-2007)*, pp. 1–5, June 2007.

[206] R. Thobaben, T. Schmalen, and P. Vary, "Joint source-channel coding with inner irregular codes," in *IEEE International Symposium on Information Theory (ISIT-2008)*, pp. 1153–1157, July 2008.

[207] M. Adrat, P. Vary, and T. Clevorn, "Optimized bit rate allocation for iterative source-channel decoding and its extension towards multi-mode transmission," in *in Proceedings of IST Mobile and Wireless Communications Summit (Dresden, Germany)*, pp. 1153–1157, June 2005.

[208] T. Clevorn, J. Brauers, M. Adrat, and P. Vary, "Turbo decodulation: iterative combined demodulation and source-channel decoding," *IEEE Communications Letters*, vol. 9, pp. 820–822, September 2005.

[209] T. Clevorn, J. Braucrs, M. Adrat, and P. Vary, "EXIT chart analysis of turbo decodulation," in *IEEE 16th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC-2005)*, vol. 2, pp. 711–715, September 2005.

[210] T. Clevorn, M. Adrat, and P. Vary, "Turbo decodulation using highly redundant index assignments and multi-dimensional mappings," *Proceedings of International Symposium on Turbo Codes & Related Topics, (Munich, Germany)*, pp. 209–215, April 2006.

[211] T. Clevorn, L. Schmalen, P. Vary, and M. Adrat, "On Redundant Index Assignments for Iterative Source-channel Decoding," *IEEE Communications Letters*, vol. 12, pp. 514–516, July, 2008.

[212] E. Steinbach, N. Farber, and B. Girod, "Standard compatible extension of H.263 for robust video transmission in mobile environments," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, pp. 872 –881, December 1997.

[213] C.-S. Kim, R.-C. Kim, and S.-U. Lee, "An error detection and recovery algorithm for compressed video signal using source level redundancy," *IEEE Transactions on Image Processing*, vol. 9, pp. 209 –219, February 2000.

[214] C.-S. Kim and S.-U. Lee, "Multiple description coding of motion fields for robust video transmission," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, pp. 999–1010, September 2001.

[215] T. Fang and L.-P. Chau, "Content-based resynchronization for robust video transmission," *IEEE Transactions on Broadcasting*, vol. 50, pp. 390–395, December 2004.

[216] C.-S. Kim, J. Kim, I. Katsavounidis, and C.-C. J. Kuo, "Robust MMSE video decoding: theory and practical implementations," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, pp. 39–51, January 2005.

[217] R. Puri, A. Majumdar, and K. Ramchandran, "PRISM: A video coding paradigm with motion estimation at the decoder," *IEEE Transactions on Image Processing*, vol. 16, pp. 2436 –2448, October 2007.

[218] J. Wang, A. Majumdar, and K. Ramchandran, "Robust video transmission with distributed source coded auxiliary channel," *IEEE Transactions on Image Processing*, vol. 18, pp. 2695 –2705, December 2009.

[219] M. H. Lu, P. Steenkiste, and T. Chen, "Robust wireless video streaming using hybrid spatial/temporal retransmission," *IEEE Journal on Selected Areas in Communications*, vol. 28, pp. 476 –487, April 2010.

[220] M. Adrat, P. Vary, and J. Spittka, "Iterative source-channel decoder using extrinsic information from softbit-source decoding," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, (Salt Lake City, UT, USA), pp. 2653–2656, May 2001.

[221] M. Adrat, U. V. Agris, and P. Vary, "Convergence behavior of iterative source-channel decoding," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP-2003)*, vol. 4, pp. 269–272, April 2003.

[222] M. Adrat, J. Brauers, T. Clevorn, and P. Vary, "The EXIT-characteristic of softbit-source decoders," *IEEE Communications Letters*, vol. 9, pp. 540–542, June 2005.

[223] M. Adrat, J. Picard, and P. Vary, "Analysis of extrinsic information from softbit-source decoding applicable to iterative source-channel decoding," in *Proceedings of International Conference on Source and Channel Coding (Berlin, Germany)*, pp. 269–275, January 2002.

