DESIGN TECHNIQUES FOR GRAPH-BASED ERROR-CORRECTING CODES AND THEIR APPLICATIONS

A Dissertation

by

CHING FU LAN

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

December 2004

Major Subject: Electrical Engineering

DESIGN TECHNIQUES FOR GRAPH-BASED ERROR-CORRECTING CODES AND THEIR APPLICATIONS

A Dissertation

by

CHING FU LAN

Submitted to Texas A&M University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Approved as to style and content by:

Krishna R. Narayanan (Co-Chair of Committee)	Zixiang Xiong (Co-Chair of Committee)
Costas N. Georghiades (Member)	Chin B. Su (Member)
Jyh-Charn Liu (Member)	Chanan Singh (Head of Department)

December 2004

Major Subject: Electrical Engineering

ABSTRACT

Design Techniques for Graph-based Error-correcting Codes

and Their Applications. (December 2004)

Ching Fu Lan, B.S., National Central University;

M.S., National Chiao-Tung University Taiwan

Co–Chairs of Advisory Committee: Dr. Krishna R. Narayanan
Dr. Zixiang Xiong

In Shannon's seminal paper, "A Mathematical Theory of Communication", he defined "Channel Capacity" which predicted the ultimate performance that transmission systems can achieve and suggested that capacity is achievable by error-correcting (channel) coding. The main idea of error-correcting codes is to add redundancy to the information to be transmitted so that the receiver can explore the correlation between transmitted information and redundancy and correct or detect errors caused by channels afterward. The discovery of turbo codes and rediscovery of Low Density Parity Check codes (LDPC) have revived the research in channel coding with novel ideas and techniques on code concatenation, iterative decoding, graph-based construction and design based on density evolution. This dissertation focuses on the design aspect of graph-based channel codes such as LDPC and Irregular Repeat Accumulate (IRA) codes via density evolution, and use the technique (density evolution) to design IRA codes for scalable image/video communication and LDPC codes for distributed source coding, which can be considered as a channel coding problem.

The first part of the dissertation includes design and analysis of rate-compatible IRA codes for scalable image transmission systems. This part presents the analysis with density evolution the effect of puncturing applied to IRA codes and the asymptotic analysis of the performance of the systems.

In the second part of the dissertation, we consider designing source-optimized IRA codes. The idea is to take advantage of the capability of Unequal Error Protection (UEP) of IRA codes against errors because of their irregularities. In video and image transmission systems, the performance is measured by Peak Signal to Noise Ratio (PSNR). We propose an approach to design IRA codes optimized for such a criterion.

In the third part of the dissertation, we investigate Slepian-Wolf coding problem using LDPC codes. The problems to be addressed include coding problem involving multiple sources and non-binary sources, and coding using multi-level codes and non-binary codes.

To my family.

ACKNOWLEDGMENTS

This work never have been possible without the generous support and help, and the inspiring guidance I have been given during my stay at Texas A&M University as a Ph.D. student. In addition to that, it is a privilege for me to work with the best people who are constantly coming up with innovations. Here I would like to acknowledge them with my gratitude and respect.

My heartfelt gratitude first goes to my advisors, Prof. Krishna Narayanan and Prof. Zixiang Xiong. It was Prof. Narayanan's lectures on channel coding theory that enlightened me with his approaches to the gist of the discipline and helped me build the working knowledge for the research. It was Prof. Xiong's patient guidance and inspiration that kept me motivated and intellectually challenged. During this tournament, I grew personally and professionally and shaped my career in the future with their constant support, help and guidance. I would also like to thank Prof. Costas Georghiades, Prof. Chin B. Su and Prof. Jyh-Charn Liu and for serving on my defense committees and for providing valuable suggestions to improve this dissertation. I am also grateful to all the current and former colleagues at Multimedia Lab and Wireless Communications Lab, for making my stay at Texas A&M University a pleasant one.

This dissertation is dedicated to my parents and Yu-Kwun. Their understanding and unlimited support gave me the courage to pursue my goal. My love for them shall not be put in words.

TABLE OF CONTENTS

CHAPTER Page

Page

LIST OF TABLES

LIST OF FIGURES

FIGURE Page

FIGURE Page

CHAPTER I

INTRODUCTION

In his seminal paper [1], "A Mathematical Theory of Communication", Shannon defined the capacity of a communication channel which predicted the rates at which transmission systems can transmit and receive information reliably over a noisy channel. Shannon suggested that the capacity is achievable with good channel codes. A channel code is good in a sense that the decoders of the code at rate slightly smaller than the channel capacity is error-free asymptotically. In his proof, Shannon used random coding to achieve the capacity. However, from an engineering perspective, random coding is too complex to implement because random codes lack structures. For fifty years, researchers have been searching for practical capacity-achieving error-correcting codes. Recently, with a reasonable complexity, LDPC and related codes have been shown to perform only several tenths of a dB away from the capacity. In the same paper, Shannon also presented a theorem stating that a source with entropy H can be reliably transmitted over a channel with capacity C as long as $H < C$. In principle this can be achieved by first applying a source code that reduces the rate of the source down to its entropy and subsequently applying a channel code. In the receiver, the channel decoder, being unaware of the type of source, outputs the most probable codeword to source decoder. Finally, the source decoder reconstructs the source without the knowledge of the channel statistics. The principle, which greatly simplifies the complexity in communication system design, is known as the separation principle [2]. However, the separation principle is an asymptotic result. The other problem with the separation principle is that, one has to tolerate infinite delay and

---

This dissertation follows the style of *IEEE Transactions on Communications*.

complexity. Therefore, in practical multi-media communication systems with delay and complexity constraints, the joint source-channel coding (JSCC) strategy should be implemented to provide acceptable performance. In the following two chapters, we will present design techniques for graph-based error-correcting codes with application to the JSCC problems.

The other application of error-correcting codes is distributed source coding or Slepian-Wolf coding. Driven by the severe physical constraints in sensor networks, there has been great interest in developing distributed signal processing algorithms to leverage the limited power in sensors to achieve maximal system performances. One such example is lossless distributed data compression, which was first considered by Slepian and Wolf [3]. The Slepian-Wolf theorem determines the region of achievable rates for compressing two correlated and physically separated sources, and reconstructing two sources jointly at the decoder. In the case of asymmetric coding, one source can be compressed down to its entropy while the other is encoded at the rate equal to the conditional entropy. Since two sources are correlated, the former sources (after decompressed) can be viewed as channel output and the latter source serves as the side information. Therefore, the Slepian-Wolf coding problem can be considered as a channel coding problem. Symmetric coding is achievable via time sharing.

The main idea of error-correcting codes is to add redundancy that is correlated to the information to be transmitted so that the receiver can exploit the correlation between the information bits and the redundancy bits and then correct or detect errors caused by channels. There are two major classes of codes, namely, block codes and convolutional codes. Examples of block codes are Hamming codes, Bose-Chaudhuri-Hocquenghem (BCH) codes, Reed-Solomon (RS) codes (RS) [4] and newly rediscovered LDPC codes. Block codes like Hamming, BCH and RS codes codes have nice mathematical structures. However, there is a limitation when it comes to code

lengths. A bounded-distance decoding algorithm is usually employed in decoding block codes and, except LDPC codes, it is generally hard to use soft decision decoding for block codes.

Convolutional codes are represented with finite state machines, in which going from a start state to a next state is called a state transition. A sequence of state transitions is called trellis path and a sequence of allowed state transitions constitutes a valid trellis path. The decoder can output the most probable valid trellis in the maximum likelihood (ML) or maximum *a posteriori* (MAP) sense based on received signal. The decoding algorithms can be based on hard-decision data or soft-decision data and produce soft output. Soft-output decoding algorithms are especially useful in iterative decoding, in which a concatenated coding system is used and the soft output of the decoder of a component code can be further processed by the decoder of the other component code without loss of information due to quantization.

Both codes provide good coding gain but there is still a gap from capacity. More coding gain can be obtained by using code concatenation. Concatenation is a scheme first introduced by Forney [5] in which two codes, an inner code and an outer code, are used in a serial cascade fashion. It is possible for this concatenated coding system that decoders have the probability of error decreasing exponentially while having decoding complexity increasing only algebraically. Later, concatenation is a key ingredient in turbo codes which have performance very close to the Shannon limit. First appearing in a paper by Berrou, Glavieux and Thitimajshima [6], turbo codes can achieve performance very close to the Shannon limit by combining parallel concatenation, proper choice of component codes and an inter-leaver and an iterative decoding algorithm. The iterative decoding (processing) techniques also find their applications in a number of related areas such as multi-user detection, equalization and synchronization [7, 8, 9]. Prompted by their impressively near-capacity performance, serial concatenated con-

volutional codes (SCCC) [10], hybrid concatenated convolutional codes (HCCC) [11], and other concatenated schemes are proposed and shown to provide similar coding gain. This landmark paper has sprung a new exciting research area in coding theory, which led to the rediscovery of Low Density Parity Check (LDPC) codes. LDPC codes was discovered by Gallager [12] and later rediscovered by Mackay [13]. LDPC codes be described using bipartite (Tanner) graphs which contain variable and check nodes connected by edges. For binary codes, the modulo-2 sum of binary values conveyed on the edges connected to a check must equal zero. When all the variable or check nodes have the same number of edges, the graphs are said to be regular. In [14, 15, 16, 17], an irregular graph is used to optimized the performance. Richardson discovered that LDPC codes exhibit a threshold phenomenon and the threshold can be computed with a numerical technique called density evolution. The graph interpretations lay the foundation for other class of codes like regular/Irregular Repeat Accumulate (RA/IRA) codes [18, 19] and product accumulate codes [20, 21]. IRA codes, introduced by Jin, Khandekar and McEliece [19], are another class of codes that perform competitively with LDPC codes and turbo codes. The message-passing algorithms, which have polynomial complexity in code length, can be applied to decode both LDPC and RA/IRA codes. However, IRA codes have a simpler encoding algorithm.

Besides the progress in channel coding theory in the recent years, multi-media transmission over various channels has quickly evolved from novelty to an important way for information distribution. Compression is a must in image and video transmissions. In the past decade, great progresses have also been made in practical source code designs. Several wavelet-based image/video coding algorithms [22, 23, 24, 25, 26] have been developed. These algorithms have the distinct feature of being able to generate scalable (or embedded) bitstreams while offering superb coding performance.

Moreover, scalable encoding enables progressive transmission, which facilitates designing the JSCC strategy in communication systems.

In this dissertation, we will focus on the design aspect of graph-based channel codes such as LDPC and Repeat Accumulate codes and their applications to the JSCC problems and distributed source coding problems using channel codes. Distributed source coding problems can be interpreted as channel coding problems in which a good channel corresponds to a good source code. Hence, the primary interest of the dissertation is the design of graph-based, soft iteratively decodable codes with good performance and low complexity.

A. Organization of the Dissertation

In this dissertation, the design LDPC or IRA codes for applications such as scalable multi-media transmission and distributed source coding are considered.

This dissertation is organized as follows.

In Chapter II, some background material is provided.

In Chapter III of the dissertation, we propose rate-compatible IRA codes in JSCC problems. In a scalable multi-media transmission system, rate-compatible channel codes are required to provide UEP for bits of different importance. Conventional rate-compatible channel codes include RCPC and RCPT. In this chapter, we design rate-compatible IRA codes and, analyze using density evolution the effect of puncturing applied to IRA codes and the asymptotical performances of the codes applied to scalable multi-media transmission systems. We test with two scalable image coders (SPIHT [22] and JPEG-2000 [23]) and two scalable video coders (3-D SPIHT [24] and H.26L-based PFGS [27]). Simulations show better results with IRA codes than those reported in [28] with JPEG-2000 and turbo codes. The IRA codes proposed

here also have lower decoding complexity than the turbo codes used in [28].

In Chapter IV of the dissertation, we consider designing source-optimized IRA codes. The idea is to take advantage of the capability of unequal error protection (UEP) of IRA codes against errors because of their irregularities. Most existing UEP codes only provide different error rates for information bits of different importance but fail to achieve optimal performance for a criterion other than bit error rate (e.g. average distortion). In this chapter, we propose an approach to design IRA codes optimized for such a criterion. We assume a cost function for the error events caused by the decoders. The cost function is determined by the location of the first error bit. We incorporate this cost function into channel code design and obtain IRA codes that are optimized over a new cost function instead of probabilities of error. This approach is applied to design IRA codes for the transmissions of SPIHT-coded images over noisy channels. It is shown that for long codes (with one codeword for the whole 512×512 image), the simulated results of the codes designed with our approach are only 15% away the theoretical limit compared to 39% reported in [29]. Therefore, compared with conventional IRA codes, the advantage of source-optimized IRA codes is not significant, which reconfirms Shannon's separation principle. For applications such as the transmission of QCIF images where short IRA codes are required and Shannon's separation principle is no longer applicable, source-optimized IRA codes perform much better.

Chapter V of the dissertation includes the investigation of Slepian-Wolf coding problem using LDPC codes. The issues to be addressed include coding problems involving the compression of multiple non-binary sources to their joint entropy and practical code design. LDPC codes are suitable for such an application because the correlation model can be used in the LDPC code design. In [30] and [31], with the exploitation of the correlation between two and three binary sources, it had been

shown that it is possible to approach the Slepian-Wolf limits with appropriate LDPC code designs. Using chain rule for the joint entropy, we propose multi-level codes with LDPC codes as component codes for Slepian-Wolf coding of multiple sources and a multiple stage decoding algorithm for the codes. This chapter also presents a design technique, EXIT charts, for non-binary LDPC codes which can be component codes in a multi-level code for Slepian-Wolf coding of non-binary sources. Examples of code design for four binary sources and two ternary sources are given with simulations results that are very close to the theoretical limits.

Finally, Chapter VI summarizes the dissertation and presents concluding remarks.

B. Contributions of the Dissertation

This dissertation made the following contributions:

- Rate-compatible IRA Codes: Punctured turbo codes suffer from performance loss at high rate due to excess puncturing. On the other hand, although LDPC codes can designed individually with best performances at each rate, it is usually difficult to obtain high rate LDPC codes with good performances via puncturing a low rate LDPC code. Regarding the shortcoming of turbo and LDPC codes, we propose rate-compatible IRA codes which provide good performances for a range of rates.

- Source-optimized IRA Codes: Conventional approach to the joint source-channel coding problems is to design channel codes and the UEP schemes separately. We propose a novel approach which integrates the UEP scheme in channel code design and provides performance gain over the conventional approaches.

- Code Design for Slepian-Wolf Coding of Multiple Non-binary Sources: Previous work regarding this problem only considers two sources. Inspired by the work of Liveris [30], we propose an approach for Slepian-Wolf coding of multiple non-binary sources based on the idea of multi-level coding.

CHAPTER II

BACKGROUND

A. LDPC Codes

An (n, k) binary LDPC code is defined by an $(n-k) \times n$ sparse parity check matrix H in a non-systematic form. Another representation of LDPC codes is using Tanner graphs. A Tanner graph of an LDPC code, as shown in Fig. 1, is a bipartite graph using variable nodes and check nodes to represent the columns and rows of the parity check matrix in which the index of a non-zero entry of the matrix represents the index of the pair of neighboring variable and check nodes. The most important parameters that characterize an LDPC code ensemble are a pair of degree profiles, $\lambda(x)$ and $\rho(x)$. $\lambda(x)$ is given as

$$\lambda(x) = \sum_i \lambda_i x^{i-1}, \tag{1}$$

and $\rho(x)$ is defined as

$$\rho(x) = \sum_j \rho_j x^{j-1}, \tag{2}$$

where λ_i and ρ_i are the fraction of edges coming from check nodes and information nodes incident to i edges, respectively. An LDPC code ensemble is said to be regular if λ_i and ρ_j are one for some i and j; otherwise it is said to be irregular. For a given pair of degree profiles, the rate corresponding to LDPC codes is

$$R = 1 - \frac{\sum_i \rho_i / i}{\sum_i \lambda_i / i} \tag{3}$$

For non-binary LDPC codes, the entries in the parity matrix are elements from a Galois field and their arithmetic operations are defined on the associated field. Encoding is an issue for LDPC codes. However, through some arrangement on parity

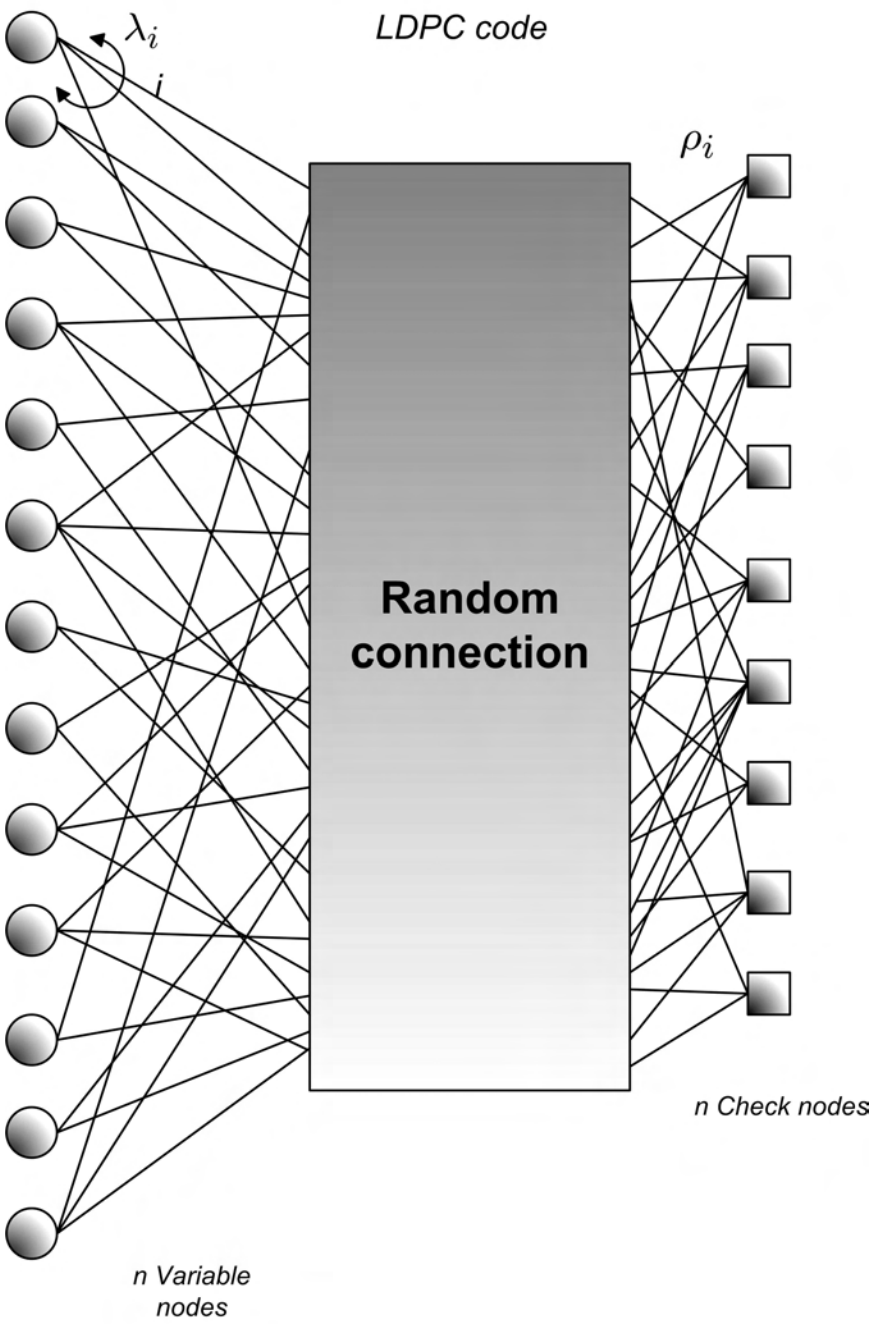

Fig. 1. Graph representation of LDPC codes.

check matrices or having the matrices in a certain form, encoding can be done with near linear or linear complexity with code lengths [32, 33]. Another desirable property of channel codes is rate compatibility. Although puncturing and extending are possible for LDPC codes, change of rate and/or length would usually require a reconstruction of the parity check matrix H and the corresponding generator matrix G [34]. On the other hand, IRA codes are better in offering such flexibility.

B. IRA Codes

IRA codes are a generalization of RA codes introduced in [18], and as such have a natural encoding algorithm. The Tanner graphs of IRA codes are shown in Fig. 2. There are two kind of nodes in the graph: variable nodes (circle) and check nodes (square). There are k variable nodes on the left, called information nodes; and there are k check nodes and $n-k$ variable nodes on the right, called parity nodes. The difference between LDPC and IRA codes is that in the Tanner graph of IRA codes, the information bits $\{u_1....u_k\}$ of IRA codes are first repeated irregularly and the repeated information bits are interleaved. Then groups of interleaved bits are encoded by single parity checks (SPC) and a rate-1 convolutional encoder with generator polynomials $1/1+D$ to generate parity bits as shown in the figure. Hence, IRA codes as shown are in a systematic form and have a natural encoding algorithm. It is also possible to construct nonsystematic IRA codes, in which only the bits corresponding to parity nodes are transmitted. Due to the applications in this work, we only consider systematic IRA codes. For IRA codes, we can also define a pair of degree profiles, $\lambda(x)=\sum_i \lambda_i x^{i-1}$ and $\rho(x)=\sum_j \rho_j x^{j-1}$, where λ_is are the fractions of the edges connected to information nodes to which are exactly i edges connected and ρ_j are the fractions of the edges connected to check nodes to which are exactly

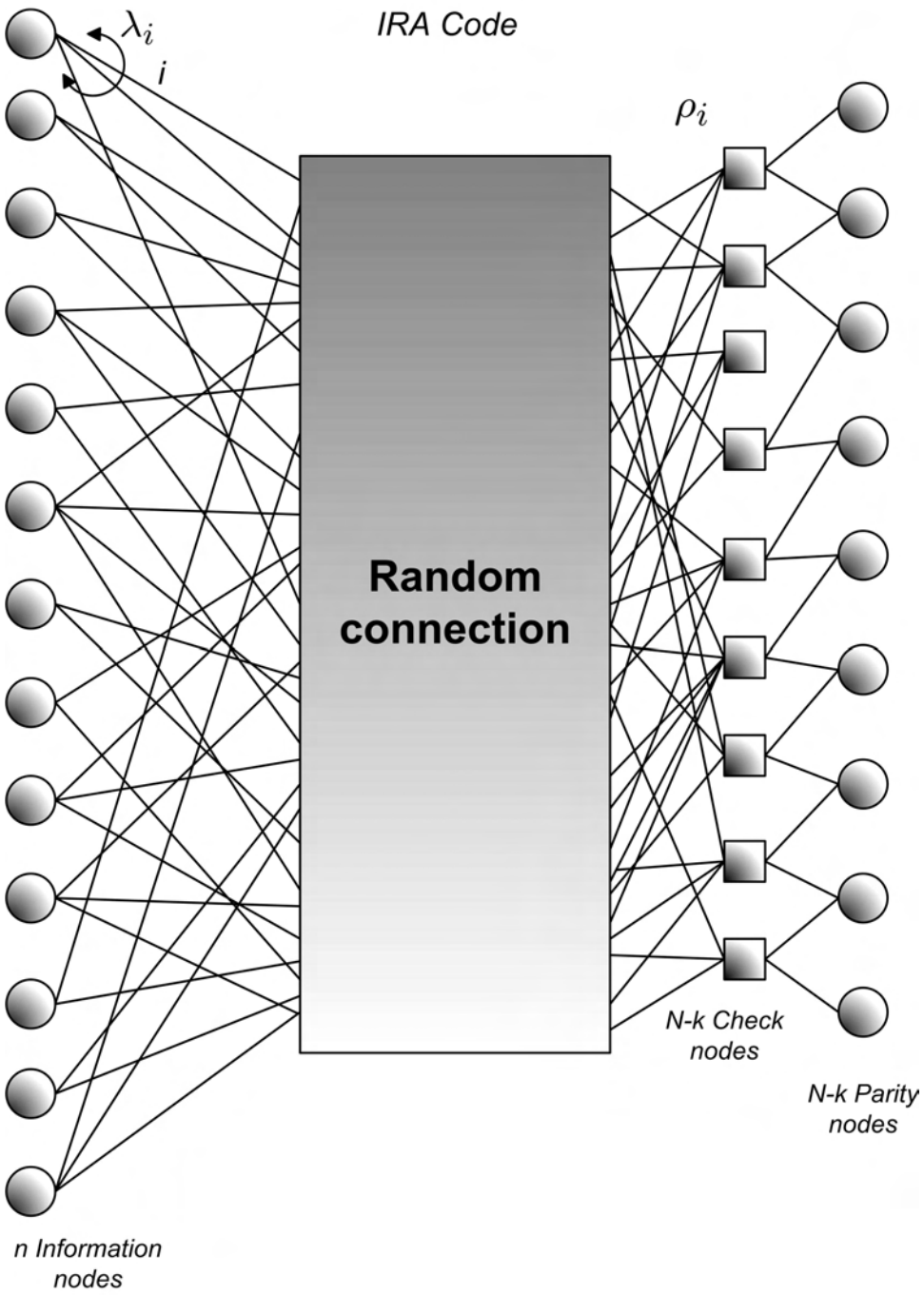

Fig. 2. Graph representation of IRA codes.

$j+2$ edges connected (two edges are connected to parity nodes). The rate of the nonsystematic IRA code is easily seen to be

$$R = \frac{\sum_i \frac{\lambda_i}{i}}{\sum_j \frac{\rho_j}{j}} \tag{4}$$

and the rate of the systematic IRA code is

$$R = \frac{\sum_i \frac{\lambda_i}{i}}{\sum_i \frac{\lambda_i}{i} + \sum_j \frac{\rho_j}{j}}. \tag{5}$$

It is shown by Narayananswami [35] that IRA codes can be viewed as a special class of LDPC codes whose parity check matrices take a particular form as shown in Fig. 3. The decoding algorithms used for LDPC codes can be applied to IRA codes as well. In addition to having a simple encoding algorithm, it is also simple to have good higher rate codes by puncturing a lower rate code, which is a useful property in communication systems with ARQ protocol and scalable multi-media transmission systems.

C. Decoding of Codes on the Tanner Graphs

The decoding of binary LDPC codes uses iterative an message-passing algorithm, which is essentially an instance of Pearl's belief propagation algorithm operating on the Tanner graph of the codes. The algorithm is used to solve the probabilistic inference problems in Bayesian networks [36]-[40]. The message-passing algorithm is also known as the sum-product algorithm [41], in which the messages are passed between the variable and check nodes. It is shown by Wiberg [41] that if there are no loops in the graph and the graph is finite, then the sum-product algorithm after finitely many iteration is equivalent to a MAP decoding algorithm. Although in practice cycles are unavoidable in an LDPC code, the sum-product algorithm (suboptimal for codes with

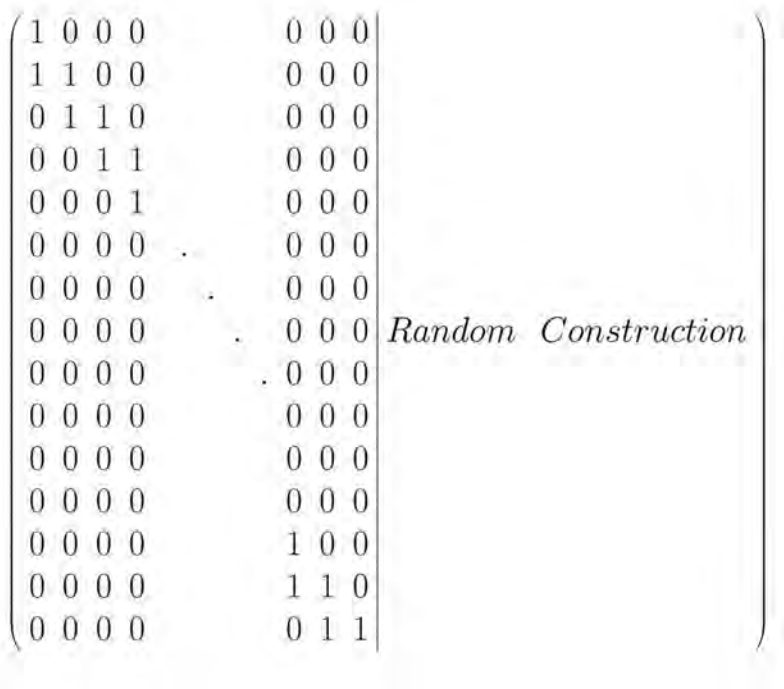

Fig. 3. Parity check matrix of IRA codes.

cycles) nevertheless performs quite well. For decoding a non-binary LDPC code, the message-passing algorithm operates in terms of probabilities of the estimated values of information symbols not log-likelihood ratios. It has been shown that carefully designed LDPC codes perform just as well as turbo codes but with less decoding complexity. The message-passing algorithm can be employed to decode LDPC and IRA codes. For binary codes, at information, parity and variable nodes, the outgoing messages at the nodes are updated as:

$$x_k = \sum_{i \neq k} x_i. \tag{6}$$

For an edge connected to a check node, it can be shown that the outgoing messages on the edge are given by the tanh rule [42]:

$$\tanh \frac{x_k}{2} = \prod_{i \neq k} \tanh \frac{x_i}{2}. \tag{7}$$

where x_i's are the messages conveyed on the edges incident on the nodes.

For non-binary LDPC codes, the entries in the parity matrices are elements from Galois field and their arithmetic operations are defined on associated field. In decoding of the non-binary codes, messages passed in the graphs are vectors. Let $N(m)$ be the set of variable nodes that participate in the check node m and $M(n)$ be the set of check nodes that depend on the variable node n . Define q^a_{mn} as the probability that variable node n is a given the information obtained via check other than m. Define r^a_{mn} as the probability that check m is satisfied if the variable node n is fixed at a. The update rule of the message at variable nodes is given as

$$q^a_{mn} = \alpha_{mn} f^a_n \prod_{j \in M(n) \backslash m} r^a_{jn} \tag{8}$$

and the update rule at check nodes is given as

$$r_{mn}^{a} = \sum_{X':X'_n=a} \mathrm{pr}[Z_m|X'] \prod_{j\in N(m)\backslash n} q_{mj}^{x'_j}. \tag{9}$$

where a is an element in GF(q).

D. Design of Code Ensembles

LDPC and IRA code ensembles are both determined by λ_i and ρ_i. Richardson and Urbanke [14, 15, 16, 17] proposed a numerical technique called *Density Evolution* to analyze LDPC code ensembles. The basic idea of density evolution is to keep track of the evolution of probability density functions, which evolve with each iteration during decoding, of message passed in the graph. When there are no loops in the graph, this numerical procedure can be implemented with the Fast Fourier Transform (FFT) and a change of measure. For a Gaussian channel, the evolving probability density function can be described with one parameter by approximating the evolving probability density functions with Gaussian distributions. Another approach proposed by ten Brink [43] is based on EXtrinsic Information Transfer (EXIT) charts. EXIT charts are used to characterize with information rates the input-output properties of variable and check nodes by treating them as individual information processing units. One can use the charts to evaluate the performances of LDPC codes. For a capacity-approaching code, the input-output information rate transfer curves of variable nodes should not intersect with those of check nodes, i.e. fixed-point should be avoided. With the information rate transfer curves of variable and check nodes of different degrees, we can find a pair of λ_i and ρ_i which maximize 3 and 5 provided no fixed points. The coding rates corresponding to the resulting λ_is and ρ_is are the upper bound of achievable rates. Note that the bound is exact for erasure channels.

E. Joint Source-channel Coding

When communication systems operating under a delay-constraint on a time-varying channel, it is generally no longer optimal to design channel and source coders separately. Given a fixed amount of resource for transmission of image or video over a noisy channel, it is necessary to balance the amount of the parity bits used for the protection from noise introduced by channel and the amount of the source bits used for the quality of image or video. The algorithms for optimal allocation of channel bits and source bits have been a very active research area.

If scalable source coders are used in the systems, we need to take into account the characteristics of the source bitstream when considering joint source channel coding problems. Scalability induces a sequential dependency among the compressed multimedia source bits, calling for unequal error protection (UEP) in the form of more protection for the beginning part of the bitstream, less for the middle part, yet still less or even no protection at all for the last part. This is different from conventional data communications where all bits are equally important and hence equal error protection (EEP) suffices. UEP can be achieved by using channel codes of different rates via puncturing a low rate channel code. The requirement for rate-compatibility requirement restricts the puncturing rule such that all of the code bits of a higher rate punctured code are used by the lower rate codes. In other words, higher rate codes are embedded in lower rates codes. Therefore, only one encoder and decoder are required at the transmitter and the receiver respectively, which reduces hardware requirements. Traditional rate-compatible coding schemes such as convolutional codes and turbo codes with practical constraint lengths suffer from performance loss at high rate due to excess puncturing. The excess puncturing also causes error floors for turbo codes, which makes the analysis of the effect of puncturing difficult. The Richardson-

Urbanke-type analysis used for designing LDPC and IRA codes can also be applied to evaluate the performance of IRA codes after puncturing, in which we can treat periodical puncturing as erasures. With Richardson-Urbanke-type analysis, we are able to design good IRA codes as UEP codes used in JSCC.

F. Slepian-Wolf Coding Problem

Consider lossless source coding. It is well-known that the rate to encode a source X_1 is at least $H(X_1)$. Suppose that we have to encode two sources (X_1, X_2) together. A rate of $H(X_1, X_2)$ is sufficient. A interesting question arises when these two source must be separately encoded. It is obvious that a rate not less than $H(X_1)+H(X_2)$ is sufficient. However, in a landmark paper by Slepian and Wolf, it is shown that a total rate $H(X_1, X_2)$ is sufficient for separate encoding of X_1 and X_2, provided that the decoding is performed jointly at the decoder. The proof of achievability is based on random binning. The idea of binning is best described as follows: Choose a large number index for each source sequence. If the set of typical source sequences is small, decoders can recover the source sequence from the index with high probability since different source sequence corresponds to different index. This above theorem also holds true for n sources where a total rate of $H(X_1, X_2, ..., X_n)$ is sufficient for compression of X_1, X_2, ..., X_n. The asymmetric Slepian-Wolf coding problem for two binary sources , X_1 and X_2 (side information), can be treated as the problem of channel coding problem with side information at the decoder. Wyner [44] suggested modelling correlations with equivalent channels and applying the syndrome approach to the asymmetric coding problem. Assuming asymmetric coding, from the chain rule for the joint entropy, X_i can be encoded with a channel code at rate $1 - H(X_i|X_1, X_2, ..., X_{i-1})$ and decoded given $X_1, X_2, ..., X_{i-1}$ for $i = 1, 2, ..., n$. In consequence, we are able to relate Slepian-

Wolf coding problems for n sources to channel coding problems via multi-level coding.

CHAPTER III

SCALABLE IMAGE AND VIDEO TRANSMISSION USING IRREGULAR REPEAT ACCUMULATE (IRA) CODES WITH FAST ALGORITHM FOR OPTIMAL UNEQUAL ERROR PROTECTION[1]

This chapter considers designing and applying punctured IRA codes [19] for scalable image and video transmission over binary symmetric channels. IRA codes of different rates are obtained by puncturing the parity bits of a mother IRA code which uses a systematic encoder. One of the main ideas presented here is the design of the mother code such that the entire set of higher rate codes obtained by puncturing are good. To find the optimal unequal error protection for embedded bit streams, we employ the fast joint source-channel coding algorithm in [45] to minimize the expected end-to-end distortion. We test with two scalable image coders (SPIHT [22] and JPEG-2000 [23]) and two scalable video coders (3-D SPIHT [24] and H.26L-based PFGS [27]). Simulations show better results with IRA codes than those reported in [28] with JPEG-2000 and turbo codes. The IRA codes proposed here also have lower decoding complexity than the turbo codes used in [28].

A. Introduction

In the recent years, multi-media transmission over various channels has quickly evolved from novelty to an important way for information distribution. Compression is a must in image and video transmission. In the past decade, standards like JPEG-2000 [23], MPEG-2/4 [46, 47] and H.26L [48] have been developed for multimedia data compres-

[1] ©2004 IEEE. Reprinted with permission, from *IEEE Transactions Communications*, vol. 52, pp. 1092 - 1101, July 2004.

sion. Besides efficient compression, scalability is a desirable property for image and video coders. A scalable bit stream can be decoded at different rates with commensurate reconstruction quality [26]. This means that the source needs to be encoded only once for different quality requirements. Moreover, scalable encoding enables progressive transmission and bandwidth adaptation. In image transmission, for example, a receiver does not have to wait until all bits are received before decoding the image, instead it can use additional received bits to improve the quality of the previous reconstructed image.

In many cases, the channel is noisy and can be modelled as a binary symmetric channel (BSC). Recently, scalable image and video transmission over both BSC and packet erasure channels have been considered in [49, 50, 51, 52]. These schemes generally involve embedded source coding [22, 23, 24], channel coding [53, 54] and joint source-channel coding (JSCC) [49, 55]. In [29], Sherwood and Zeger showed the effectiveness of using set partitioning in hierarchical trees (SPIHT) coder [22] and equal error protection with rate-compatible punctured convolutional (RCPC) codes [53] for image transmission over the BSC channel. In [28], for the same channel, the source and channel codes were updated to JPEG-2000 and turbo codes, respectively, and unequal error protection (UEP) was used to provide more efficient error control.

Along with the advances in source coding there have been exciting new developments in channel coding – from turbo codes in the mid-1990s to the recent rediscovery of low-density parity-check (LDPC) codes. In 2000, Jin *et al.* introduced a class of codes closely related to LDPC codes called irregular repeat accumulate (IRA) codes in [19]. Like turbo codes (and unlike LDPC codes), IRA codes can be encoded easily in linear time. Like LDPC codes (and unlike turbo codes), they are amenable to an exact Richardson-Urbanke type analysis [14]. IRA codes also perform very close to the capacity limit on AWGN channels.

In this chapter, we propose a JSCC technique for scalable image and video transmission over BSC channels whose crossover probability (COP) can change in time. By using UEP, we find the optimal source-channel coding trade-off to provide error robustness for transmission of scalable bit streams. A novel algorithm for computing the required channel coding rate for a given source and COP is proposed. For the channel coding part, we propose to use IRA codes of different rates obtained by puncturing the parity bits of a mother IRA code. The novelty is that only the degree profiles of the mother IRA code are optimized such that the entire range of higher rate codes obtained by puncturing the mother code is still good. We show that the proposed JSCC scheme can be analyzed asymptotically (in length) in terms of mean square error (MSE). Simulation results for JPEG-2000 image transmission show that our proposed solution is uniformly better than the scheme in [28], with significantly lower decoding complexity for the channel codes, and significantly lower complexity for finding the optimal UEP solution. We also apply our framework to transmit 3-D SPIHT and H.26L-PFGS video over BSC.

The rest of this chapter is organized as follows. Section B overviews scalable source coding with examples of the different coding algorithms that we use in this work. Section C presents the design and analysis of the proposed punctured IRA codes. Section VI explains the JSCC scheme and the fast UEP algorithm. Section D presents simulation results and Section E summarizes the chapter.

B. Scalable Source Coding

1. Image Coding

The SPIHT algorithm [22], like the EZW coding algorithm [26], is based on the idea of using multi-pass "zero-tree" coding to transmit the largest wavelet coefficients (in

magnitude) first. The underlying assumption is that most images can be modelled as having decaying power spectral densities. That is: if a parent node in the wavelet coefficient tree is insignificant, it is very likely that its descendants are also insignificant. The zero-tree symbol is used very efficiently in this case to signify a spatial subtree of zeros. When the thresholds are powers of two, SPIHT coding can be thought of as a bit-plane coding scheme. It encodes one bit-plane at a time, starting from the most significant bit. With the sign bits and refinement bits (for coefficients that become significant earlier) being coded on the fly, SPIHT achieves *embedded* coding in the wavelet domain. The SPIHT coder performs competitively with most other coders published in the literature [56], while possessing desirable features such as relatively low complexity and rate embeddedness.

In response to the rapid progress in wavelet image coding research, the International Standards Organization has adopted the wavelet transform as the workhorse in the new JPEG-2000 image coding standard. The baseline JPEG-2000 coder employs the embedded block coding with optimized truncation (EBCOT) [57] algorithm for bit-plane coding of wavelet coefficients. While the SPIHT algorithm applies arithmetic coding [58] on the significant bits only, EBCOT additionally uses arithmetic coding on the sign bits and refinement bits. Furthermore, EBCOT breaks one bit-plane into three $fractional$ bit-planes and compresses them in decreasing order of rate-distortion (R-D) importance. Because of this, the complexity of JPEG-2000 coding is higher than that of SPIHT coding. In terms of compression efficiency, JPEG-2000 performs comparably to SPIHT. The strength of the JPEG-2000 standard lies in its rich set of features such as lossy and lossless compression, scalability in rate and image resolution, region of interest (ROI) coding, and error resilience.

2. Video Coding

The 2-D SPIHT algorithm [22] was extended to 3-D embedded SPIHT video coding in [24]. Besides motion compensation, the 3-D SPIHT algorithm is in principle the same as 2-D SPIHT, except that 3-D wavelet coefficients are treated as a collection of 3-D spatio-temporal orientation trees and that context modelling in arithmetic coding is more involved. Spatio-temporal orientation trees coupled with powerful SPIHT sorting and refinement turns out to be very efficient. Even without motion compensation, the 3-D SPIHT coder provides comparable performance to H.263 [59] objectively and subjectively when operating at bit rates of 30 to 60 Kbps. It outperforms MPEG-2 at the same bit rate (1.5 to 4 Mbps). In addition to being rate scalable, the 3-D SPIHT video coder allows multi-resolutional scalability in encoding and decoding in both time and space. This added functionality along with many desirable features, such as full embeddedness for progressive transmission, precise rate control for constant bit rate traffic, and low complexity for possible software only video applications, makes the video coder an attractive candidate for applications like wireless video.

The H.26L [48] video coding algorithm aims at higher coding efficiency and is developed by ITU-T as a long-term video coding standard for low bit rate applications. H.26L-based PFGS [27] coding adds an embedded enhancement layer to the H.26L bit stream. It compresses a video sequence into two bit streams: one for the low-quality, non-scalable base layer and another for the scalable enhancement layer. The base layer bit stream is generated by the classic motion-compensated discrete cosine transform (DCT) approach. The difference between the original video sequence and the reconstructed version from the base layer is compressed with bit-plane coding to form the enhancement layer bit stream. Since bit-plane coding produces an embedded bit stream with fine granularity scalability, the enhancement bit stream can be

arbitrarily truncated to fit the available channel bandwidth.

C. IRA Codes as Channel Codes in Joint Source-channel Coding

1. System Model and Background

Irregular Repeat Accumulate (IRA) codes are a class of LDPC codes which can be easily encoded in systematic form [19]. It was shown by Jin *et al* that IRA codes can achieve capacity on the erasure channel and can perform very close to capacity on the binary input AWGN channel. IRA codes can also be thought of as a generalization of repeat accumulate (RA) codes proposed in [18]. The IRA code structure can be best explained in terms of the Tanner graph of the parity check matrix as shown in Fig 2. In the encoder for an IRA code, the information bits $\{u_1....u_k\}$ are first repeated irregularly and the repeated information bits are interleaved. Then groups of interleaved bits are encoded by a single parity check (SPC) code as shown in the figure. The parity bits of the SPC code are encoded using a differential encoder or, equivalently, a rate-1 convolutional encoder with generator polynomials $1/1+D$, the output of which is the sequence $\{p_1....p_r\}$. The graph representation of the differential encoder is shown in Fig 2. Note that in the graph, the check (square) nodes represent the fact that the modulo-2 sum of the nodes connected to the check is zero.