[224] N. Görtz, "Analysis and performance of iterative source-channel decoding," *Proceedings 2nd International Symposium on Turbo Codes & Related Topics, Brest, France,*, pp. 251–254, September 2000.

[225] N. Görtz, "On the iterative approximation of optimal joint source-channel decoding," *IEEE Journal on Selected Areas in Communications*, vol. 19, pp. 1662–1670, September 2001.

[226] P. Robertson, P. Hoeher, and E. Villebrun, "Optimal and sub-optimal maximum a posteriori algorithms suitable for turbo decoding," *European Transactions on Telecommunication*, vol. 8, pp. 119–125, march 1997.

[227] A. Q. Pham, L.-L. Yang, and L. Hanzo, "Joint optimization of iterative source and channel decoding using over-complete source mapping.," in *IEEE 65th Vehicular Technology Conference (VTC-2007 Spring)*, (Dublin), pp. 2404–2408, April 2007.

[228] T. Hindelang, M. Adrat, T. Fingscheidt, and S. Heinen, "Joint source and channel coding: from the beginning until the 'EXIT'," *European Transactions on Telecommunications*, vol. 18, no. 8, pp. 851–858, 2007.

[229] S. M. Alamouti, "A simple transmit diversity technique for wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 1451–1458, October 1998.

[230] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communication: performance criterion and code construction," *IEEE Transactions on Information Theory*, vol. 44, pp. 744–765, March 1998.

[231] B. Hochwald, T. L. Marzetta, and C. B. Papadias, "A transmitter diversity scheme for wideband CDMA systems based onspace-time spreading," *IEEE Journal on Selected Areas in Communications*, vol. 19, pp. 48–60, January 2001.

[232] L. Hanzo, O. Alamri, M. El-Hajjar, and N. Wu, *Near-Capacity Multi-Functional MIMO Systems*. New York, USA: John Wiley & Sons, 2009.

[233] D. G. Brennan, "Linear diversity combining techniques," in *Proceedings of the IRE*, vol. 47, pp. 1075–1102, June 1959.

[234] A. Wittneben, "Base station modulation diversity for digital simulcast," in *41st IEEE Vehicular Technology Conference on Gateway to the Future Technology in Motion*, (St. Louis, MO), pp. 848–853, May 1991.

[235] A. Wittneben, "A new bandwidth efficient transmit antenna modulation diversity scheme for linear digital modulation," in *IEEE International Conference on Communications (ICC-1993)*, vol. 3, (Geneva), pp. 1630–1634, May 1993.

[236] N. Seshadri and J. H. Winters, "Two signaling schemes for improving the error performance of frequency-division-duplex transmission systems using transmitter antenna diversity," in *IEEE Vehicular Technology Conference (VTC-1993 Spring)*, (Secaucus, NJ), pp. 508–511, April 1993.

[237] J. H. Winters, "The diversity gain of transmit diversity in wireless systems with Rayleigh fading," in *IEEE International Conference on Communications (ICC-1994)*, (New Orleans, LA), pp. 1121–1125, May 1994.

[238] T. Eng, N. Kong, and L. Milstein, "Comparison of diversity combining techniques for Rayleigh-fading channels," *IEEE Transactions on Communications*, vol. 44, pp. 1117–1129, September 1996.

[239] V. Tarokh, H. Jafarkhani, and A. R. Calderbank, "Space-time block codes from orthogonal designs," *IEEE Transactions on Information Theory*, vol. 45, pp. 1456–1467, July 1999.

[240] V. Tarokh, H. Jafarkhani, and A. R. Calderbank, "Space-time block coding for wireless communications: performance results," *IEEE Journal on Selected Areas in Communications*, vol. 17, pp. 451–460, March 1999.

[241] J.-C. Guey, M. P. Fitz, M. R. Bell, and W.-Y. Kuo, "Signal design for transmitter diversity wireless communication systems over Rayleigh fading channels," *IEEE Transactions on Communications*, vol. 47, pp. 527–537, April 1999.