We only consider IRA codes in their systematic form in this paper. That is, the codeword corresponding to information bits $\{u_1....u_k\}$ is given by $\{p_1....p_r, u_1....u_k\}$. In general, the IRA code ensemble is specified by two degree profiles $\lambda(x) = \sum_i \lambda_i x^{i-1}$ and $\rho(x) = \sum_i \rho_i x^{i-1}$, where λ_i and ρ_i are the fraction of edges incident on information nodes and check nodes with degree i, respectively. In this paper, the check degrees are assumed to be concentrated, i.e., $\rho(x) = x^{a-1}$ for some integer a. This means that all single parity check codes used are of rate $a/(a+1)$. In this case, given $\lambda(x)$ (or,

simply λ) and a, the rate of the code is:

$$R = \frac{\sum_i \lambda_i / i}{1/a + \sum_i \lambda_i / i}. \tag{10}$$

A BSC channel with COP ϵ is assumed. This received vector $r = \{r_0, r_1, \ldots, r_{N-1}\}$, $r_j \in \{0, 1\}$ can be decoded with an iterative message-passing algorithm. The messages passed between nodes are extrinsic log-likelihood ratios (LLR). In order to facilitate the discussion of the code design, we first establish some notation and briefly describe the decoding algorithm for the BSC.

- $x_{i\to c}$ and $x_{c\to i}$ denote the messages passed between check and information nodes and let $U_{i\to c}(x)$ and $U_{c\to i}(x)$ be their corresponding probability density functions (pdf)s.
- $x_{p\to c}$ and $x_{c\to p}$ denote messages passed between check and parity nodes and let $U_{p\to c}(x)$ and $U_{c\to p}(x)$ be their corresponding pdfs.
- x_p denotes the LLR provided by the channel on the parity bit and $U_p(x)$ its pdf.
- x_i denotes the LLR provided by the channel on the information bits and $U_i(x)$ its pdf.

The LLR for the jth bit in the codeword (information or parity bit) is given by $(1 - 2r_j) \log \frac{1-\epsilon}{\epsilon}$, where ϵ is the COP of the BSC. At an information or parity node, the outgoing message from the node along the kth edge is given by

$$x_k = \sum_{m \neq k} x_m, \tag{11}$$

where x_m is the incoming messages conveyed on the edges m. At a check node, it can

be shown that the outgoing message along the kth edge is given by the tanh rule [42]

$$\tanh \frac{x_k}{2} = \prod_{i \neq k} \tanh \frac{x_m}{2}, \tag{12}$$

where x_m's are incoming messages. We then have the following decoding algorithm:

- Step 1: Initialize $x_{i \to c}$ and $x_{p \to c}$ with given x_i and x_p.
- Step 2: Calculate $x_{c \to p}$ and $x_{c \to i}$ according to (12).
- Step 3: Calculate $x_{p \to c}$ and $x_{i \to c}$ according to (11).
- Step 4: Repeat Step 2 to Step 3 until the maximum number of iteration is reached.

2. Density Evolution

The key idea in density evolution is to track the pdf of the messages in every iteration assuming the codeword length is infinite. When the codeword length $N \to \infty$, the Tanner graph has no cycles, hence all the messages that are passed in the decoder can be modelled as independent random variables. In this case, the pdfs of the random variables and the probability of error can be computed at each iteration as explained below.

Assume that the all-zero codeword is transmitted and follow the notation described in the previous section. For the BSC, the pdfs of the channel outputs $U_i(x)$ and $U_p(x)$ are given by

$$U_i(x) = U_p(x) = U_{ch}(x) = (1 - \epsilon)\delta(x - t) + \epsilon\delta(x + t) \tag{13}$$

where $t = \log \frac{1-\epsilon}{\epsilon}$, $\delta(x) = 1$ if $x = 0$ and $\delta(x) = 0$ if $x \neq 0$. Note that although $U_p(x)$ and $U_i(x)$ are identical for the BSC when puncturing is not used, we do make a distinction between $U_p(x)$ and $U_i(x)$ since it is required to handle the puncturing

case.

From (11), we can see that the pdf of the outgoing message at an information or parity node is simply the convolution of the pdfs of all the incoming messages. From (12), we can compute the pdf of x_k with a change of measure and the Fourier transform if the pdf $f_m(x)$ of each incoming message is known. We use $\bigotimes$ to denote this operation, then the pdf of the outgoing message is given by

$$f_k(x) = \bigotimes_{m \neq k} f_m(x) \tag{14}$$

Readers interested in how to compute $f_k(x)$ are referred to [15, 42].

The pdfs of the messages passed in the graph at the $(n+1)$-th iteration can be computed from those at the nth iteration according to:

$$\begin{aligned}
U_{c\to p}^{n+1}(x) &= \bigotimes {}^{a}U_{i\to c}^{n}(x) \bigotimes U_{p\to c}^{n}(x), && (15)\\
U_{c\to i}^{n+1}(x) &= \bigotimes {}^{a-1}U_{i\to c}^{n}(x) \bigotimes {}^{2}U_{p\to c}^{n}(x), && (16)\\
U_{i\to c}^{n+1}(x) &= \sum_{i} \lambda_i *^{i-1} U_{c\to i}^{n+1}(x) * U_i(x), && (17)\\
U_{p\to c}^{n+1}(x) &= U_{c\to p}^{n+1}(x) * U_p(x), && (18)
\end{aligned}$$

where $*^i$ and $\bigotimes^i$ mean applying conventional convolution and the operation defined in (14) i times, respectively.

The probability of decoding error after the nth iteration is given by

$$P_n(\lambda, a, U_i, U_p) = \int_{-\infty}^{0} \sum_{k} \lambda_k \{*^k U_{c\to i}^{n}(x)\} * U_i(x)\, dx, \tag{19}$$

which will be referred to as the probability of error from now on.

3. IRA Code Design

Note that the probability of error depends on the degree profiles $\lambda(x)$ and a and on the channel condition through the initial pdfs $U_i(x)$ and $U_p(x)$. The design of good IRA codes involves the design of λ and a for a fixed rate such that the probability of error is minimized for given $U_i(x)$ and $U_p(x)$. For a given a, it can be formulated as

$$\lambda_{opt} = \arg \min_{a,\lambda \subseteq \lambda'(R,a)} P_n(\lambda(x), a, U_{i,R}(x), U_{p,R}(x)), \tag{20}$$

where $\lambda'(R,a)$ is the set of all valid degree profiles that satisfy the rate constraint, i.e.,

$$\lambda'(R,a) \doteq \left\{ \lambda_i \Big| \sum_i \lambda_i = 1, \frac{\sum_i \lambda_i/i}{1/a + \sum_i \lambda_i/i} = R, \lambda_i \geq 0 \right\}.$$

4. Code Design for Punctured IRA Codes

Scalable image and video transmission requires channel codes with good performance over a range of rates. The optimum solution is to optimize the degree profiles separately for each rate; however, this is computationally intensive. Here, we propose to optimize the degree profile for the mother code such that higher rate codes can be derived by puncturing. The main point, however, is the design of the degree profile of the mother code such that the entire set of codes obtained by puncturing performs well. Hence only one degree profile needs to be designed. In practice, this also makes the design of the decoder easier since the decoder needs to be designed for only one degree profile.

We propose to modify the above optimization problem to fit this application. To analyze the performance of IRA codes after puncturing, we treat punctured bits as erasures. That is, we assume that the channel as seen by the parity bits is a mixture of the actual BSC channel and an imaginary erasure channel. To take puncturing into

account, we define $U_{i,R_i}(x)$ and $U_{p,R_i}(x)$ as the initial pdfs of the LLRs for information and parity nodes, respectively. These pdfs depend on the COP of the BSC channel and the channel code rates before and after puncturing, namely R and R_i (more precisely on the fraction of the bits that are punctured). For the all-zero codeword being transmitted and a given BER ϵ, $U_{i,R_i}(x)$ denotes the initial pdf of x_i and it is given by

$$U_{i,R_i}(x) = U_{ch}(x) = (1-\epsilon)\delta(x-t) + \epsilon\delta(x+t), \tag{21}$$

where $t = \log\frac{1-\epsilon}{\epsilon}$. The pdf of $U_i(x)$ is unaffected due to puncturing. $U_{p,R_i}(x)$ denotes the initial pdf corresponding to the parity bits after being punctured to a higher rate R_i and it is given by

$$U_{p,R_i}(x) = (1-\gamma)U_{ch}(x) + \gamma\delta(x), \tag{22}$$

where γ is the fraction of parity bits being punctured.

The objective is to design an IRA code degree profile $[\lambda(x), a]$ of rate R_{min} and from this obtain a set of higher rates $\{R_{min}, R_2, \ldots, R_{max}\}$ by puncturing only the parity bits. For a given set of rates $\{R_{min}, R_2, \ldots, R_{max}\}$, the optimization cost function should involve the probability of error for each rate in the set. That is, the true optimum $[\lambda(x), a]$, if it exists, should simultaneously minimize the probability of error for all rates in the set. First of all, it is not clear that such a solution exists. Secondly, even in the case that the solution exists, the optimization problem can be quite hard depending on the number of desired rates. In order to simplify this procedure without much loss in performance, we choose to minimize the maximum of the error probabilities of the highest and the lowest rates R_{min} and R_{max} under the constraint that $R_{min} = \frac{\sum_i \lambda_i/i}{1/a + \sum_i \lambda_i/i}$. In other words, we try to design a mother code that performs well for both the lowest rate (with no puncturing) and highest rate (with maximum puncturing). With the code being good for the two extremes, we

expect that it is good for the entire rate range, which will be corroborated through simulations. The problem can be mathematically stated as:

$$\lambda = \arg \min_{a,\lambda \subseteq \lambda'(R_{min},a)} [\max(P_n(\lambda_1, ..., \lambda_d, U_{i,R_{min}}, U_{p,R_{min}}), P_n(\lambda_1, ..., \lambda_d, U_{i,R_{max}}, U_{p,R_{max}}))]$$

with $\lambda'(R,a) = \left\{ \lambda_i \middle| \sum_i \lambda_i = 1, \frac{\sum_i \lambda_i/i}{1/a+\sum_i \lambda_i/i} = R, \lambda_i \geq 0 \right\}$ and $n, U_{i,R}(x), U_{p,R}(x)$ given *a priori*. Note that the value of ϵ (channel condition) used in determining $U_{i,R_{min}}$ or $U_{p,R_{min}}$ and $U_{i,R_{max}}$ or $U_{p,R_{max}}$ will be different. For a given value of a, this problem can be solved using a non-linear optimization routine. The optimization must be performed for a few values of a and the best one must be selected.

Example 1: Consider the design and construction of punctured IRA codes for the range of COPs $0.0225 \leq \epsilon \leq 0.1715$. We picked $R_{min} = 1/3$ and $R_{max} = 10/12$, since these are fairly close to the BSC capacity limits for $\epsilon = 0.1715$ and $\epsilon = 0.0225$, respectively. Then, the optimization for $\lambda(x)$ and $\rho(x)$ was performed with $n = 15$. A gradient search is used along with several randomly chosen initial profiles in order to bypass the local minima, and the resulting $\lambda(x)$ and $\rho(x)$ are

$$\begin{aligned} \lambda(x) &= 0.240008x^3 + 0.139965x^4 + 0.073483x^5 \\ &\quad +0.025974x^6 + 0.134669x^{23} + 0.385901x^{24} \\ \rho(x) &= x^3. \end{aligned} \tag{23}$$

The performance of several punctured codes obtained from a mother code with this degree profile will be discussed in the following sections.

5. Asymptotic Analysis

In this section, the asymptotic performance of the JSCC scheme based on SPIHT image coding and IRA channel codes is analyzed as the channel codeword length

approaches infinity. The same approach can be applied to video transmission as well.

We assume that the transmission of a symbol is completed in one channel use (on average) at a certain information rate R_ϵ. Then for the JSCC problem with IRA codes and a given channel BER ϵ, the achievable bits per symbol can be defined as

$$R_\epsilon = \max\ R' \quad \text{subject to} \quad \lim_{n\to\infty} P_n(\lambda, a, U_{i,R'}, U_{p,R'}) \to 0. \tag{24}$$

with $U_{i,R'}$ and $U_{p,R'}$ defined in (21) and (22). To obtain R_ϵ for a given COP ϵ, we find the maximum γ (fraction of parity bits being punctured) with a mother code whose degree profiles are fixed to $[\lambda(x), a]$. For IRA codes, a discrete maximization can be performed fairly easily. That is, starting from $\gamma = 0$, we increase γ in small steps and for each γ, the probability of error can be computed using density evolution as in (15) to (19). R_ϵ is the largest rate for which (24) is satisfied. Once R_ϵ is determined, we can define the achievable MSE D_ϵ as follows:

$$D_\epsilon = D(\upsilon, \psi^{-1}(\psi(\upsilon, R_\epsilon), R_\epsilon)),$$

where υ is the original source. ψ and ψ^{-1} are a given source coder and decoder pair. D is the distortion function that computes the MSE between two images.

For a BSC channel with a given BER ϵ, the maximum information rate (capacity) for this channel is given by $C = 1 + \epsilon\ \texttt{log}\ \epsilon + (1-\epsilon)\ \texttt{log}\ (1-\epsilon)$. Then the MSE lower bound for this channel is given by

$$D_{min} = D(\upsilon, \psi^{-1}(\psi(\upsilon, C), C)). \tag{25}$$

Therefore, it is straightforward to obtain the MSE lower bound for BSC channels with a given encoder/decoder pair and a fixed source.

Fig 4 shows the achievable rates (R_ϵ) as a function of ϵ for punctured IRA codes used in this paper along with the BSC channel capacity. It can be seen that the

achievable rates with the optimized mother code are fairly close to the BSC capacity limit over a wide range of rates.

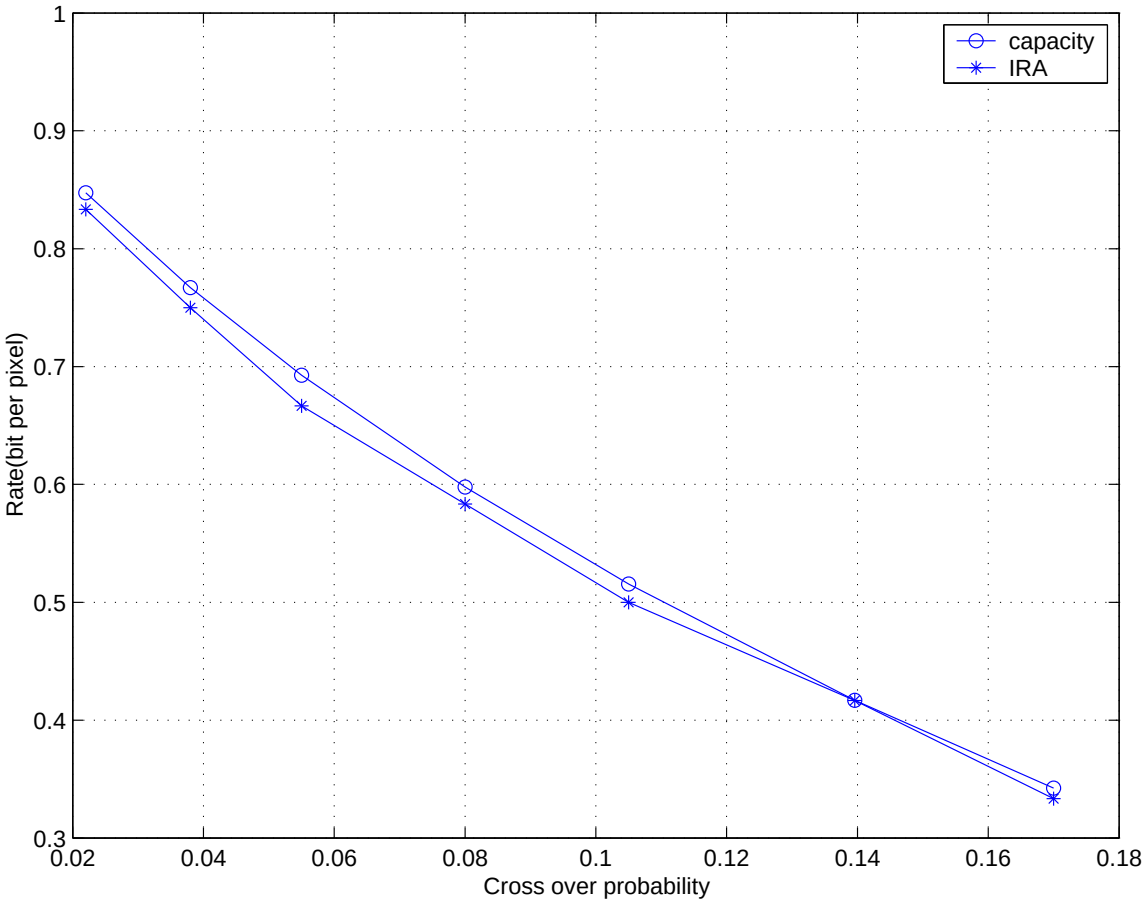

Fig. 4. Achievable rates of IRA codes for BSC channel

Fig 5 depicts the achievable MSE in terms of PSNR using punctured IRA codes and SPIHT encoded 512×512 image Lena. For comparison, the highest achievable PSNR (corresponding to MSE of D_{min}) with the SPIHT encoder and the same image

[1]PSNR is defined as PSNR $= 10\log_{10}(255^2/\text{MSE})$ and measured in decibels (dB).

is also shown. From the plot, it can be seen that asymptotically, the designed codes provide near optimal performance.

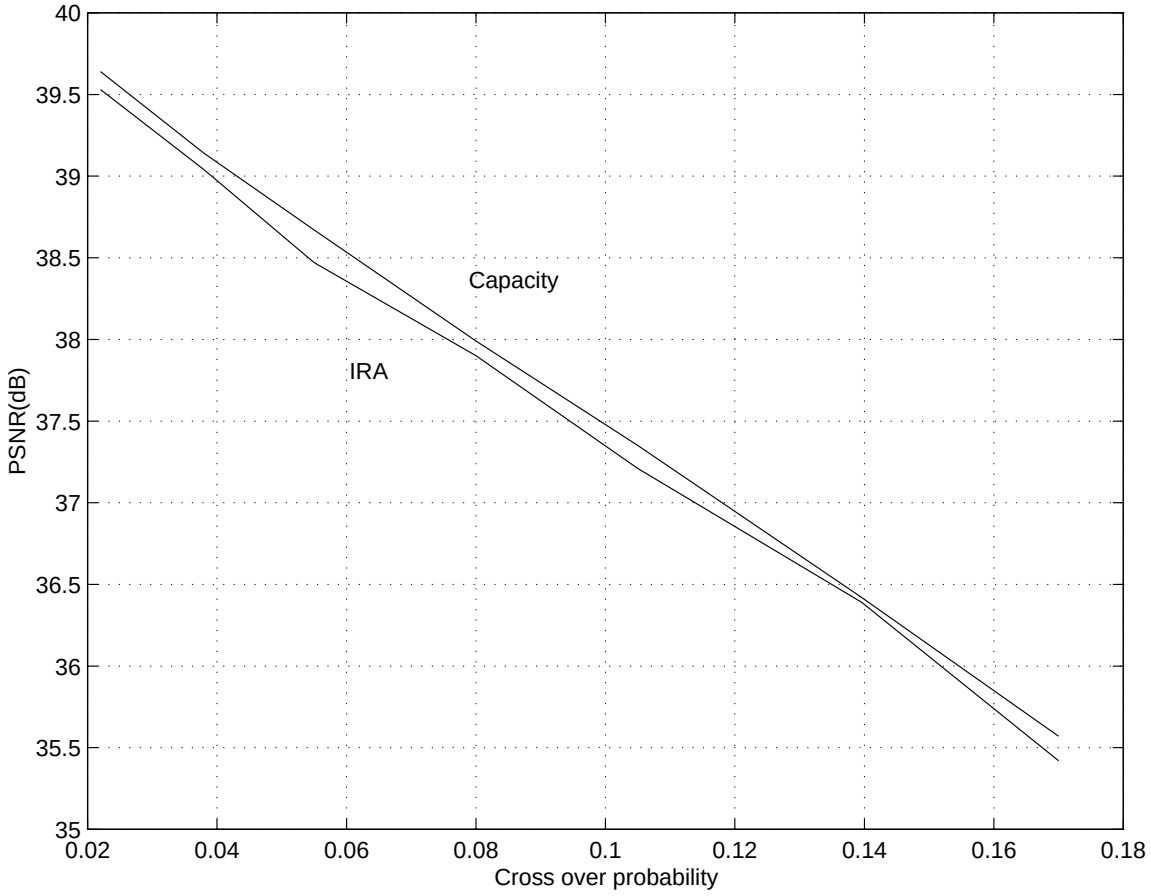

Fig. 5. Achievable PSNR using IRA codes and SPIHT for BSC channel

6. Finite Length Code Construction and Inter-leaver Design for IRA Codes

Once a degree profile is obtained for an IRA code, in practice, one has to construct a particular parity check matrix (or, equivalently the inter-leaver in Fig 2). The concentration theorem in [15] shows that when the length of the codewords approach

infinity, almost all codes constructed will have identical properties and, hence, a randomly chosen code will perform well. However, for finite lengths, one has to pay careful attention to the construction of the parity check matrix. Specifically, the inter-leaver in Fig 2 should be designed such that low weight output codewords are avoided (since the overall code is linear, it is enough to look at the weight of a non-zero codeword).

For the punctured codes, when the source packet length is fixed for all rates, true rate compatibility can be achieved. That is, only one graph or inter-leaver of a fixed length needs to be designed. All the higher rate codes can be obtained by puncturing the parity bits from this graph. However, in this paper, we fix the codeword length to be the same for all rates. As a result, for each rate a separate inter-leaver or graph is designed for each rate; however, the degree profiles are the same as that of the mother code. Uniform puncturing of the parity bits is assumed for all rates.