[242] H. Jafarkhani, "A quasi-orthogonal space-time block code," *IEEE Transactions on Communications*, vol. 49, pp. 1–4, January 2001.

[243] B. Hassibi and B. M. Hochwald, "High-rate codes that are linear in space and time," *IEEE Transactions on Information Theory*, vol. 48, pp. 1804–1824, July 2002.

[244] P. Stoica and G. Ganesan, "Space-time block codes: trained, blind and semi-blind detection," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 2, (Orlando, FL), pp. 1609–1612, November 2002.

[245] H. Wang and X.-G. Xia, "Upper bounds of rates of complex orthogonal space-time block codes," *IEEE Transactions on Information Theory*, vol. 49, pp. 2788–2796, October 2003.

[246] W. Su, Z. Safar, and K. J. R. Liu, "Space-time signal design for time-correlated Rayleigh fading channels," in *IEEE International Conference on Communications (ICC-2003)*, vol. 5, pp. 3175–3179, May 2003.

[247] H. Zhang and T. A. Gulliver, "Capacity and error probability analysis for orthogonal space-time block codes over fading channels," *IEEE Transactions on Wireless Communications*, vol. 4, pp. 808–819, March 2005.

[248] T.-H. Liew and L. Hanzo, "Space-time trellis and space-time block coding versus adaptive modulation and coding aided OFDM for wideband channels," *IEEE Transactions on Vehicular Technology*, vol. 55, pp. 173–187, January 2006.

[249] P. Luo and H. Leib, "Class of full-rank space-time codes combining orthogonal designs with delay diversity," *IEEE Transactions on Vehicular Technology*, vol. 57, pp. 260–272, January 2008.

[250] M. El-Hajjar, O. Alamri, S. X. Ng, and L. Hanzo, "Turbo Detection of Precoded Sphere Packing Modulation Using Four Transmit Antennas for Differential Space-time Spreading," *IEEE Transactions on Wireless Communications*, vol. 7, pp. 943–952, March 2008.

[251] J. H. Conway, N. J. Sloane, and E. Bannai, *Sphere packings, lattices, and groups.* New York, USA: Springer-Verlag, September 1999.

[252] M. El-Hajjar, O. Alamri, and L. Hanzo, "Differential space-time spreading using iteratively detected sphere packing modulation and two transmit antennas," in *IEEE Wireless Communications and Networking Conference (WCNC-2006)*, vol. 3, (Las Vegas, NV, USA), pp. 1664–1668, 3 to 6 April 2006.

[253] L. Hanzo, L.-L. Yang, E.-L. Kuan, and K. Yen, *Single and multi-carrier DS-CDMA: Multi-user detection, space-time spreading, synchronisation, networking and standards.* Chichester, England:: John Wiley & Sons IEEE Press, 2003.

[254] N. S. Othman, M. El-Hajjar, O. Alamri, and L. Hanzo, "Soft-bit assisted iterative AMR-WB source-decoding and turbo-detection of channel-coded differential space-time spreading using sphere packing modulation," in *IEEE 65th Vehicular Technology Conference (VTC-2007 Spring)*, (Dublin), pp. 2010–2014, April 2007.

[255] S. ten Brink, J. Speidel, and R. H. Yan, "Iterative demapping and decoding for multilevel modulation," *IEEE Global Telecommunications Conference GLOBECOM*, vol. 1, pp. 579–584, 1998.

[256] D. W. Redmill, D. R. Bull, and C. N. Canagarajah, "Absolute value coding for robust and scalable video coding," *IEEE Electronics Letters*, vol. 43, pp. 1074 –1075, September 2007.

[257] N. Thomas, D. Bull, and D. Redmill, "A novel H.264 SVC encryption scheme for secure bit-rate transcoding," *Picture Coding Symposium (PCS-2009), (Chicago, illinois, USA)*, pp. 1 –4, 6 to 8 May 2009.