Divsalar et al. [60] proposed a constrained random inter-leaver characterized by a spreading factor s for use with turbo codes. Denote $\{\pi(i)\}$ as the interleaved sequence from the original ordered sequence $\{i\}$. For two numbers j and k in the sequence $\{i\}$, the s-random inter-leaver ensures that if $|k-j| < s$, then $|\pi(k) - \pi(j)| > s$. S-random inter-leavers can be applied to IRA codes directly and the performance is fairly good. One way to justify the use of S-random inter-leavers in IRA codes is to visualize the generation of parity bits $\{p_1, p_2, \ldots, p_r\}$ from the Tanner graph as shown in Fig 6.

Consider an input sequence U of weight one, that is, only one of the information nodes is non-zero. Let $\{c_1$,c_2,c_3, ..., $c_j\}$ be the check nodes (in ascending order) connected to the non-zero information node. Since the output of the check nodes are input to an accumulator, the output weight starts to accumulate at time instant c_1 and increases until time instant c_2 and, hence, results in an error event of weight

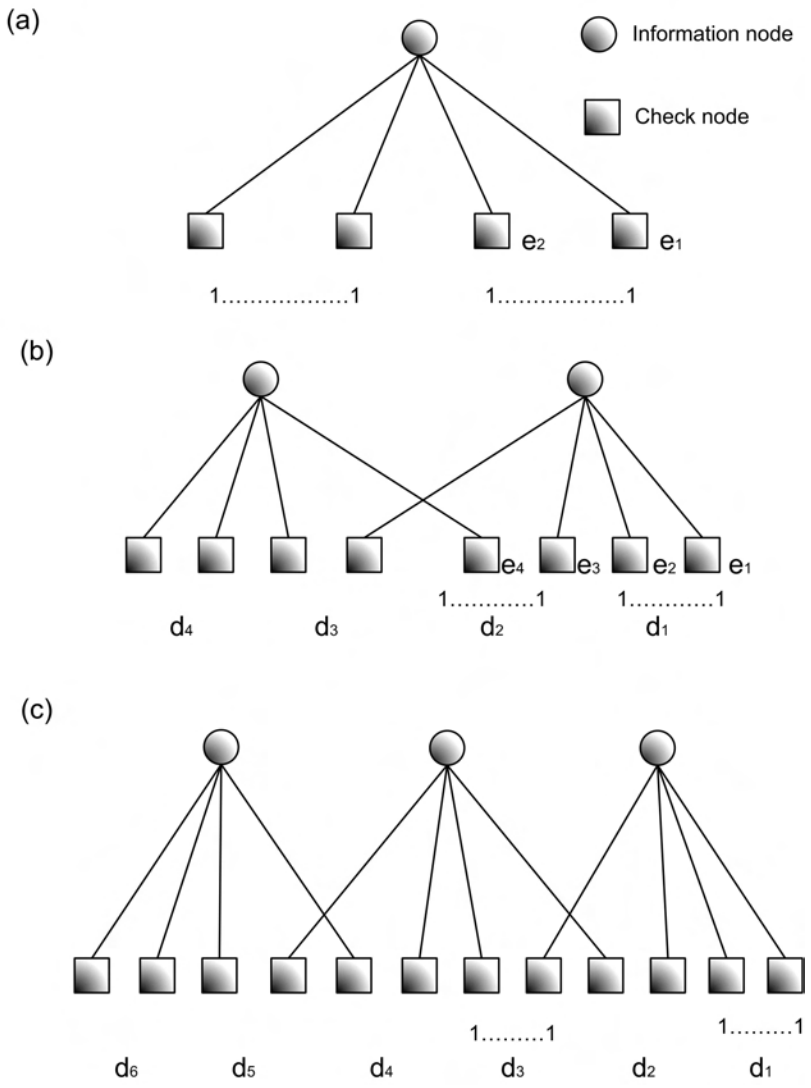

Fig. 6. Error patterns due to weight-1, weight-2 and weight-3 input information sequence

$c_2 - c_1$. Since there are several error events, each with a certain accumulated weight, the total weight of the codeword is given by $\sum_k d_k$, where $d_k = c_{2k} - c_{2k-1}$.

For all weight-1 input sequences, the S-random inter-leaver ensures that $d_k > s$ as long as s is larger than the maximum left degree. Therefore, it is a good choice.

The S-random inter-leaver does not necessarily maintain high weight for the parity bits when the input sequence U has a weight of more than one. Consider the situation when the input weight is two as shown in Fig 6(b). It can be seen that the S-random inter-leaver cannot guarantee $d_k > s$ since two non-zero information nodes with separation larger than s can be connected to check nodes close to each other. Hence, for weight-2 and higher weight input sequences, the inter-leaver construction has to modified to ensure that the sum $\sum_k d_k$ is greater than a threshold s_{th}. Clearly, this is quite complex and we restrict our attention only to weight-2 and 3 input sequences. The proposed modified inter-leaver design procedure can be summarized as follows:

At the ith information node,

- Let the degree of the node be j. Assign the first $j-1$ edges to check nodes such that the S constraint is satisfied.

- Assign the last edge (jth edge) to a check node such that the S constraint is satisfied and, in addition, the codeword weight corresponding to input sequences of weight-2 and 3 with a one in the ith position is greater than a threshold s_{th}. Referring to Fig 6, the constraint can be expressed as $\sum_k d_k > s_{th}$.

To demonstrate the effectiveness of this construction, in Fig 7, we compare the performance of two IRA codes of rate-5/12 and length 4136 bits obtained by puncturing the mother code in Example 1. The first code is constructed using an S-random

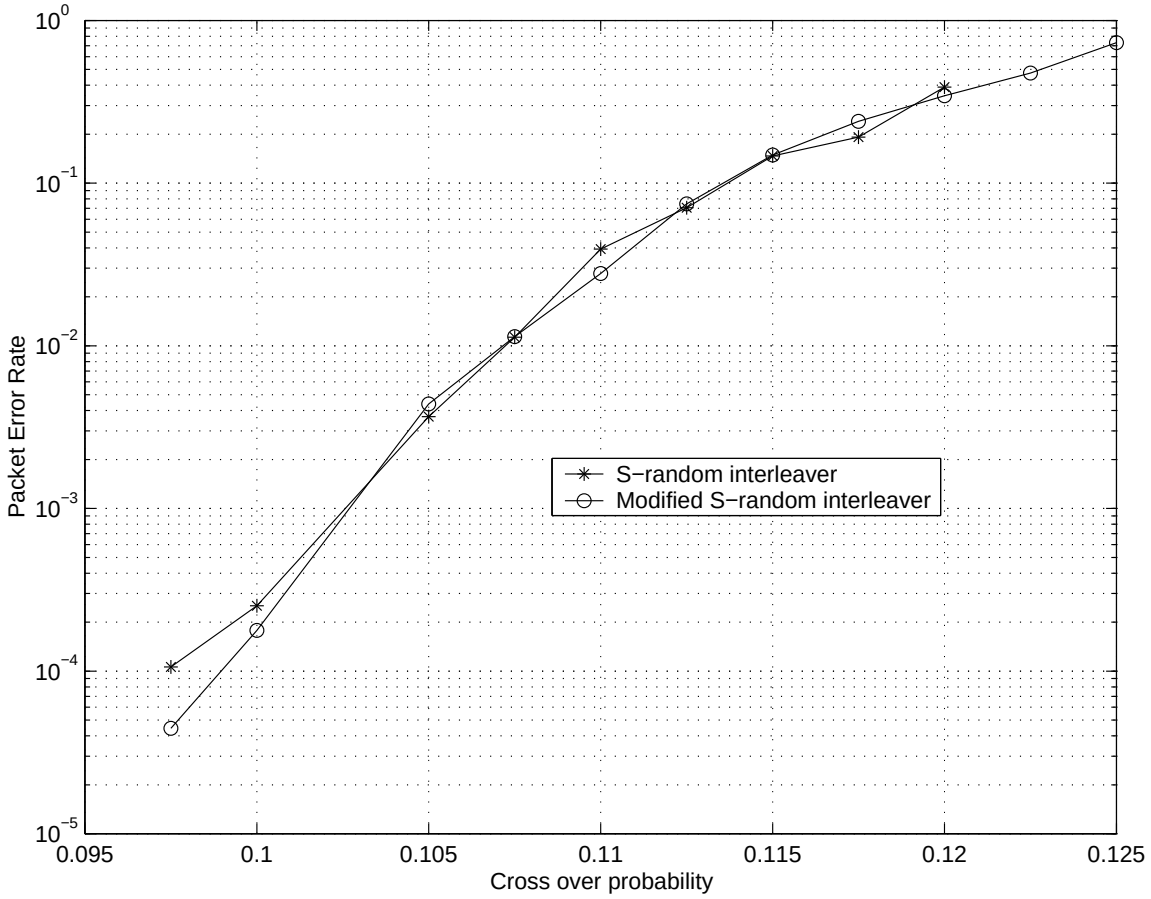

Fig. 7. Performance of IRA code punctured to rate 5/12 code with conventional and modified s-random inter-leaver

inter-leaver with $s = 68$ and the second code is constructed using the proposed modified $S-$random inter-leaver. Observe that the modified $S-$random inter-leaver helps reduce the effect of the floor in the packet error rate (PER) and performs as well as the $S-$random inter-leaver at higher PERs. Fig 8 shows the packet error rate for

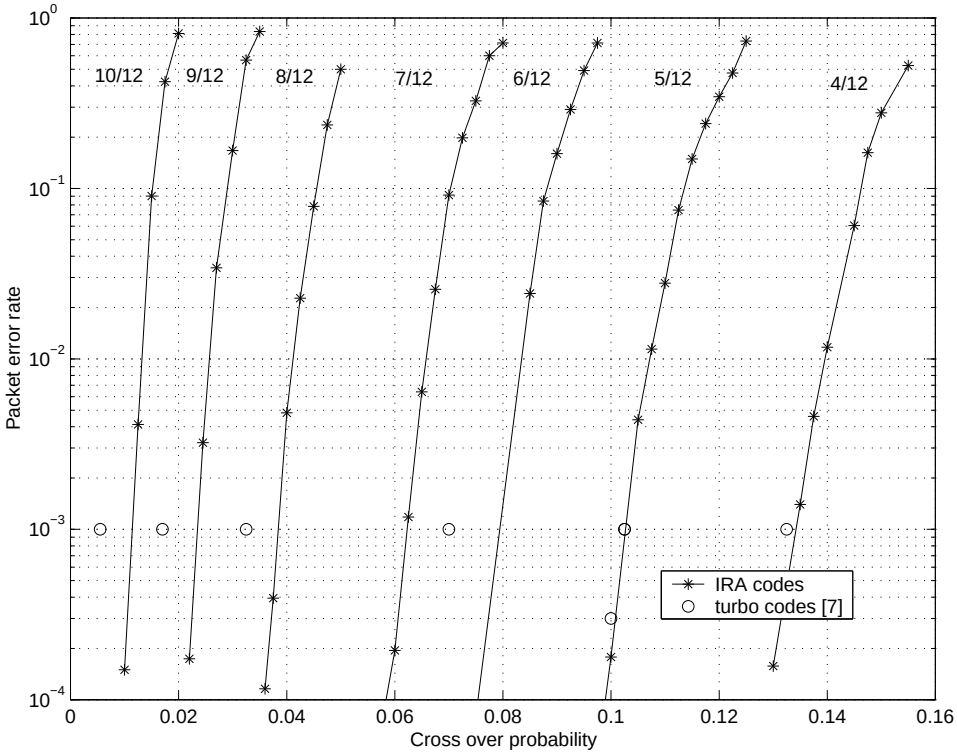

Fig. 8. Packet error rates of IRA codes (code length = 517 bytes). 80 decoding iterations used for each code.

the mother rate and the higher rate codes obtained by puncturing the parity bits. For rate 5/12, 6/12, 7/12, 8/12, 9/12 and 10/12 codes, the fractions of parity bits being punctured are 3/10, 1/2, 18/28, 3/4, 5/6 and 9/10, respectively. Comparing the performances of the proposed IRA codes with those of the turbo codes used in [28], we observe that the IRA codes we designed perform better and there are almost

no error floors for these codes in the tested PER range.

7. Decoding Complexity

A good estimate of the number of additions and lookups required for decoding can be obtained for both IRA codes and turbo codes. The sum product decoding algorithm for IRA codes is assumed to be implemented directly in terms of $\log(\tanh(L(x)/2))$, where $L(x)$ are the extrinsic LLRs. At an information node of degree i, we require $2i$ adds/subs. At a check node of degree j, we require $2j$ lookups and $(2j-1)$ add/subs. At a parity node, we require 2 adds. For a rate-1/3 mother code, there are twice the number of check nodes and parity nodes as that of information nodes. For an average information node degree of i and check node degree of j, the total number of ops (counting adds and lookups as the same) per information bit for rate-1/3 code is $2i + 8j + 2$. For the code in Example 1, this is approximately 68 operations. For the turbo code used in [28], the same complexity can be calculated as in [61] and seen to be 400 operations/information bit/iteration. Even taking into account the fact that IRA codes use 80 iterations, whereas only 20 iterations were used with turbo codes in [28], the decoding complexity is significantly lower for our IRA codes.

D. Simulation Results

In our simulations, scalable bit streams are generated off-line, using SPIHT, JPEG-2000, 3-D SPIHT and H.26L-based PFGS, respectively. Each source is encoded into an embedded bit stream only once and the operational R-D data are stored as look-up tables. The transmission "server" computes using fast algorithms (see VI)the rate allocation based on the R-D data, the channel condition and the target transmission rate. Then it generates CRC and IRA codes and packetizes the truncated bit stream

into equal-length packets. We choose the packet length to be 517 bytes in order to compare with the results in [28]. The length of source in a 517-byte-long packet for IRA code rates 4/12, 5/12, 6/12, 7/12, 8/12, 9/12 and 10/12 are 169, 212, 255, 298, 341, 384 and 427 bytes, respectively.

The BSC channel is simulated by randomly flipping the bit stream at the given bit error rate. The altered bit stream is then delivered to the receiver, where IRA and source decoding are sequentially performed. We assume that all IRA decoding errors can be detected by the CRC code and the channel coding rate for each packet can be correctly decoded. Finally, the reconstruction is compared with the original source to get the PSNR results. The overall simulation setup is illustrated in Fig. 9.

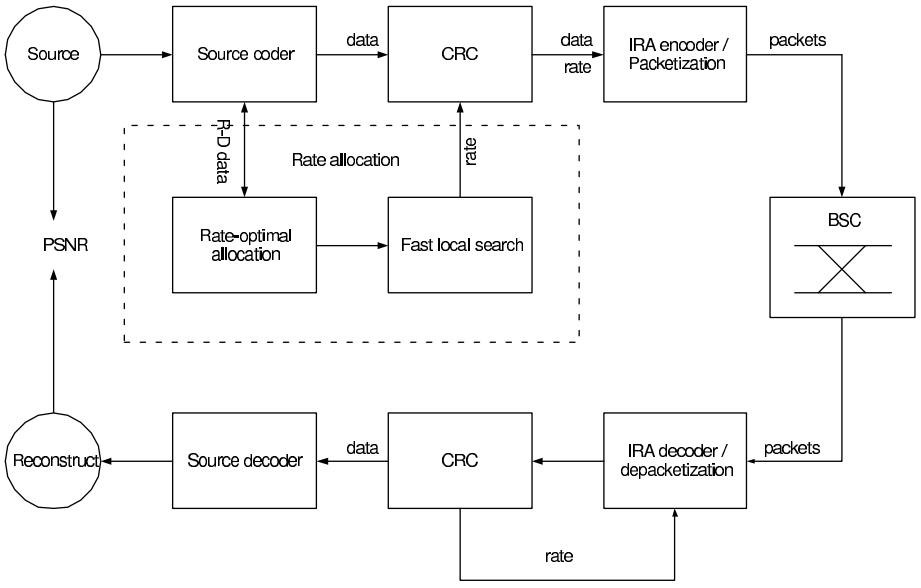

Fig. 9. Simulation system setup.

Table I presents the average MSE results in PSNR (dB) for three grayscale 512×512 images (*Lena*, *Goldhill* and *Barbara*) and four BERs. Each image is encoded

using JPEG-2000 in generic scalable mode and each result is averaged over 1,000 runs. We see that our results are uniformly better than those in [28], with more gains when the transmission rate is higher than 0.252 bpp. These gains stem from stronger protection provided by IRA codes over turbo codes. In addition, the availability of the rate 5/12 IRA code makes the performance improvement for BER=0.08 more pronounced than that for other BERs.

Table I. Results for JPEG-2000 encoded 512x512 images over four BSCs.

Image	Rate (bpp)	Channel BER			
		0.1 ([28])	0.08 ([28])	0.03 ([28])	0.01 ([28])
Lena	0.994	36.03(35.85)	36.26(36.15)	38.26(37.74)	38.93(38.78)
	0.505	33.13(32.76)	33.37(33.20)	35.48(35.15)	36.18(35.67)
	0.252	29.92(29.40)	30.24(29.94)	32.09(31.90)	32.75(32.56)
Goldhill	0.994	32.20(32.10)	32.30(32.24)	34.18(34.05)	35.00(34.81)
	0.505	29.99(29.89)	30.12(30.04)	31.65(31.38)	32.10(31.79)
	0.252	27.90(27.69)	28.01(27.88)	29.23(29.16)	29.74(29.64)
Barbara	0.994	30.77(30.51)	30.85(30.82)	33.74(33.54)	34.89(34.40)
	0.505	27.42(26.99)	27.66(27.54)	29.88(29.41)	30.84(30.35)
	0.252	24.73(24.71)	24.89(24.84)	26.37(26.32)	27.00(26.93)

Table II presents the results for SPIHT encoded images. Each result is an average over 1,000 runs. The results are compared to those in [29] where possible. We see that our results are uniformly better than those in [29] with a gain of up to 1.9 dB.

Our JSCC scheme can be easily extended to video transmission. Take 3D SPIHT as an example, the encoder blocks the video sequence into a series of Group of Frames (GOF) containing 16 frames per GOF. Each GOF is then encoded into an embedded bit stream. We process each GOF bit stream separately by applying our JSCC design to generate a fixed number of equal-length packets according to the target transmission rate. The R-D data of each GOF bit stream can be computed in advance and fed into the rate allocation algorithm. Or, for simplicity, one can use the average

Table II. Results for SPIHT encoded 512x512 images over four BSCs.

Image	Rate (bpp)	Channel BER			
		0.1 ([28])	0.08 ([28])	0.03 ([28])	0.01 ([28])
Lena	0.994	36.10(34.2)	36.39	38.24	39.12(38.0)
	0.505	33.21(31.1)	33.61	35.50	36.34(35.2)
	0.252	30.21(28.4)	30.70	32.32	33.16(32.0)
Goldhill	0.994	32.07(30.7)	32.35	34.25	35.11(34.0)
	0.505	29.92(28.6)	30.22	31.50	32.32(31.2)
	0.252	27.92(26.7)	28.30	29.21	29.88(29.0)
Barbara	0.994	30.00	30.26	33.12	34.36
	0.505	26.77	27.10	29.30	30.21
	0.252	24.25	24.64	25.99	26.70

R-D data for all GOFs. Fig. 10 shows the average PSNR results as a function of the transmission rate for 3D SPIHT encoded QCIF(176×144) video sequences *Akiyo* and *Foreman*. Each sequence contains 288 frames and the number of GOFs is 18.

In the case of H.26L-based PFGS, we assume that the base layer bit stream is transmitted losslessly and we only consider the transmission of the enhancement layer bit stream. We apply our JSCC algorithm to the enhancement layer bit stream of each frame based on the computation of an average R-D curve of the whole sequence. The simulation results for QCIF video sequences *Foreman* and *Akiyo* are shown in Fig. 11. The encoding frame rate is 10Hz and the quantization parameter (QP) is 25. The base layer bit rate for the two sequences are 31.95 kbps and 11.37 kbps, respectively.

E. Summary

We have presented a JSCC system for scalable image and video transmission based on scalable source coding and rate-compatible IRA code design. The set of IRA codes obtained by puncturing achieves rate-compatibility while maintaining a high

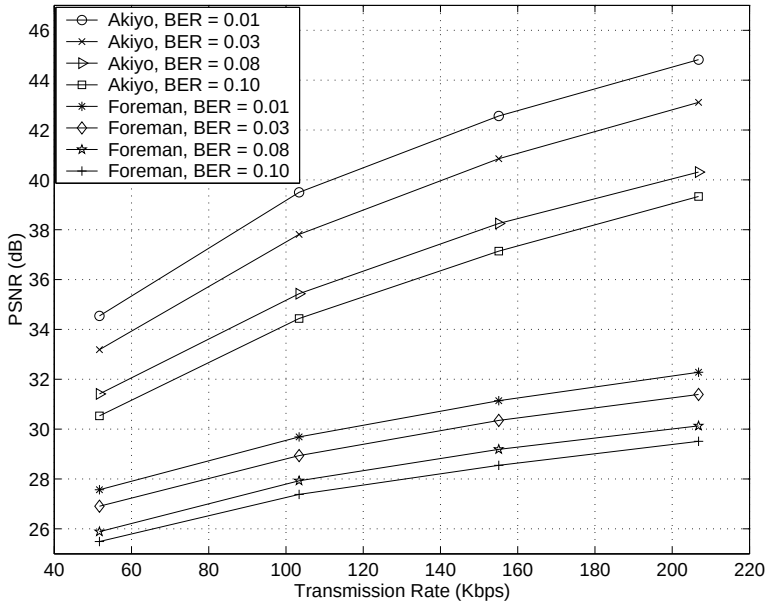

Fig. 10. Simulation of 3D SPIHT encoded video transmitted over four BSCs.