[258] R. Atta, R. Rizk, and M. Ghanbari, "Motion-compensated DCT temporal filters for efficient spatio-temporal scalable video coding," *Image Communication*, vol. 24, pp. 702–717, October 2009.

[259] L. Hanzo and B.-J. Choi, "Near-instantaneously adaptive hsdpa-style ofdm versus mc-cdma transceivers for wifi, wimax, and next-generation cellular systems," *Proceedings of the IEEE*, vol. 95, pp. 2368 – 2392, December 2007.

[260] P. Cherriman, T. Keller, and L. Hanzo, "Orthogonal frequency-division multiplex transmission of H.263 encoded video over highly frequency-selective wireless networks," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, pp. 701 –712, August 1999.

[261] L.-L. Yang and L. Hanzo, "Performance of fractionally spread multicarrier cdma in awgn as well as slow and fast nakagami-m fading channels," *IEEE Transactions on Vehicular Technology*, vol. 54, pp. 1817 – 1827, September 2005.

[262] A. S. Akbari, C. N. Canagarajah, D. Redmill, and D. Bull, "Disparity compensated view filtering wavelet based multiview image code using lagrangian optimization," *3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video*, pp. 309 –312, May 2008.

[263] A. S. Akbari, C. N. Canagarajah, D. Redmill, and D. Agrafiotis, "A novel H.264/AVC based multi-view video coding scheme," *3DTV Conference, 2007*, pp. 1 –4, May 2007.

[264] S. Adedoyin, W. A. C. Fernando, and A. Aggoun, "A joint motion disparity motion estimation technique for 3d integral video compression using evolutionary strategy," *IEEE Transactions on Consumer Electronics*, vol. 53, pp. 732 –739, May 2007.

[265] S. L. P. Yasakethu, C. Hewage, W. Fernando, and A. Kondoz, "Quality analysis for 3d video using 2D video quality models," *IEEE Transactions on Consumer Electronics*, vol. 54, pp. 1969 –1976, November 2008.

[266] S. L. P. Yasakethu, W. A. C. Fernando, B. Kamolrat, and A. Kondoz, "Analyzing perceptual attributes of 3d video," *IEEE Transactions on Consumer Electronics*, vol. 55, pp. 864 –872, May 2009.

[267] D.-X. Li, W. Zheng, X.-H. Xie, and M. Zhang, "Optimising inter-view prediction structure for multiview video coding with minimum spanning tree," *IEEE Electronics Letters*, vol. 43, 8 2007.

[268] S. Veres and D. Ionescu, "A performance model and measurement framework for diffserv implementations," *IEEE Transactions on Instrumentation and Measurement*, vol. 56, pp. 1473 –1480, August 2007.

[269] D. J. C. MacKay, "Fountain codes," *IEE Proceedings - Communications*, vol. 152, pp. 1062 – 1068, December 2005.

[270] X. Yuan and L. Ping, "Quasi-systematic doped LT codes," *IEEE Journal on Selected Areas in Communications*, vol. 27, pp. 866 –875, August 2009.

[271] M. Cominetti, V. Mignone, A. Morello, and M. Visintin, "The European system for digital multiprogramme television by satellite," *IEEE Transactions on Broadcasting*, vol. 41, pp. 49 –62, June 1995.

[272] J. S. Choi, J. W. Kim, D. S. Han, J. Y. Nam, and Y. H. Ha, "Design and implementation of dvb-t receiver system for digital tv," *IEEE Transactions on Consumer Electronics*, vol. 50, pp. 991 – 998, November 2004.

[273] J. Lei and W. Gao, "A backward-compatible solution for next generation DVB-C system," *IEEE International Conference on Communications (ICC-2008), (Beijing, China)*, pp. 1962 –1966, May 2008.

[274] C.-C. Hsieh, C.-H. Lin, and W.-T. Chang, "Design and implementation of the interactive multimedia broadcasting services in DVB-H," *IEEE Transactions on Consumer Electronics*, vol. 55, pp. 1779 – 1787, November 2009.