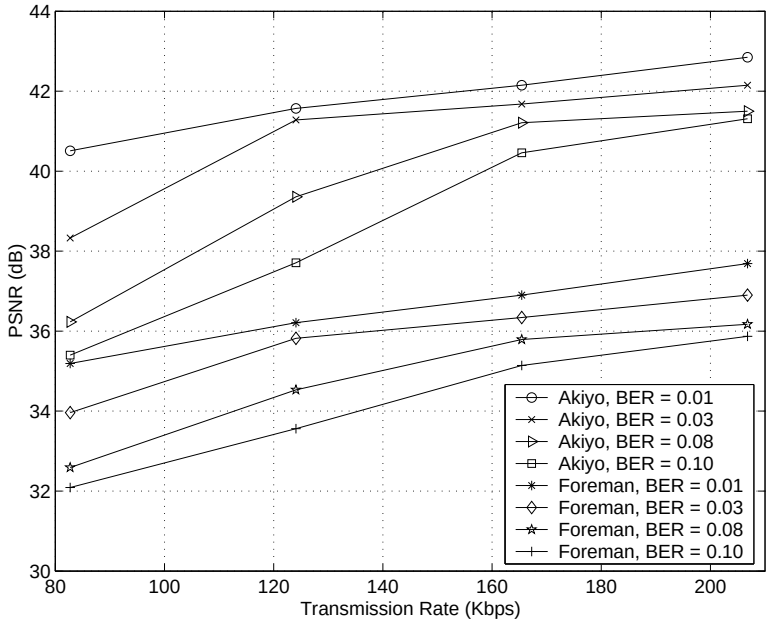

Fig. 11. Simulation results from transmitting the enhancement layer H.26L-PFGS bit stream of two sequences over four BSC channels. The base layer bit rates are 11.37 kbps for *Akiyo* and 31.95 kbps for *Foreman*.

performance and lower decoding complexity. By concatenating a rate-optimal solution with a modified local search algorithm (See APPENDIX), our error control scheme achieves compatible performance to the Viterbi algorithm used in [28], while the complexity is lower.

We applied this joint design to the transmission of scalable image and video transmission over BSC channels, using state-of-the-art source coders. Simulations show uniformly better results over those reported in previous works.

CHAPTER IV

SOURCE-OPTIMIZED IRREGULAR REPEAT ACCUMULATE CODES WITH INHERENT UNEQUAL ERROR PROTECTION CAPABILITIES AND THEIR APPLICATION TO IMAGE TRANSMISSION

In conventional data communication systems, channel codes are usually designed to minimize the probability of error. In multimedia communication systems, however, the compressed audio, image or video bitstreams are usually scalable or embedded, resulting a sequential dependency among the compressed data. This calls for unequal error protection (UEP) for improved end-to-end performance in multimedia communications. However, the common practice in achieving UEP in these systems is to design rate-compatible punctured channel codes before computing the UEP rate assignment separately. This paper proposes a new approach to designing powerful Irregular Repeat Accumulate (IRA) codes that are optimized for the multimedia source and to exploring the inherent irregularity in IRA codes for UEP. Using the end-to-end distortion due to the first error bit in channel decoding as the cost function, which is readily given by the operational rate-distortion function of embedded source codes, we incorporate this cost function into the channel code design process via density evolution and obtain IRA codes that minimize the average cost function instead of the usual probability of error. Because the resulting IRA codes have inherent UEP capabilities due to irregularity, the new IRA code design effectively integrates channel code optimization and UEP rate assignment, resulting in source-optimized channel coding or joint source-channel coding.

Using our source-optimized IRA codes for transporting SPIHT-coded images over the BSC, when the channel code length is long (e.g., with one codeword for the whole 512×512 image), we show that 1) the image transmission system can operate at

only 15% away from the channel capacity, thus achieving the best published results in terms of average PSNR; 2) the performance gain due to IRA code optimization for the source is marginal, reconfirming Shannon's separation principle. When the channel code length is is relatively short, source-optimized IRA codes perform much better.

A. Introduction

Shannon's separation principle [1] states that, asymptotically, source coding and channel coding can be done separately without performance loss. However, it was shown by Massey [62] that separation does not necessarily lead to less complex solution or is always applicable. Indeed, for the separation principle to hold, one has to tolerate infinite delay and complexity. Therefore, in practical multimedia communication systems with delay and complexity constraints, joint source-channel coding (JSCC) should be employed to provide acceptable performance.

Since the early 1990s, great progresses have been made in practical source code and channel code designs. On one hand, several wavelet-based image/video coding algorithms [22, 23, 24, 25, 26] have been developed. These algorithms have the distinct feature of being able to generate scalable (or embedded) bitstreams while offering superb coding performance. Scalability induces a sequential dependency among the compressed multimedia source bits, calling for unequal error protection (UEP) in the form of more protection for the beginning part of the bitstream, less for the middle part, yet still less or even no protection at all for the last part. This is different from conventional data (e.g., satellite) communications where all bits are equally important and hence equal error protection (EEP) suffices. On the other hand, turbo codes were invented in 1993 [6] and low-density parity-check (LDPC) codes [12] rediscovered in

1998 [13]. Representing LDPC codes with bipartite graphs, Richardson and Urbanke [14] were able to analyze LDPC codes with density evolution. Jin *et al.* [19] introduced a class of codes closely related to LDPC codes called Irregular Repeat Accumulate (IRA) codes in 2000 that combine the advantages of both turbo and LDPC codes. IRA codes have simple encoding algorithms and are amenable to exact Richardson-Urbanke analysis.

For JSCC, the SPIHT image coder and Rate-Compatible Punctured Convolution (RCPC) codes [63] were concatenated in [29] for image transmission over the binary symmetric channel (BSC). Despite the sequential dependency induced by the scalable SPIHT bitstream that calls for UEP, EEP was employed in [29]. For example, when the crossover probability the BSC is $p = 0.1$, a rate 2/7 RCPC code was chosen. Note that the channel capacity of the BSC is $C(p) = p \log_2 \frac{1}{p} + (1-p) \log_2 \frac{1}{1-p}$, and $C(0.1) = 0.469$. Using the rate 2/7 RCPC code only reaches 61% of the channel capacity in this case. Nevertheless, the scheme in [29] gave the best performance at its time because it took advantage of the high performance of the SPIHT source coder. The impressive results in [29] inspired many new approaches to JSCC (in the form of UEP of *embedded* multimedia bitstreams [64, 65, 66]) for improved performance. These approaches generally follow one of the two directions: channel-optimized source coding or source-optimized channel coding.

The aim of channel-optimized source coding is to design robust quantization schemes against errors introduced by the channel (see [67, 68]). Related works are channel optimized scalar quantization [69, 70], channel optimized vector quantization [71, 72, 73], trellis waveform coding [74], and channel optimized trellis-coded quantization [75].

Source-optimized channel coding tries to match the amount of introduced redundancy to the significance of the source bits. For scalable multimedia transmission,

this approach usually involves two steps: rate-compatible channel code design [63] and UEP optimization algorithms [55]. The former provides different levels of error protection for source bits of different importance; the latter performs optimal channel code rate assignment. To enable UEP, rate-compatible channel codes are usually designed beforehand and the best code rate assignment chosen algorithmically using dynamic programming [49], Lagrangian-based methods [52], the Viterbi algorithm [28], or a local search algorithm [45].

Toward UEP code design, the common practice is the use of RCPC. In this chapter, we propose an approach for the channel coding problems from the perspective of source coding. The channel coding problems we consider is to design IRA codes providing the best visual performance while they are used together with scalable source coders in image transmissions. With the inherent UEP capabilities of IRA codes, the channel codes are designed to be optimized over a cost function such as the average distortion, which is derived from the distortion-rate function of the source. Most source coders are designed with performances as close to the distortion-rate functions as possible. According to the distortion-rate function, the information bits causing more distortion need more protection. For example, state-of-the-art image coders (e.g., SPIHT [22] and JPEG2000 [23]) induces a sequential dependency among the compressed image bitstream, i.e., the arriving bits are only useful for the image decoder if previously arrived bits are correctly decoded by channel decoders assuming a natural order of transmission. Therefore, a single bit error in the previous bitstream will render the rest of the bitstream useless. In consequence, UEP is called for in the channel code design for scalable image transmission. Taking advantage of inherent UEP capabilities of IRA codes, we assign the cost determined by the location of the first error bit to the corresponding event where the decoder made an error, and calculate the average cost used as the criterion in the IRA code design. Therefore,

channel codes designed with this approach are not optimized for the probabilities of errors but for the average cost function, the average distortion as shown in this example. By the approach as described above, the optimization of channel codes and the choice of UEP schemes are embedded in the design process of IRA codes.

The problem of UEP code design was also considered in the literature. Huebner *et al.* [76] considered the serial concatenation of an array of repetition codes, an interleaver and turbo codes. With this scheme, different levels of protection are achieved by different numbers of repetitions. Caire and Biglieri [77] suggested the use of turbo codes to achieve UEP by 1) dividing the input to the encoder into different small blocks, which is designated to contain information of given class of importance. 2) keeping class separation while choosing a interleaver. 3)applying puncturing to parity bits according to the rate assigned to individual class. Vasic *et. al.* [78] [79] proposed a class of LDPC codes based on balanced incomplete block design (BIBD) and cyclic difference families (CDF), where the BIBD is defined on the union of orbits of the basic blocks of a CDF. By partitioning the set of the basic blocks into disjoint subsets, the point-block incidence matrix of the subset is the sub-matrix of the parity check matrix to match the desired protection level since the bits corresponding the sub-matrix are checked on a designed number of equations. However, these methods still need an algorithm to determine the best code rate and classes.

In the examples given in this chapter, long IRA codes are designed for SPIHT-coded images with cost functions based on the assumption that the events of the information bits of different positions being flipped are independent, and short IRA codes are designed for QCIF images with modified cost functions because the assumption does not hold true for short IRA codes. The operational distortion-rate function of the SPIHT coder is incorporated into density evolution to design IRA codes. The object of the design is to minimize the average distortion. A scalable image trans-

mission system is simulated with IRA codes designed with the described approach. In our simulation, CRC check bits are embedded in the SPIHT-coded bitstream in order to detect the first error. Our simulation results are uniformly better than those in the literature [29], [28], [80], [81]. It is shown that for larger codes (with one codeword for the whole 512×512 image), the simulated results of the codes designed with our approach are only 15% away the theoretical limit since larger IRA codes are capacity-approaching. Therefore, when compared with conventional IRA codes, the advantage of source-optimized IRA codes is not significant , which reconfirms Shannon's separation principle. For applications such as the transmission of QCIF images where smaller IRA codes are required and Shannon's separation principle is no longer applicable, source-optimized IRA codes perform much better.

The rest of the chapter is organized as follows. We first review in section 2 the encoding and decoding of IRA codes and the details of the density evolution used in the design of IRA codes. In section 3, we establish a cost model used in IRA code design. Section 4 presents the numerical results of the proposed method of designing IRA codes applied to the transmissions of sources with an exponentially decayed distortion-rate function, SPIHT-coded images and QCIF images.

B. The Theory of UEP Based on Irregularity

The assumption that UEP, in which the bit nodes with more neighboring check nodes are more protected, provided by IRA codes comes from the irregularity of IRA codes provides us a mean to design a source-optimized IRA code. Although this phenomenon can be easily observed in density evolution, we will verify the assumption in this section the assumption in a more theoretical setting.

The idea to verify this assumption is borrowed from [15] in which the idea is

originally used to prove the probability of errors is non-increasing with iterations. The same idea can be used to verify the assumption of UEP based on irregularity. Since IRA codes is a class of LDPC codes, without loss of generality, we treat information nodes and parity nodes as variable nodes collectively. For completeness, we restate the theorem and proof [15] in our notation here.

Theorem IV.1. *For LDPC codes with given degree profiles, the probability of errors in density evolution is non-increasing with the number of iterations l.*

Proof. Because of the assumption that there are no cycles in the graphs, the estimation of the a particular information bit at the l^{th} iteration is given by an estimator operating on a set of observations and these observations form a tree of depth l. Assume the number of iterations is l. For a particular information bit of degree k, $\mathbf{B}_k$, define B_k as the binary value carried by $\mathbf{B}_k$ and $m_{c\to v}^l$ as messages passed from its child (neighboring) nodes. Similarly, taking each one of these child nodes as a parent node, the message passed from its child nodes is given as $m_{v\to c}^{l-1}$ (excluding $\mathbf{B}_k$). Repeat the process until reaching leaf nodes. In this way, we construct a tree of depth l as shown in Fig. 12 and denote it as $\mathbf{T}_{\mathbf{B}_k}^l$.

Clearly, we have $\mathbf{V}_{\mathbf{T}_{\mathbf{B}_k}^{l-1}} \subset \mathbf{V}_{\mathbf{T}_{\mathbf{B}_k}^{l}}$, where $\mathbf{V}_{\mathbf{T}}$ is the collection of variable nodes in $\mathbf{T}$. Denote $\mathcal{V}_{\mathbf{T}}$ as the observations from channel carried by $\mathbf{V}_{\mathbf{T}}$. Thus we have

$$\mathcal{V}_{\mathbf{T}_{\mathbf{B}_k}^{l}} = \mathcal{V}_{\mathbf{T}_{\mathbf{B}_k}^{l-1}} \bigcup \mathcal{V}_l. \tag{26}$$

where $\mathcal{V}_l$ the observations from channel carried by leaf nodes. Under the assumption of independence, the message passing algorithms at the l^{th} iteration actually perform maximum *a posteriori* (MAP) estimation of B_k based on observations at variable

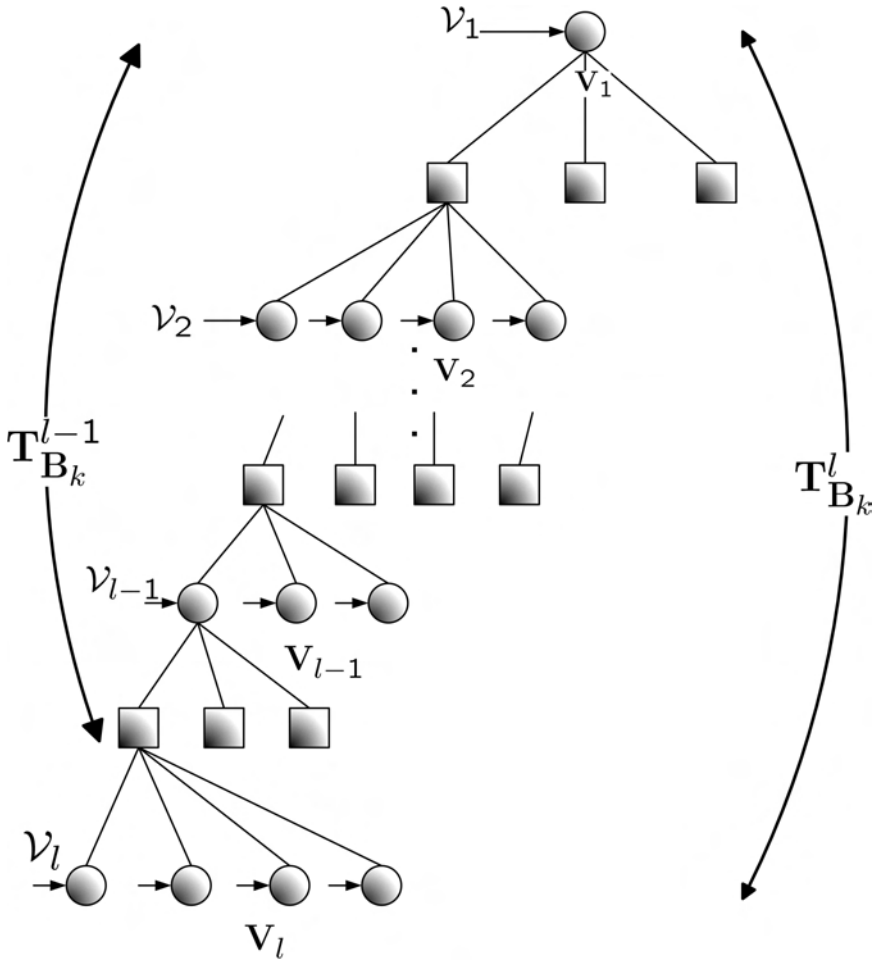

Fig. 12. Support tree for decoding B_k.

nodes of $\mathbf{T}^l_{\mathbf{B}_k}$, i.e.,

$$\begin{aligned}\hat{B^l_k} &= \mathtt{sgn}\left[\sum_{i=1}^{k} m^l_{c\to v} + m_c\right] \\ &= \mathtt{sgn}\left[\log\left[\frac{P\left(B_k = 0 | v_{\mathbf{T}^l_{\mathbf{B}_k}} \in \mathcal{V}_{\mathbf{T}^l_{\mathbf{B}_k}}\right)}{P\left(B_k = 1 | v_{\mathbf{T}^l_{\mathbf{B}_k}} \in \mathcal{V}_{\mathbf{T}^l_{\mathbf{B}_k}}\right)}\right]\right]. \end{aligned} \tag{27}$$

where B_k is the binary value conveyed by $\mathbf{B}_k$ and m_c is the message from channel. According to the MAP rule, $\mathcal{V}_{\mathbf{T}^l_{\mathbf{B}_k}}$ is partitioned as: $\mathcal{V}^0_{\mathbf{T}^l_{\mathbf{B}_k}}$ denoted as the region of observations in $\mathcal{V}_{\mathbf{T}^l_{\mathbf{B}_k}}$ such that $\hat{B^l_k} = 0$ and $\mathcal{V}^1_{\mathbf{T}^l_{\mathbf{B}_k}}$ denoted as the region of the observations in $\mathcal{V}_{\mathbf{T}^l_{\mathbf{B}_k}}$ such that $\hat{B^l_k} = 1$. Hence $\mathcal{V}_{\mathbf{T}^l_{\mathbf{B}_k}} = \mathcal{V}^0_{\mathbf{T}^l_{\mathbf{B}_k}} \bigcup \mathcal{V}^1_{\mathbf{T}^l_{\mathbf{B}_k}}$. Note that $P_e(v_{\mathbf{T}^l_{\mathbf{B}_k}})$ for $v_{\mathbf{T}^l_{\mathbf{B}_k}} \in \mathcal{V}^0_{\mathbf{T}^l_{\mathbf{B}_k}}$, the probability of errors for a MAP decoder, is not greater than $\frac{1}{2}$, since

$$\begin{aligned} & P(B_k = 0 | v_{\mathbf{T}^l_{\mathbf{B}_k}}) + P(B_k = 1 | v_{\mathbf{T}^l_{\mathbf{B}_k}}) = 1 \\ \Rightarrow\ & P(B_k = 1 | v_{\mathbf{T}^l_{\mathbf{B}_k}}) + P(B_k = 1 | v_{\mathbf{T}^l_{\mathbf{B}_k}}) \leq 1 \\ \Rightarrow\ & 2P_e(v_{\mathbf{T}^l_{\mathbf{B}_k}}) \leq 1 \\ \Rightarrow\ & P_e(v_{\mathbf{T}^l_{\mathbf{B}_k}}) \leq \frac{1}{2} \end{aligned} \tag{28}$$

and similarly for $v_{\mathbf{T}^l_{\mathbf{B}_k}} \in \mathcal{V}^1_{\mathbf{T}^l_{\mathbf{B}_k}}$. Furthermore, we have

$$H(B_k | v_{\mathbf{T}^l_{\mathbf{B}_k}}) \leq H(B_k | v_{\mathbf{T}^{l-1}_{\mathbf{B}_k}}) \tag{29}$$

because $\mathcal{V}_{\mathbf{T}^l_{\mathbf{B}_k}} = \mathcal{V}_{\mathbf{T}^{l-1}_{\mathbf{B}_k}} \bigcup \mathcal{V}_l$ and $\{v_{\mathbf{T}^{l-1}_{\mathbf{B}_k}}\} \subset \{v_{\mathbf{T}^l_{\mathbf{B}_k}}\}$, which is equivalent to

$$H(P_e(v_{\mathbf{T}^l_{\mathbf{B}_k}})) \leq H(P_e(v_{\mathbf{T}^{l-1}_{\mathbf{B}_k}})). \tag{30}$$

From (28) and (30), it follows that

$$P_e(v_{\mathbf{T}^l_{\mathbf{B}_k}}) \leq P_e(v_{\mathbf{T}^{l-1}_{\mathbf{B}_k}}). \tag{31}$$

Next, we have

$$\begin{aligned} E[P_e(v_{\mathbf{T}^l_{\mathbf{B}_k}})] &= \int_{v_{\mathbf{T}^l_{\mathbf{B}_k}} \in \mathcal{V}_{\mathbf{T}^l_{\mathbf{B}_k}}} P_e(v_{\mathbf{T}^l_{\mathbf{B}_k}}) f(v_{\mathbf{T}^l_{\mathbf{B}_k}}) dv_{\mathbf{T}^l_{\mathbf{B}_k}} \\ &\leq \int_{v_{\mathbf{T}^l_{\mathbf{B}_k}} \in \mathcal{V}_{\mathbf{T}^l_{\mathbf{B}_k}}} P_e(v_{\mathbf{T}^{l-1}_{\mathbf{B}_k}}) f(v_{\mathbf{T}^l_{\mathbf{B}_k}}) dv_{\mathbf{T}^l_{\mathbf{B}_k}} \\ &= E[P_e(v_{\mathbf{T}^{l-1}_{\mathbf{B}_k}})]. \end{aligned} \tag{32}$$

where $f(v_{\mathbf{T}^l_{\mathbf{B}_k}})$ is the joint probability density function of $\mathcal{V}_{\mathbf{T}^{l-1}_{\mathbf{B}_k}}$ and $\mathcal{V}_l$. Based on the argument above, we conclude that the probability of errors is non-increasing in the number of iteration l. □

Corollary IV.1. *For LDPC codes with given degree profiles and the number of iterations l, the probability of error for variable nodes $\mathbf{B}_{k+\Delta}$ of degree $k+\Delta$ is not greater than that for variable nodes $\mathbf{B}_k$ of degree k.*