[275] G. Faria, J. A. Henriksson, E. Stare, and P. Talmola, "DVB-H: digital broadcast services to handheld devices," *Proceedings of the IEEE*, vol. 94, pp. 194–209, January 2006.

[276] C. A. Booker and M. E. Clark, "Cooperative study of power system communication results in improved facilities," *Transactions of the American Institute of Electrical Engineers*, vol. 50, pp. 1188 –1193, December 1931.

[277] R. Zhang and L. Hanzo, "Interleaved random space time coding for multisource cooperation," *IEEE Transactions on Vehicular Technology*, vol. 58, pp. 2120 –2125, May 2009.

[278] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Transactions on Information Theory*, vol. 19, pp. 471 – 480, July 1973.

[279] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Transactions on Information Theory*, vol. 22, pp. 1 – 10, January 1976.

[280] "Generic coding of moving pictures and associated audio information- part2: Video," in *ITU-T Rec. H.262 and ISO/IEC 13818-2 (MPEG-2 Video), ITU-T and ISO/IEC JTC*, November 1994.

[281] "Video coding for low bit-rate communication," in *ITU-T Rec. H.263 ITU-T,Version 3*, November 2000.

[282] R. Puri and K. Ramchandran, "PRISM: a "reversed" multimedia coding paradigm," *International Conference on Image Processing (ICIP-2003), (Barcelona, Spain)*, vol. 1, pp. I – 617–20 vol.1, September 2003.

[283] R. Puri and K. Ramchandran, "PRISM: an uplink-friendly multimedia coding paradigm," *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP-2003), (Hong Kong)*, vol. 4, pp. IV – 856–9 vol.4, April 2003.

[284] N. Anantrasirichai, D. Agrafiotis, and D. Bull, "A concealment based approach to distributed video coding," *IEEE 15th International Conference on Image Processing (ICIP-2008), (California, USA)*, pp. 2232 –2235, October 2008.

[285] N. Anantrasirichai, D. Agrafiotis, and D. Bull, "Distributed video coding for wireless multi-camera networks," *5th International Conference on Visual Information Engineering (VIE-2008), (Xi'an, China)*, pp. 67 –70, 29 August 2008.

[286] P. Ferre, D. Agrafiotis, and D. Bull, "Fusion methods for side information generation in multi-view distributed video coding systems," *IEEE International Conference on Image Processing (ICIP-2007), (San Antonio, Texas)*, vol. 6, pp. VI –409 –VI –412, October 2007.

Glossary

ACs	Arithmetic Codes
AGM	Anti-Gray Mapping
AOFDM	Adaptive Orthogonal Frequency Division Multiplex
APP	*A Posteriori* Probability
ASO	Arbitrary Slice Ordering
AWGN	Additive White Gaussian Noise
B	Bidirectional
BCJR	Bahl-Cocke-Jelinek-Raviv
BER	Bit Error Ratio
BICM-ID	Bit-Interleaved Coded Modulation using Iterative Decoding
BMA	Block Matching Algorithms
CABAC	Context-Adaptive Binary Arithmetic Coding
CAVLC	Context-Adaptive Variable Length Coding
CBP	Coded Block Patterns
CCF	Cross Correlation Function
CCIR	Consultative Committee for International Radio
CDMA	Code Division Multiple Access
CIF	Common Intermediate Format
CSA	Cross Search Algorithm
DCT	Discrete Cosine Transform
DiffServ	Differentiated Services
DP	Data-Partitioning
DPCM	Differential Pulse Code Modulation
DSTS	Differential Space Time Spreading
DVB	Digital Video Broadcast

DVB-C DVB over cable
DVB-S DVB over satellites
DVB-T DVB over Terrestrial
DVDs Digital Versatile Discs

EEP Equal Error Protection
EOSBC EXIT-chart Optimised Short Block Codes
EXIT Extrinsic Information Transfer

FIR Finite Impulse Response
FMO Flexible Macro-block Ordering
fps frames-per-second
FUMCTI Fast Unidirectional Motion Compensation Temporal Interpolation