Proof. Consider the estimation at the l^{th} iteration of B_k and $B_{k+\Delta}$ associated to $\mathbf{B}_k$ and $\mathbf{B}_{k+\Delta}$ of degree k and $k+\Delta,\ \Delta > 0$ respectively. The corresponding support trees for decoding B_k and $B_{k+\Delta}$ are shown in Fig. 13. Although two trees $\tilde{\mathbf{T}}^l_{\mathbf{B}_k}$ and $\hat{\mathbf{T}}^l_{\mathbf{B}_{k+\Delta}}$ stem from $\mathbf{B}_k$ and $\mathbf{B}_{k+\Delta}$ respectively, the statistics of the estimation at bit nodes in $\mathbf{V}^l_{\mathbf{B}_k}$ and $\mathbf{V}^l_{\mathbf{B}_{k+\Delta}}$ are the same for given degree profile. Therefore

$$E[P_e(v_{\tilde{\mathbf{T}}^l_{\mathbf{B}_k}})] = E[P_e(v_{\hat{\mathbf{T}}^l_{\mathbf{B}_{k+\Delta}}})].$$

Since $\mathbf{V}_{\hat{\mathbf{T}}^l_{\mathbf{B}_{k+\Delta}}} \subset \mathbf{V}_{\mathbf{T}^l_{\mathbf{B}_{k+\Delta}}}$, applying the same argument, we have

$$E[P_e(v_{\hat{\mathbf{T}}^l_{\mathbf{B}_{k+\Delta}}})] \leq E[P_e(v_{\hat{\mathbf{T}}^l_{\mathbf{B}_k}})],$$

and

$$E[P_e(v_{\tilde{\mathbf{T}}^l_{\mathbf{B}_{k+\Delta}}})] \leq E[P_e(v_{\hat{\mathbf{T}}^l_{\mathbf{B}_k}})].$$

We conclude that the variable nodes with more neighboring check nodes have prob-

abilities of errors not greater than those with less neighboring check nodes. □

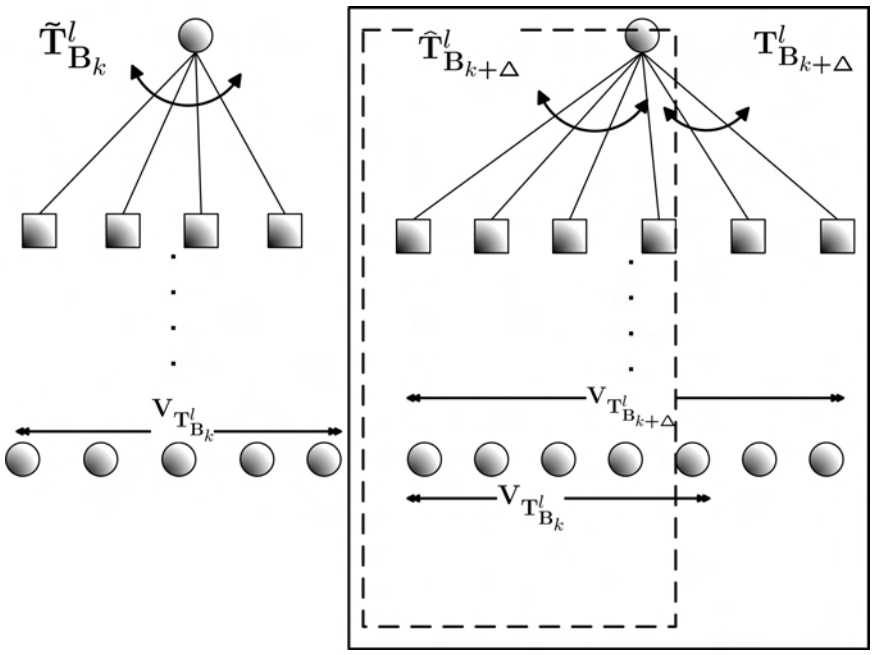

Fig. 13. Support trees for decoding B_k and $B_{k+\Delta}$.

C. IRA Code Design with Enhanced Unequal Error Protection

The conventional ways of designing graph-based channel code are either to find the codes with maximum threshold for a given rate or to find the codes with maximum rate for a given channel both of which are equivalent in a strict sense. With the traditional approaches, the design problem is formulated as

$$\min_{\lambda \subseteq \lambda_{R_c}} P_n(\lambda_1, ..., \lambda_d, a, U_{i,R_c}(x), U_{p,R_c}(x)) \tag{33}$$

where $\lambda_{R_c} \doteq \left\{ \lambda_i \middle| \sum_i \lambda_i = 1, \frac{\sum_i \lambda_i / i}{1/a + \sum_i \lambda_i / i} = R_c, \lambda_i \geq 0 \right\}$.

In this chapter, we consider designing IRA codes with optimal UEP. Because of the irregularity, we observe that the information bits connected to more checks

are more protected than those connected to fewer checks from density evolution and actual code simulation. We also give a proof in Section B to justify our observation. Therefore, by using this property of IRA codes and assuming a cost function, we are able to design a code to minimize the average cost. Before exploring this problem further, we shall assume that the event that the i-th bit is flipped is independent of the event that the j-th bit is flipped for $i \neq j$, which is assumed in the derivation of density evolution. To formulate the problem, we encode the bitstreams into a single channel codeword and assume a cost model for the outcomes from the decoder. In this model, the cost due to decoding errors is determined by the location of the first error bit assuming a natural order of transmission. With this model and the assumption of independence, we are able to establish the average cost as

$$\bar{C} = C_0 P_e(1) + \sum_{i=2}^{L} C_{i-1} P_e(i) \prod_{j=1}^{i-1} [1 - P_e(j)], \tag{34}$$

where C_i is the cost assigned to the event, in which the first bit error occurs at position i, and $P_e(i)$ is the error probability of that no errors before i-th bit and L is the length of the information bits. For the i-th information bit of degree k, the error probability is given as

$$P_e(i) = \int_{-\infty}^{0} \{*^k U_{c\to i}^n(x)\} * U_i(x)\ dx. \tag{35}$$

Therefore, the problem of code design can be formulated as :

$$\min_{\lambda \subseteq \lambda_R} \bar{C}(\lambda_1, ..., \lambda_d, U_{i,R_c}(x), U_{p,R_c}(x)). \tag{36}$$

It is usual in the literature to design the IRA codes with highest threshold when assuming infinite graphs. However, note that (34) are established upon the assumption of finite L. In this case, we look for codes with highest cost (threshold) under a

proper constraint on the allowed number of iterations in the density evolution.

The previous formulation of the problem is based on the assumption that the errors of the decision made by the decoders on the information bits of different positions occur independently. Although this assumption can be derived directly from the assumption of independence in density evolution, it is necessary to validate this assumption since the assumption of infinite graphs is not true in practice. Define V_i as the binary value carried by the i-th bit after decoding. For a specific error event, V_i is assigned a value of 1 if the i-th bit is flipped, otherwise V_i is assigned 0. Generally, to validate the independence between V_i and V_j, we need to verified that

$$f_{V_i,V_j}(v_i, v_j) = f_{V_i}(v_i) f_{V_j}(v_j) \tag{37}$$

where $f_{V_i}(v_i)$, $f_{V_j}(v_j)$ and $f_{V_i,V_j}(v_i, v_j)$ are (joint) probability density functions for continuous random variables or (joint) probability mass functions for discrete random variables. Note that (37) is true if and only if

$$E\{V_i^m V_j^n\} = E\{V_i^m\} E\{V_j^n\} \tag{38}$$

is true for all m and n.

For two binary random variables V_i and V_j, (38) becomes

$$\begin{cases} E\{V_i^m V_j^n\} = P_e(i,j), \\ E\{V_i^m\} E\{V_j^n\} = P_e(i) P_e(j). \end{cases} \tag{39}$$

Therefore, the simplest way to test the independence between V_i and V_j is to check if $P_e(i,j)$ and $P_e(i)P_e(j)$ are equal. Since no analytic approach can prove the assumption of independence, we can only resort to actually simulating a particular IRA code. The code we simulate has length of 10000 bits. It is shown in Fig. 14 the plot of $P_e(i)P_e(j)$ averaged over 10000 error events; and it is shown in Fig. 15 the plot of $P_e(i)P_e(j)$

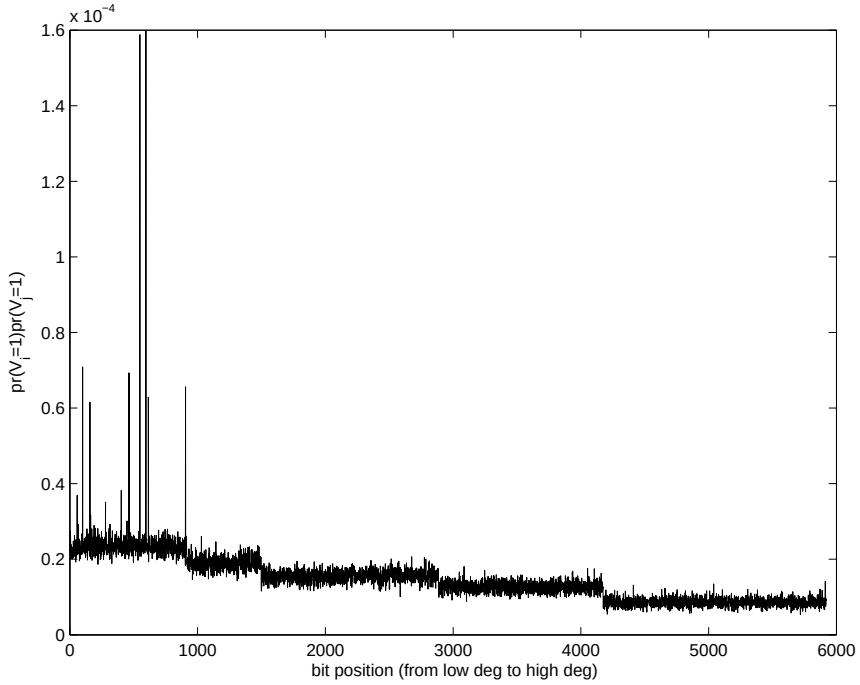

Fig. 14. $\mathrm{p}(V_i = 1)\mathrm{p}(V_j = 1)$, $i = 0, j = 1 \sim 5919$.

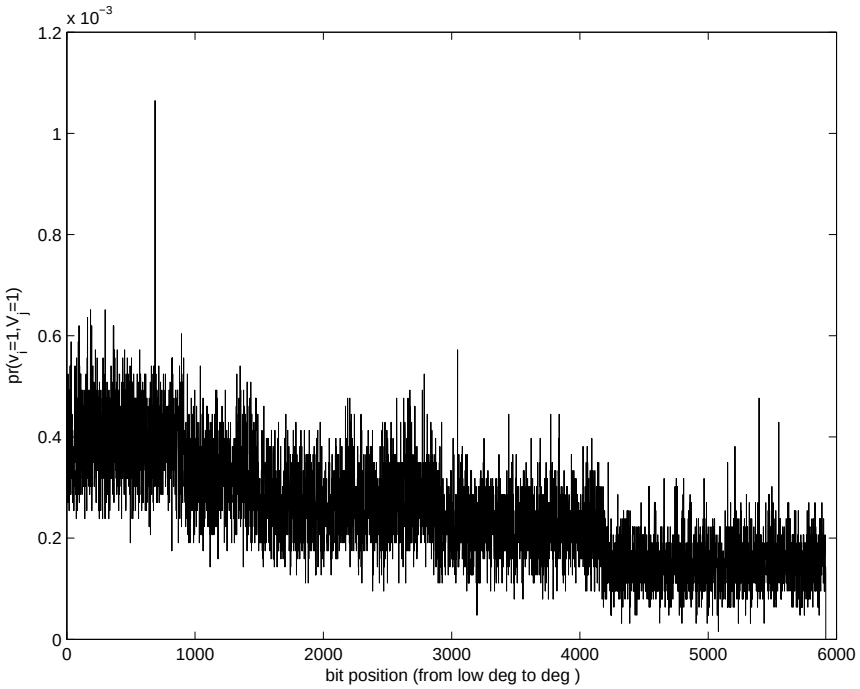

Fig. 15. $\mathrm{p}(V_i = 1, V_j = 1)$, $i = 0, j = 1 \sim 5919$.

averaged over 10000 error events. In these two plots, we see that $P_e(i,j) > P_e(i)P_e(j)$, which means the probability that the j^{th} bit is flipped is greater than $P_e(j)$ given the i^{th} bit decoded wrong. Therefore, for the finite-length IRA codes, the assumption is not true and thus (34) needs to be modified. One way to deal with the dependence is to compute the average distortion only for the error events where the first error bits have high degrees and the corresponding distortion can be approximated as $\bar{C} \approx C_0 P_e(1) + \sum_{i=2}^{L} C_{i-1} P_e(i) \prod_{j=1}^{i-1}[1 - P_e(j)]$. The reasons for this approximation are twofold:

First, we want the distortion due to the error events, where first error bits are those of high degrees, as small as possible if the overall distortion is mostly made up by the distortion due to these error events. On the other hand, if the distortion due to this type of errors is not a major contributor to the overall distortion, we also need the distortion due to this type of errors to be small to avoid hiking up the overall distortion. The bottom line is the distortion due to this type of error events should be as small as possible.

Second, intuitively the source-optimized channel codes should be only slightly worse than the conventional channel codes in terms of error-correcting performances. Hence, even after taking the dependence into account, $P_e(j)$ that the decision from decoders on the j^{th} bit is wrong, is still much smaller than 1 given that the j^{th} is a high degree node and can be approximated as $P_c(j) \doteq 1 - P_e(j)$ given that the decisions made by the decoder on the k^{th} bit $k = 1 \sim i-1, k \neq j$. Therefore,

$$\begin{aligned} P_c(1,2,\ldots,i-1) &\approx P_c(1)P_c(2)\ldots P_c(i-1) \\ &= (1-P_e(1))(1-P_e(2))\ldots(1-P_e(i-1)) \end{aligned} \tag{40}$$

In consequence, we can approximate the joint probability with (40) in the calculation

of the overall distortion.

D. Design Results

1. IRA Codes Design Results

Four IRA codes were designed according to the distortion-rate function of the SPIHT coder. These codes, designed for binary symmetric channel (BSC) with $p = 0.1$ and $p = 0.03$, have codewords of length 66060 and 132380 bits, respectively. When the transform coding operated at rate $R < 1$, by separating the coefficients quantized to zero from those that are greater than zero, it is shown by Mallat and Falzon [82] that the distortion-rate function is governed by

$$D(R) \sim CR^{1-2\gamma} \tag{41}$$

where γ is the negative of the speed of decay of logarithm of sorted coefficients after transformation, which can be approximated as 1 for most natural images and C is constant. Therefore, the distortion-rate function for SPIHT coders at $R < 1$ is proportional to R^{-1}. Using density evolution and (34) as the criterion instead of the probability of errors in IRA codes design, we obtain the profiles of IRA codes shown in Table III for four scenarios.

To validate the effectiveness of our method, another set of codes designed to minimize the probabilities of errors are introduced for comparison. Code A, which is designed for $p = 0.1$ and $L = 66060$ in table III, and Code B, which is designed to minimize probabilities of errors with the same p and L, are simulated. The rates of both codes are 0.46. In order to make a convenient comparison with the experiment results, Table IV lists the profiles of both codes in terms of node perspective, where the length of information bits, 30147, is used in calculating the node perspective

Table III. Profiles of IRA codes of different code length $L's$ designed for BSC with crossover probability p's. Note: λ_i is the fraction of nodes of degree i.

Code A	Code B	Code C	Code D
$p = 0.1$		$p = 0.03$	
$R_c = 0.46$	$R_c = 0.4705$	$R_c = 0.71$	$R_c = 0.73$
$L = 66060$bits	$L = 132380$bits	$L = 66060$bits	$L = 132380$bits
$\rho(x) = x^3$		$\rho(x) = x^8$	
$\lambda(x) = \sum_i \lambda_i x^{i-1}$		$\lambda(x) = \sum_i \lambda_i x^{i-1}$	
λ_2= 0.4373	λ_2=0.4501	λ_2=0.2689	λ_2=0.7530
-	λ_3=0.0193	λ_3=0.7287	-
-	-	-	λ_4=0.2469
λ_5= 0.2534	-	λ_5=0.0007	-
-	λ_7=0.5261	-	-
λ_9= 0.1951	-	-	-
-	-	λ_{10}=0.0005	-
-	-	λ_{14}=0.0001	-
-	-	λ_{15}=0.0001	-
-	-	λ_{18}=0.0002	-
λ_{20}= 0.1141	-	λ_{20}=0.0001	-
-	-	λ_{22}=0.0001	-
-	-	λ_{24}=0.0005	-

Table IV. Profiles of Code A (source-optimized)and Code B (conventional) in terms of node perspective.

$\lambda(x) = \sum_i \lambda_i x^{i-1}$	$\lambda(x) = \sum_i \lambda_i x^{i-1}$
Code A	Code B
λ_2= 20637	λ_2=15673
-	λ_3=5821
-	λ_4=1834
λ_5= 5978	-
-	λ_6=5794
λ_9= 2762	-
-	-
-	-
-	-
-	-
λ_{20}= 796	λ_{20}=1025

profile. The information bits are arranged according to their importance, in a way that the more important parts of the bitstream are assigned to the information bits with higher degrees. In constructing both codes, we use S-random inter-leavers with the same spreading factor $s = 205$. The block error rate of the Code A is 8.06×10^{-4}. The block error rate of the Code B ,which is expected lower than that of Code A, is 4.23×10^{-4}. We also recorded the first error bit for Code A and Code B and calculated the average cost. The cost function is $1/R$, where R is the number of the information bits before the first error bit divided by 262144 (the number of image pixels). After Mapping the number of information before first error (if any) to the corresponding bit rate and calculating the cost, we obtain the average cost 8.7322 for Code A and 8.8527 for Code B, respectively. Although Code B has lower probabilities of errors, the number of correct information bit before the first error bit is lower than that of code A where errors occur. This effect degrades the overall performance. because the cost increases dramatically at low R and this part of information needs more protection.

2. Applications to Scalable Image Transmission Systems

We use these four codes in the simulation of the scalable image transmission system with the SPIHT coder. The images considered here are Lena, Goldhill and Barbara, which are 512×512 in dimension and 8-bit in resolution. The lengths of the IRA codes correspond to total bit rates of 0.252 and 0.505. In our simulation, we assign CRC parity check bits in order to be consistent to that in [28]. Specifically, bitstream are divided into units and each consists of CRC check bits and information bits in total 4136 bits. We assume that errors can be detected perfectly and allocated with the employment of CRC parity check bits. If an error is detected by some CRC error check bits, the information bits following the check bits are discarded by the

source decoder. Tables V and VI show simulation results of the proposed schemes

Table V. Average PSNR in dB for 512x512 images over BSC with crossover probability 0.1 and 0.03 and total transmission rate at 0.252 bpp.

		Lena	Goldhill	Barbara
p=0.1	[29]	28.40	26.70	N/A
	[28]	29.40	27.69	24.71
	[80][81]	30.21	27.92	24.25
	New	30.68	28.25	24.49
	Limit	31.39	28.65	25.11
p=0.03	[29]	N/A	N/A	N/A
	[28]	31.90	29.16	26.32
	[80][81]	32.32	29.21	25.99
	new	32.74	29.44	26.18
	Limit	33.22	29.89	26.71

for these four scenarios. For each image, each result is averaged over 2000 runs and our IRA code rates are chosen such that at most one decoding error occurs in every 1000 runs. The results from [29],[28], [80] and [81] are also included for comparison. Note that JPEG2000 used in [28] usually has higher PSNR than SPIHT at the same rate. Finally, we give the limiting performances which can be computed as (in dB) $10 \times \log_{10} \frac{255^2}{D(RC)}$, where $D(.)$ is the operational distortion-rate function of the SPIHT coder, R is the total transmission rate and $C = 1 + p \log_2 p + (1-p) \log_2 (1-p)$ is the channel capacity of BSC with crossover probability p. It is readily seen from Tables V and VI that our results are uniformly better than those in [29], [28], [80] and [81]

and are only $0.40 \sim 0.73$ dB away from the theoretical limits.

Table VI. Average PSNR in dB for 512x512 images over BSC with crossover probability 0.1 and 0.03 and total transmission rate at 0.505 bpp.

		Lena	Goldhill	Barbara
p=0.1	[29]	31.10	28.60	N/A
	[28]	32.76	29.89	26.99
	[80][81]	33.21	29.92	26.77
	New	33.79	30.31	27.31
	Limit	34.43	30.79	27.88
p=0.03	[29]	N/A	N/A	N/A
	[28]	35.15	31.38	29.41
	[80][81]	35.50	31.50	29.30
	new	35.82	31.80	29.68
	Limit	36.31	32.24	30.17

For IRA codes with long code length, the performances are very close to the Shannon limit. A small sacrifice in channel coding rates can boost up the performance in terms of the probability of errors and will not cause too much decrease in PSNR. On the other hand, for shorter codes, a small increase in channel coding rate can enhance the performances in term of PSNR when the source coder is operated at low bit-rates. Therefore, it is desirable to slightly increase the channel coding rate in some applications in which the requirements of the probability of errors are not so restricted or people do not get annoyed to view bad images once a while. Moreover, higher probability of errors, i.e., more error events, also helps showcase the advantage of our

source-optimized IRA code design methodology. Considering this scenario, we design two IRA codes for SPIHT-coded QCIF (176×144) images with code length 6387 bits for BSC with crossover probability 0.1. The distortion-rate function is replaced by one generated by actual encoding and decoding an image *mad*. The IRA code designed to minimize the average distortion has the degree profile $0.0952x^1 + 0.3370x^3 + 0.5678x^6$. The degree profile of the IRA code designed to minimize the probability of errors is $0.0657x^1 + 0.0628x^2 + 0.1998x^3 + 0.2322x^4 + 0.4395x^6$. Here we denote the first code as Code E and the latter as Code F. Both codes have the same $\rho(x)$ and rate, which are x^3 and 0.46, respectively. When two codes compared in terms of the probabilities of errors, the block error rates of Code E and Code F are 6.27×10^{-2} and 3.85×10^{-2}, respectively. Comparing the quality of the decoded image, we obtain the average PSNR 30.6920 dB as Code E is used to transmit the SPIHT-coded *mad*. However, we can only obtain the average PSNR 29.8161 dB with Code F. If no decoding errors, the PSNR is 31.4600 dB. Therefore, there is about 0.7680dB degradation in PSNR due to decoding errors. In the case that errors are detected, the average PSNR's 25.3760 and 20.3622 dB are obtained with Code E and Code F respectively, i.e., Code E provides about 5 dB gain when errors are detected. The cumulative distribution function of the locations of the first error bits for Code E and Code F are shown in Fig. 16. From these results and the profiles of Code E and Code F, ninety percent of first error bits are those of degree 2 for Code E. However, only fifty-five percent of first error bits are those of degree 2 for Code F. Figs 17 and 18 show the frequency

(normalized by the maximum of the frequencies) v.s. bit position (from high to low degrees) of Code E and Code F. In the figures, it is clear that both codes have uep capabilities against errors. However, the frequencies above the bit 1000 of Code E is lower than that of Code F because of the cost function used in the code design. This

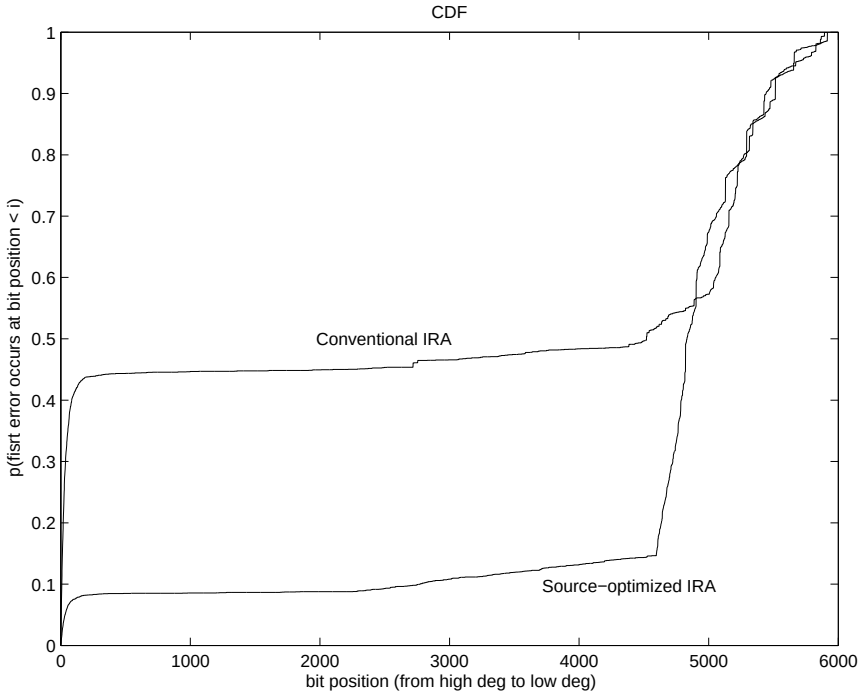

Fig. 16. Cumulative distribution function of the first error bit of Code E (source-optimized) and Code F (conventional).

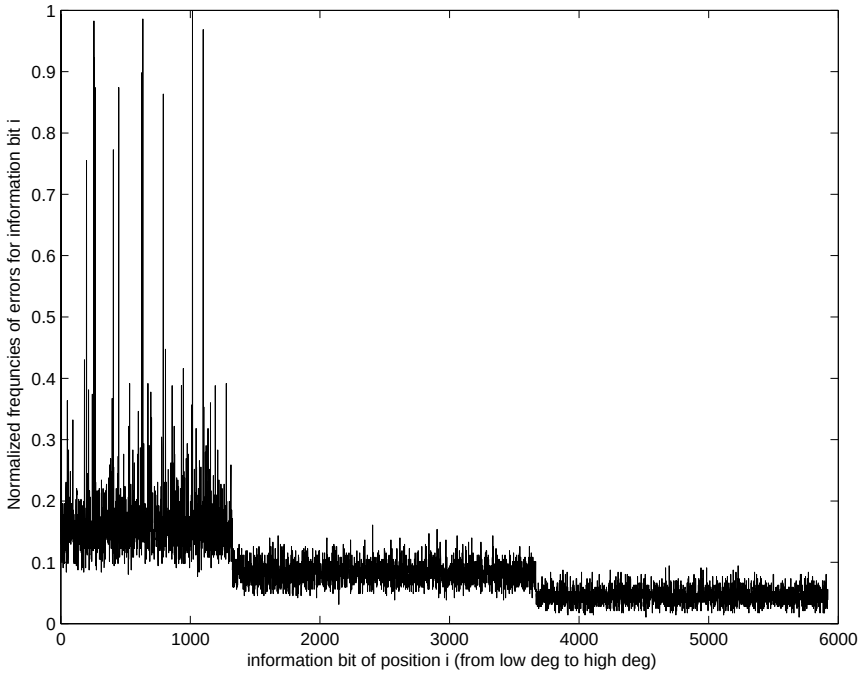

Fig. 17. Normalized frequencies of errors of information bits of Code E (source-optimized IRA).

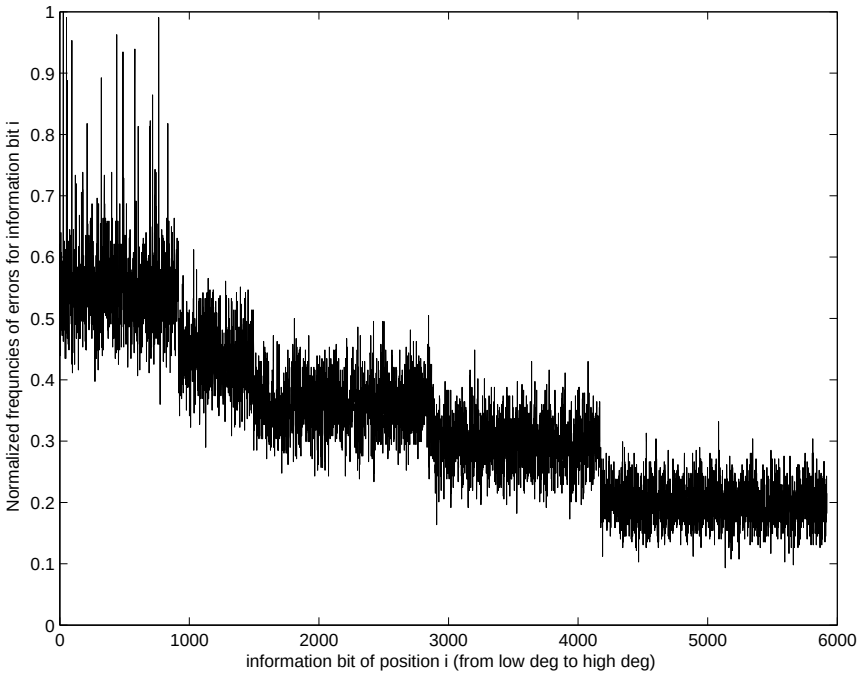

Fig. 18. Normalized frequencies of errors of information bits of Code F (conventional).

justifies the idea to use the average distortion instead of the probability of errors in the channel code design.

3. Discussion

It is shown by Lastras and Berger [83] that, for an i.i.d source and for squared error and for two steps of descriptions, the excess source rates (greater than zero) are upper-bounded by 0.5 bits per sample and the excess source rate for the finer description is upper-bounded by 1 bit per sample if the excess source rate for the coarser description is zero, which means that all sources are almost successively refinable. Therefore, we expect a well-designed scalable source coder will deliver advantageous performance over a variety of sources against other source coders. However, the corruption in some part of compressed bitstream could cause serious degradation in the quality of the delivered content. Therefore, the key of the application of embedded source coding

in noisy channel environment is the design of the policy by which the redundancies are assigned the packets. The scheme presented above take the whole image as input, which is divided into smaller packets, to the encoder of a systematic IRA code. The packets are given different levels of protection by mapping the bit in the packets to information bits connected to a designed number of edges in IRA codes.

From the comparison between Code A and Code B, the difference in the source-optimized and conventional IRA codes is not much because the code length of both codes is large and hence the performance of the IRA codes optimized for conventional criterion are capacity-approaching. In this sense, the conventional IRA codes are source-optimized. The reason is that, for IRA codes with long block lengths, the advantage of source-optimized IRA codes is less significant since the conventional IRA codes can be designed very close to the capacity with arbitrarily low error rates and hence, asymptomatically, the IRA codes optimized with conventional criterion are the best source-optimized IRA codes as separation theorem suggests. From Table V and Table VI, with higher total transmission rates, the proposed scheme provides more gain against the conventional schemes partially due to the interleaving gain. For smaller IRA codes, as in the case of QCIF images, we see considerable gain over the conventional IRA codes due to the use of source-optimized IRA codes. The general rule of thumb is to apply separation principle is when the total transmission rates and source rates are adequately high. Therefore, this proposed strategy of JSCC works better when the code lengths are relatively short or where the separation principle is no longer valid.

E. Summary

We present a method of designing IRA codes for sources characterized by cost functions. By assuming a cost function and incorporating this cost function into channel code design, we obtain source-optimized IRA codes. When applied to scalable image transmissions, the procedure to optimize IRA codes automatically decides the best UEP scheme (degree profiles) when the cost function is distortion-rate function. We justify this idea by designing IRA codes for SPIHT-coded image sources. Simulations show that our scheme outperforms conventional approaches and our results are $0.40 \sim 0.73$ dB away from the theoretical limit. Our design achieves source-optimized channel coding or true joint source-channel coding. This is different from usual combined source-channel coding designs.

CHAPTER V

SLEPIAN-WOLF CODING OF MULTIPLE M-ARY SOURCES USING LDPC CODES

We consider Slepian-Wolf coding of multiple m-ary sources. We show how Low Density Parity Check (LDPC) codes can be used for Slepian-Wolf coding of multiple equiprobable, memoryless and correlated m-ary sources with rates close to the theoretical limit. Applying the syndrome concept, we compress multiple correlated m-ary sources using multilevel codes, which is originally developed in the context of channel coding problems. With proper chosen rates for compression of each source, decoding can be achieved using multiple stage decoding. LDPC codes are suitable for such an application because the correlation model can be used in the LDPC code design. Therefore, multilevel LDPC codes can be employed in the case of coding multiple m-ary sources. Examples of code design for four binary sources and two ternary sources are given with simulations results that are very close to the theoretical limits.

A. Introduction

Driven by the severe physical constraints in sensor networks, there has been great interests in developing distributed signal processing algorithms to leverage the limited power in sensors to achieve maximal system performances. One of such examples is lossless distributed data compression, which was considered by Slepian and Wolf [3]. The Slepian-Wolf theorem determines the region of achievable rates for the compression of two correlated sources X_1 and X_2 separately as shown in Fig. 19 with

---

This chapter is joint work with Mr. A. Liveris, who is a fellow graduate student in the Department of Electrical Engineering, Texas A&M University. The author would like to thank Mr. Liveris for allowing inclusion of his work on correlation models for the completeness of this chapter.

arbitrarily small error probabilities if a joint decoder is employed. With X_1 available at the decoder, the achievable rate R_2 of lossless compression of X_2 (often referred to as lossless compression with side information) is lower-bounded by the conditional entropy $H(X_2|X_1)$, i.e., $R_2 \geq H(X_2|X_1)$. The achievable rate region is shown in Fig. 20. This theorem can be extended to the case where multiple sources are involved [2][84]. The achievable rate region for the lossless distributed compression of n correlated sources X_l's for $l \in \{1, 2, ..., n\}$ is given by

$$R(\mathbf{S}) > H(X(\mathbf{S})|X(\mathbf{S}^c)), \tag{42}$$

where R_l is compression rate for X_l and, for all $\mathbf{S} \subseteq \{1, 2, ..., n\}$, $R(\mathbf{S}) = \sum_{l \in \mathbf{S}} R_l$, $X(\mathbf{S}) = \{X_l : l \in \mathbf{S}\}$ and $S^c = \{1, 2, ..., n\} \setminus S$. The Slepian-Wolf theorem has been

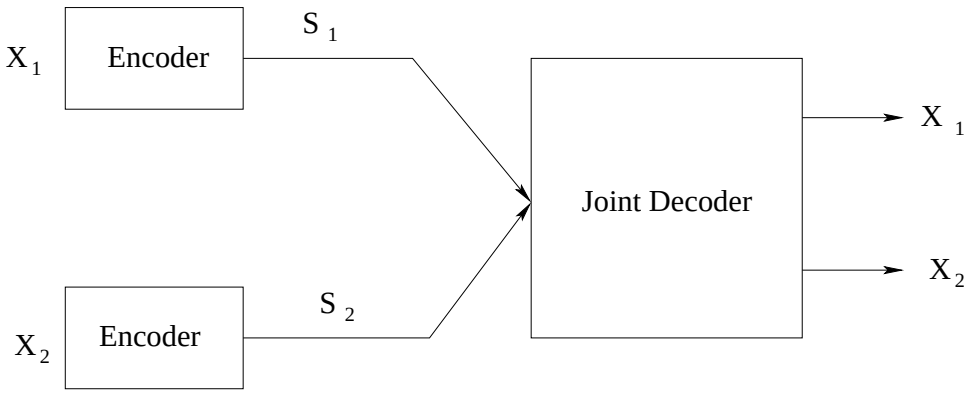

Fig. 19. The Slepian-Wolf problem.

known for a long time, but it was only recently that a practical scheme exploiting the potential of the Slepian-Wolf theorem was introduced by Pradhan and Ramchandran [85] based on channel codes like block and convolutional codes. Their setting provides a framework for asymmetry Slepian-Wolf coding of two i.i.d. correlated sources, i.e., the operating points are the corner points on the achievable rate region. In this

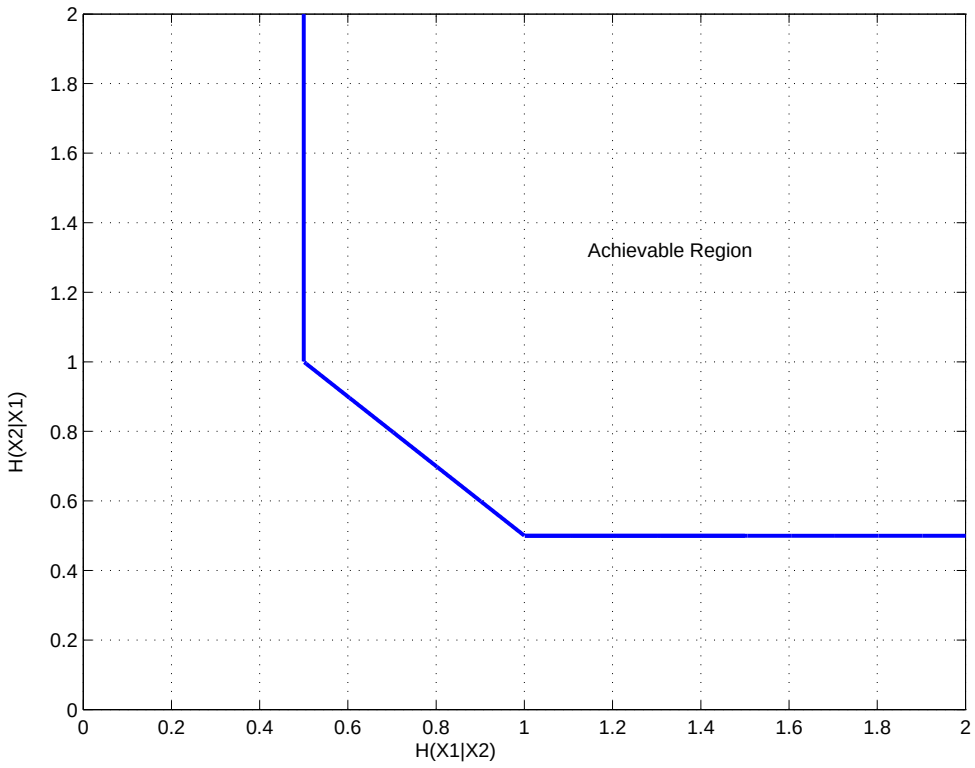

Fig. 20. Achievable rate region for Slepian-Wolf coding of X_1 and X_2.

regime, better results by using more powerful channel codes such as turbo and LDPC codes have been reported in [31, 86, 87, 89, 90, 91]. For operating points other than corner points, practical constructions using LDPC, irregular repeat accumulate (IRA) and turbo codes have been proposed by Garcia-Frias [86], Schonberg *et al.* [92, 93] and Stanković *et al.* [94]. With puncturing, Garcia-Frias used two turbo codes to compression two sources, trying to achieve joint entropy. Encoding in his proposed scheme is the same as the conventional turbo codes. Decoding is done by passing interleaved version of soft output of one turbo decoder to the other. Since correlation was not taken into consideration in code design, their results did not approach Slepian-Wolf bound. The basic idea in [93] and [94] is to partition a parity check matrix, which can be explained with factor graphs. In the scheme proposed by Schonberg *et al.*, for

symmetry coding of two sources, the variable nodes of two factor graphs corresponding to two sub-matrices of the a single parity check matrix are connected with additional nodes, which model the correlation between two sources. The approach has the advantage that sources can have any probability distribution. However, it is difficult to design with density evolution although the code has a simple graph representation. With the concept of constructing a single code based on independently designed sub-codes developed by Pradhan and Ramchandran [95], Stanković *et al.* provided a solution based on systematic codes such that powerful channel codes like turbo and IRA codes can be employed to achieve any point on Slepian-Wolf bound. An extension of the method to Slepian-Wolf coding of multiple sources was also considered by Stanković *et al.* and they concluded that a single channel code can be used to achieve the limit as long as the correlation between the sources satisfy that their sum is a Bernoulli-p process. They showed LDPC designed with their method can approach Slepian-Wolf limit. This construction has its own limitation too. First, it is only shown that the construction achieves the Slepian-Wolf limit for less general correlation models. Secondly, it assumes the sources are equiprobable. Therefore, these two reasons limit its applicability. In conclusion, it is difficult to design a limit-approaching code for symmetric Slepian-Wolf problems for general correlation models and sources with general probability distributions. In consequence, in this chapter, we shall focus on asymmetric Slepian-Wolf coding problem using limit-approaching LDPC codes.

The main novelty in this chapter is the application of LDPC codes to Slepian-Wolf coding of multiple m-ary sources with performance very close to the theoretical limit. The advantage of LDPC codes is that they can be designed for different correlation models between source outputs and the side information. In [31] and [30], with the exploitation of the correlation between two and three binary sources, it had been

shown that it is possible to approach the Slepian-Wolf limits with appropriate LDPC code designs. For Slepian-Wolf coding of three sources, a design rule of rates for coding each source was proposed in [30]. This rule not only facilitates code design but also allows multi-stage decoding.

The approach in [30] can be extended to n m-ary sources. In the case of compression of n binary sources, ($H(X_1)$, $H(X_2|X_1)$,...,$H(X_n|X_1, X_2, \ldots, X_{n-1})$) is a corner point of the region of achievable rates, for which

$$\sum_{l=1}^{n} H(X_l|X_1, \ldots, X_{l-1}) = H(X_1, \ldots, X_n) \tag{43}$$

from the chain rule for the joint entropy. Our work first focuses on four binary sources (i.e., $n = 4$ and $m = 2$) and we design codes with rate R_4 close to $H(X_4|X_1, X_2, X_3)$, assuming that there are already codes of rates R_1, R_2 and R_3 available that approach the limits $H(X_1)$, $H(X_2|X_1, X_2)$ and $H(X_4|X_1, X_2, X_3)$, respectively. This can easily be generalized based on the concept that the code of rate R_l used for the l^{th} source comes close to the i^{th} term in the summation of the chain rule, i.e., $H(X_l|X_1, X_2, \ldots, X_{l-1})$.

It is also possible to use non-binary LDPC codes as component codes in this setup. In the case of compression of two ternary sources (i.e., $n = 2$ and $m = 3$), a ternary LDPC code is carefully designed with EXIT-charts method [96] based on Davey and MacKay's decoding algorithm [97]. Our designs and simulations show that LDPC codes are able to approach the corresponding theoretical limit in each of the two cases.

The rest of the chapter is organized as follows. We first present in Section B a framework of Slepian-Wolf coding using LDPC codes. Section C presents a general structure of code for Slepian-Wolf coding of multiple m-ary sources using a multilevel

code based on LDPC codes. By assuming symmetry in the source correlation, Slepian-Wolf coding of four sources is shown as an example in Section D with the system setup, the design of LDPC codes with density evolution and design and simulation results. Section E presents the system setup with the assumption of symmetry in the correlation model, design methodology for m-ary LDPC codes and design and numerical results for two symmetrically correlated ternary sources.

B. Slepian-Wolf Coding Using Wyner's Scheme Based on LDPC Codes

The asymmetric Slepian-Wolf coding problem for two binary sources , A and B (side information), can be treated as a channel coding problem with side information at decoder. Wyner suggested modelling correlations with equivalent channels and applying the syndrome approach for the asymmetric case. For a linear binary (n, k) block channel code, there are 2^{n-k} distinct syndromes. Each syndrome corresponds to a coset leader. Using 2^{n-k} cosets as bins, the encoders transmit the the syndrome, which is the product of the parity check matrix of the binary code and A. At receiver, the decoder find the points in the coset which is is closest to the side information in terms of Hamming distance. The compression ratio is $\frac{n-k}{n}$. If the correlation between two binary sources can be modelled as a binary symmetry channel (BSC) and the binary (n,k) block channel code is capacity approaching, the compression ratio, $\frac{n-k}{n} = 1 - \frac{k}{n} \approx 1 - (1 - H(A|B)) = H(A|B)$. The above argument suggests LDPC codes is perfect for Slepian-Wolf coding problem.