GOP Group of Pictures

HD High Definition
HDTV High Definition Television
HQAM Hierarchical QAM
HSPA High-Speed Packet Access

I Intra-coded
IDR Instantaneous Decoding Refresh
IP Internet Protocol
IRCC IrRegular Convolutional Codes
IrVLC IrRegular Variable Length Coding
IrVLC Irregular Variable Length Coding
ISCD Iterative Source-Channel Decoding
ISO International Standarization Organisation
ITU-T International Telecommunications Union - Telecommunication

JPEG Joint Photographic Experts Group
JSCD Joint Source-Channel Decoding
JVT Joint Video Team

LDSP Large Diamond Search Pattern
LLR Log-Likelihood Ratio
LT Luby Transform

MAC Medium Access Control
MAD Mean Absolute Difference
MAE Mean Absolute Error
MAP Maximum A-Posteriori
MAP Maximum A-posteriori
MB Macro-Block
MBs Macro-Blocks
MCER Motion Compensated Error Residual
MCP Motion Compensated Prediction
MCTI Motion Compensated Temporal Interpolation
MDMC Multiple Description Motion Coding
MI Mutual Information
MIMO Multiple-Input Multiple-Output
MMS Multimedia Messaging Services
MMSE Minimum Mean Square Error
MOS Mean Opinion Score
MPEG Motion Picture Experts Group
MPEG-1 Motion Picture Experts Group type 1
MPEG-2 Motion Picture Experts Group type 2
MSE Mean Squared Error
MVC Multi View Coding
MVs Motion-Vectors

NAL Network Abstraction Layer
NALUs Network Abstraction Layer Units
NUQ Non-uniform Quantisers

OMCTI Overlapped Motion Compensation Temporal Interpolation

P Predicted
PPM Portable Pixel Map
PSC Picture Start Code
PSNR Peak Signal-to-Noise Ratio
PSS Packet-Switched Streaming Services
PSTN Public Switched Telephone Networks

QAM Quadrature Amplitude Modulation
QCIF Quarter Common Intermediate Format
QP Quantisation Parameter

QPSK Quadrature Phase Shift Keying
QSIF Quarter Source Input Format

RGB Red-Green-Blue
RNSC Recursive Non-Systematic Convolutional
RSC Recursive Systematic Convolutional
RSM Redundant Source Mapping
RTP Real Time Protocol
RVLC Reversible Variable-Length Codes

SBC Short Block Coding
SBCs Short Block Codes
SD Standard Definition
SDSP Small Diamond Search Pattern
SDTV Standard Digital TV
SES Slice Error Sensitivity
SI Switching I
SIF Source Input Format
SISO Soft-Input Soft-Output
SNR Signal to Noise Ratio
SP Sphere Packing
STBC Space-Time Block Codes
STS Space-Time Spreading
SVC Scalable Video Coding

T-DMB Terrestrial Digital Multimedia Broadcasting
TCM Trellis-Coded Modulation
TCP Transmission Control Protocol
TDL Two-dimensional Logarithmic
TSS Three-Step Search
TTCM Turbo Trellis-Coded Modulation

UDP User Datagram Protocol
UEP Unequal Error Protection
UMCTI Unidirectional Motion Compensated Temporal Interpolation
UMTS Universal Mobile Telecommunications System
UQ Uniform Quantisers
URC Unity Rate Code
URQ Uniform Reconstruction Quantiser

USSPA Unequal Source-Symbol Probability Aided
UVLC Universal Variable Length Coding

VBR Variable Bit Rate
VCEG Video Coding Experts Group
VCL Video Coding Layer
VCRs Video Cassette Recorders
VLC Variable Length Coding
VLD Variable Length Decoded
VQ Vector Quantisation
VRC Video Redundancy Coding

WLANs Wireless Local Area Networks

xDSL Digital Subscriber Loops

Subject Index

U

V

W

X

Author Index

C

D

E

F

G

H

I

J

K

L

M

T

V

W

X

Y

Z