LDPC-based Slepian Wolf Coding was first proposed by Liveris *et al* [31]. In their scheme, it is straightforward to compute syndrome given the parity check matrices of LDPC codes. The output syndrome is the modulo-2 sum of the binary values corresponding to the information nodes connected to the same check node. Upon

receiving the syndrome, decoders treat the side information as the observations from channel in message-passing decoding algorithm and the received syndrome are used to correct the sign when updating the outgoing messages at check nodes. When there are more than two sources available, it is possible to improve the compression efficiency to their joint entropy. Liveris *et al.* [30] consider Slepian-wolf coding of three i.i.d. correlated sources X_1, X_2 and X_3. Given X_1 available at the decoder, from Eq. 42, the achievable rates R_2 and R_3 for X_2 and X_3 have to satisfy the following inequalities: $R_2 \geq H(X_2|X_1, X_3)$, $R_3 \geq H(X_3|X_1, X_2)$ and $R_2 + R_3 \geq H(X_2, X_3|X_1)$. Trying to minimize the rate R_3, i.e. bring it as close as possible to the theoretical limit $H(X_3|X_1, X_2)$, we get that R_2 has to be no less than $H(X_2|X_1)$. Assuming X_1, X_2 and X_3 are equiprobable and $\Pr[X_2 \neq X_1|X_1] = p$ and imposing symmetry condition $\Pr[X_3 = i|X_1 = j, X_2 = k]$ =$\Pr[X_3 \neq i|X_1 \neq j, X_2 \neq k]$ for all $i, j, k \in 0, 1$ and $\Pr[X_3 \neq X_1|X_1]$=$\Pr[X_3 \neq X_2|X_2] = p$, we have lower bound for R_2 and R_3,

$$\begin{aligned} H(X_2|X_1) &= H(p) \\ &= -p \log_2 p - (1-p) \log_2(1-p), \end{aligned} \tag{44}$$

$$\begin{aligned} H(X_3|X_1, X_2) &= p\, H\left(\frac{1}{2}\right) + (1-p)\, H\left(\frac{p/2}{1-p}\right) \\ &= 1 + (1-p) \log_2(1-p) \\ &\quad - \left(1 - \frac{3}{2}p\right) \log_2(2-3p) - \frac{p}{2} \log_2 p. \end{aligned} \tag{45}$$

We have plotted both (44) and (45), i.e. $H(X_2|X_1)$ and $H(X_3|X_1, X_2)$ respectively, as a function of p, in Fig. 21.

The general structure of the encoder and joint decoder is shown in Fig. 22, in which a close-looped decoder is employed with two component decoders C_1 and C_2. To avoid the loop in the decoders and simplify the design problem, we can choose

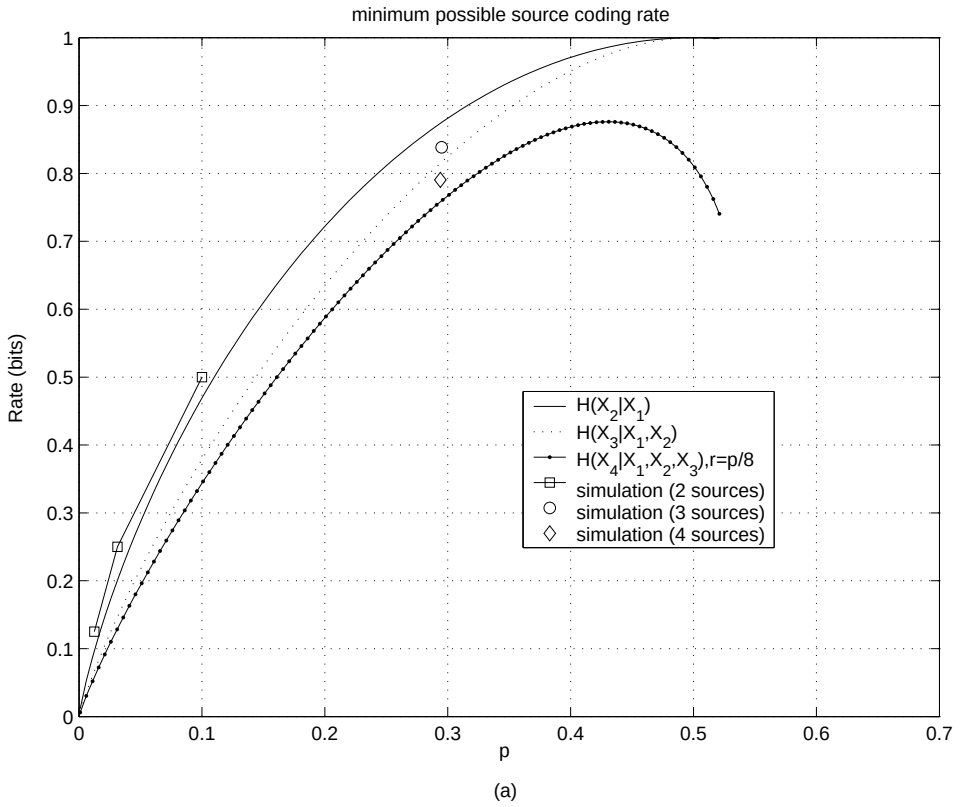

Fig. 21. Minimum achievable source coding rates $H(X_2|X_1) = H(X_3|X_1)$, $H(X_3|X_1, X_2)$ and $H(X_4|X_1, X_2, X_3)$ as functions of p.

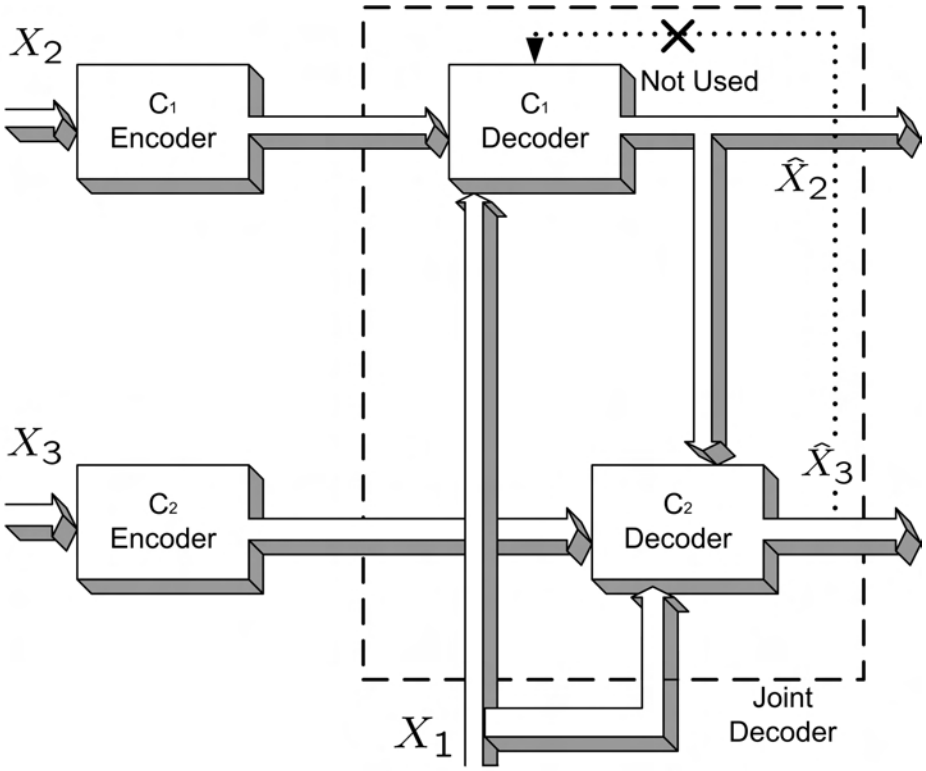

Fig. 22. System for compression of X_1, X_2, X_3.

the rate of C_1 to be close to $H(X_2|X_1)$. Otherwise, C_1 and C_2 have to be designed jointly, making the design process more involved. Theoretically, with this choice of R_2 and R_3, this approach will not cause any noticeable loss in performance if the rate of C_2 is close to $H(X_3|X_1, X_2)$. The encoding and decoding for compression and decompression of (X_1, X_2, X_3) is similar to those used in the two source case. Applying the syndrome concept, the encoders compute the syndromes, S_2 and S_3, for X_2 and X_3 using two LDPC code graphs respectively. The decoding process consists of two steps: first, X_2 is decoded with corresponding decoder using X_1 and S_2. The message-passing algorithm can be used in decoding, taking the syndrome S_2 into account. Then the estimated version of X_2 together with X_1 and S_3 are the input to

the second decoder to estimate X_3. This principle can be easily extended to multiple sources, which is going to be presented in the following section.

C. LDPC Codes for Compression of Multiple Sources

1. Encoding and Decoding

An LDPC code is determined by its graph, which consists of variable (left) and check (right) nodes [15, 17]. An ensemble of LDPC codes is described by the degree distribution pair $\lambda = \{\lambda_1, \lambda_2, \ldots\}$ and $\rho = \{\rho_1, \rho_2, \ldots\}$ [15, 17]. Assuming X_1 is encoded at rate $H(X_1)$ and decoded perfectly with a conventional source code and serves as side information for the component decoders at higher levels, X_2,X_3,...,X_n can be compressed at rates R_2,R_3,...,R_n lower-bounded by $H(X_2|X_1)$,$H(X_3|X_1, X_2)$, ...,$H(X_n|X_1, X_2, ..., X_{n-1})$. The general structure of the encoders and joint decoder is shown in Fig. 23, in which E_l and D_l are the encoder and its corresponding decoder for $l \leq n$, and a multistage decoding algorithm is employed with sub-decoders D_1,...,D_n. Starting from the code at stage 2, if properly designed, the decoder can recover X_2 with an arbitrarily small error probability from side information X_1 and syndrome S_1 when the source coding rate is $H(X_2|X_1)$. For the rest stages of decoding, the code design problems still fit into the scenario above except that the sizes of the sets of all possible side information are larger at higher levels. All of the above argument suggests a code scheme using a multilevel structure and successive decoding with proper rate allocation. Otherwise, if only partial side information is available at a component decoder, code design at each level will intervene with one another, making the design process more involved. Applying the syndrome concept, the encoders compute the syndromes, S_2, S_3,...,S_n for X_2,X_3,...,X_n using n LDPC code graphs respectively. Encoding with LDPC code graphs in the syndrome setup

means computing in the corresponding Galois field the summation of the values of all the variable nodes that are connected to the same check node [31]. The value after summation at each check node is the corresponding syndrome symbol value. Note that the component code can be binary or non-binary, which makes coding two ternary sources a special case.

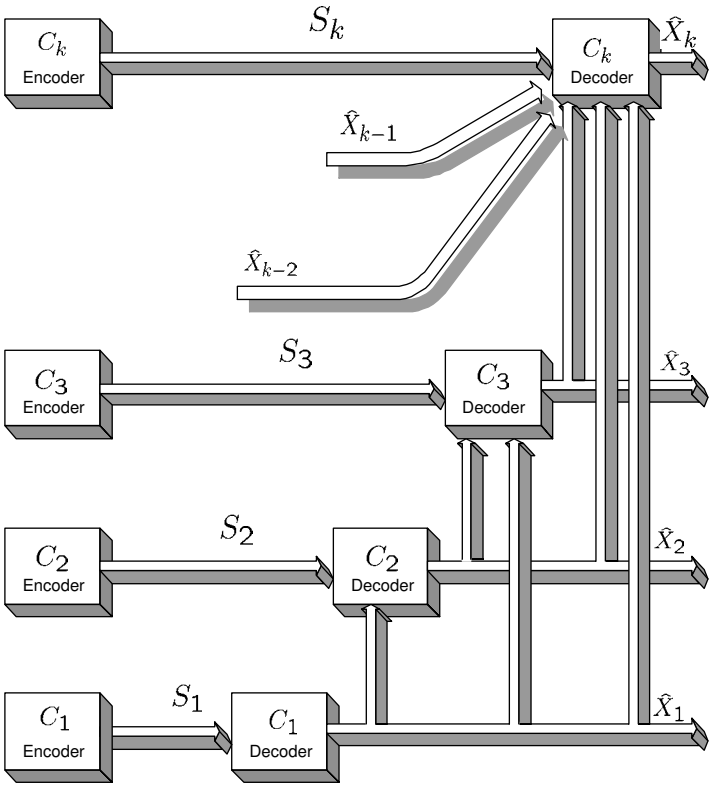

Fig. 23. System for compression of $X_1, X_2, X_3, ..., X_n$.

Since the symmetry condition is satisfied in the source correlation, we can optimize the profiles of the LDPC codes using density evolution [15, 17, 16]. What we need now are the initial pdf's of the log-likelihood ratios (LLRs) for the left node bits, which depend on the correlation between the input sources and side information.

Assuming that a Slepian-Wolf limit approaching code including (E_1, D_1),(E_2, D_2),

...., (E_n, D_n) is used to reach their theoretical limits, we do not expect significant improvement by using the true X_1, X_2, ..., X_{l-1} instead of the estimated ones in decoding of X_i, $i = 1...l$, typical for symbol error rate less than 10^{-5}, where X_1, X_2,..., X_{l-1} are grouped as a symbol. However, the error rates are expected to be higher than those at lower levels due to error propagation in side information, which will be observed from simulations.

D. Slepian-Wolf Coding of Four Binary Sources

1. System Model

Assume we have three binary sources with equiprobable outputs X_1, X_2 and X_3 with $\Pr[X_l \neq X_t | X_t] = p$ for $l \neq t$. Imposing the symmetry condition, i.e., $\Pr[X_3 = i | X_1 = j, X_2 = k]$ =$\Pr[X_3 \neq i | X_1 \neq j, X_2 \neq k]$ for all $i, j, k \in \{0, 1\}$, produces

$$\Pr[X_1 = X_2 = X_3 | X_3] = 1 - \frac{3}{2}p, \tag{46}$$

$$\Pr[X_1 \neq X_2 = X_3 | X_3] = \frac{p}{2}, \tag{47}$$

$$\Pr[X_2 \neq X_1 = X_3 | X_3] = \frac{p}{2}, \tag{48}$$

$$\Pr[X_1 \neq X_2 \neq X_3 | X_3] = \frac{p}{2}, \tag{49}$$

where Eq. (46)-(49) result from just imposing the symmetry condition on all these binary conditional probabilities. Having a fourth binary source X_4 with equiprobable output and $\Pr[X_4 \neq X_l | X_l] = p$, $l \in 1, 2, 3$, we get an extra degree of freedom

$r \in [0, \frac{p}{2}]$. The conditional probabilities $\Pr[X_1, X_2, X_3|X_4]$ in this case are

$$\begin{aligned}
\Pr[X_1 = 0 = X_2 = X_3|X_4 = 0] &= 1 - 2p + r, && (50)\\
\Pr[X_1 = 0 = X_2 \neq X_3|X_4 = 0] &= \frac{p}{2} - r, && (51)\\
\Pr[X_1 = 0 = X_3 \neq X_2|X_4 = 0] &= \frac{p}{2} - r, && (52)\\
\Pr[X_2 = 0 = X_3 \neq X_1|X_4 = 0] &= \frac{p}{2} - r, && (53)\\
\Pr[X_1 = 1 = X_2 = X_3|X_4 = 0] &= \frac{p}{2} - r, && (54)\\
\Pr[X_1 = 1 = X_2 \neq X_3|X_4 = 0] &= r, && (55)\\
\Pr[X_1 = 1 = X_3 \neq X_2|X_4 = 0] &= r, && (56)\\
\Pr[X_2 = 1 = X_3 \neq X_1|X_4 = 0] &= r, && (57)
\end{aligned}$$

and similarly when conditioned on $X_4 = 1$. This correlations between (X_1, X_2, X_3) and X_4 can be modelled as a discrete memoryless channel as shown in Fig. 24. Another view of this problem is to form four binary sources to 2 4-ary sources, i.e., to group (X_1, X_2) as Y_1 and (X_3, X_4) as Y_2. Then the equivalent correlation for two 4-ary sources model is given as:

$$\begin{aligned}
\Pr[Y_1 = 00|Y_2 = 00] &= 1 - 3q, && (58)\\
\Pr[Y_1 = 01|Y_2 = 00] &= q, && (59)\\
\Pr[Y_1 = 10|Y_2 = 00] &= q, && (60)\\
\Pr[Y_1 = 11|Y_2 = 00] &= q, && (61)\\
\Pr[Y_1 = 00|Y_2 = 01] &= 1/4, && (62)\\
\Pr[Y_1 = 01|Y_2 = 01] &= 1/4, && (63)\\
\Pr[Y_1 = 10|Y_2 = 01] &= 1/4, && (64)\\
\Pr[Y_1 = 11|Y_2 = 01] &= 1/4, && (65)
\end{aligned}$$

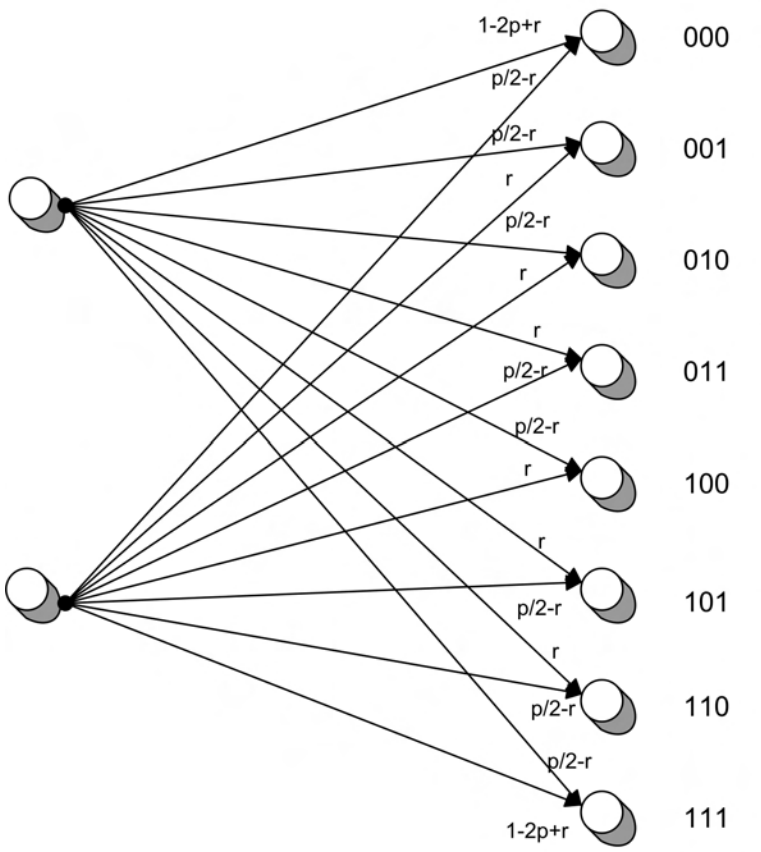

Fig. 24. A discrete memoryless channel modelling the correlation between X_4 and (X_1, X_2, X_3).

where $q = \frac{\frac{p}{4}}{1-p}$. Using Multilevel coding, the one with the four binary sources and the one with the two 4-ary sources, the same Slepian-Wolf coding system can be used to approach the theoretical limits. The reason is given as follows. In the case of the two 4-ary sources the theoretical limit for loseless compression of the second sources Y_2 assuming Y_1 is available at the decoder, is given by

$$R_{Y_2} \geq H(Y_2|Y_1) = H(X_3, X_4|X_1, X_2) = H(X_3|X_1, X_2) + H(X_4|X_1, X_2, X_3). \quad (66)$$

Therefore, the multilevel codes designed for coding of four binary sources can be used for coding two 4-ary sources, in which the sub-codes designed for compression X_3 and X_4 given (X_1, X_2) and $(X_1, X_2, \hat{X_3})$ as side information can be considered as a joint code for coding of $Y_2(X_3$ and $X_4)$ given $Y_1(X_1$ and $X_2)$. Therefore, we shall focus on coding of four binary sources in example as follows.

2. Theoretical Limits

In this case of coding four binary sources, a multilevel code of four levels can achieve the Slepian-Wolf limit where the first level uses a conventional source coder. We assume that X_1 is compressed at rate $H(X_1)$ and perfectly recovered at the output of the corresponding decoder. From Eq. (42), the sources coding rates, R_2, R_3 and

R_4, for the compression of X_2, X_3 and X_4 have to satisfy the following

$$R_2 \geq H(X_2|X_1, X_3, X_4), \tag{67}$$

$$R_3 \geq H(X_2|X_1, X_2, X_4), \tag{68}$$

$$R_4 \geq H(X_4|X_1, X_2, X_3), \tag{69}$$

$$R_2 + R_3 \geq H(X_2, X_3|X_1, X_4), \tag{70}$$

$$R_2 + R_4 \geq H(X_2, X_4|X_1, X_3), \tag{71}$$

$$R_3 + R_4 \geq H(X_3, X_4|X_1, X_2), \tag{72}$$

$$R_2 + R_3 + R_4 \geq H(X_2, X_3, X_4|X_1). \tag{73}$$

Trying to operate at a corner point of the achievable region, where $R_2+R_3+R_4$ is minimized, we set R_2, R_3 and R_4 to be $H(X_2|X_1)$, $H(X_3|X_1, X_2)$ and $H(X_4|X_1, X_2, X_3)$, respectively. Note that this setup allows us to design component codes independently and to use a multi-stage decoding algorithm.

Explicitly expressing the theoretical limit in terms of p and r, we get $H(X_2|X_1)$, $H(X_3|X_1, X_2)$ as given by Eq. (44) and Eq. (45), and

$$H(X_4|X_1, X_2, X_3) = \frac{3}{2}H(1 - 2r/p) + (1 - \frac{3p}{2})H(\frac{\frac{p}{2} - r}{1 - \frac{3p}{2}}), \tag{74}$$

where $H(p) = -p\log_2 p - (1-p)\log_2(1-p)$. Eq. (44) and (45) are the same as in [30]. We have plotted Eq. (44), Eq. (45) and Eq. (74), i.e., $H(X_2|X_1)$, $H(X_3|X_1, X_2)$ and $H(X_4|X_1, X_2, X_3)$ respectively, as a function of p and $r = \frac{p}{8}$, in Fig. 21. Using syndromes [85] or equivalently Wyner's scheme [44, 98], we can translate the minimum achievable source coding rates into channel coding rates that refer to the channel codes that are used for source coding with side information at the decoder. These maximum channel coding rates are shown in Fig. 25.

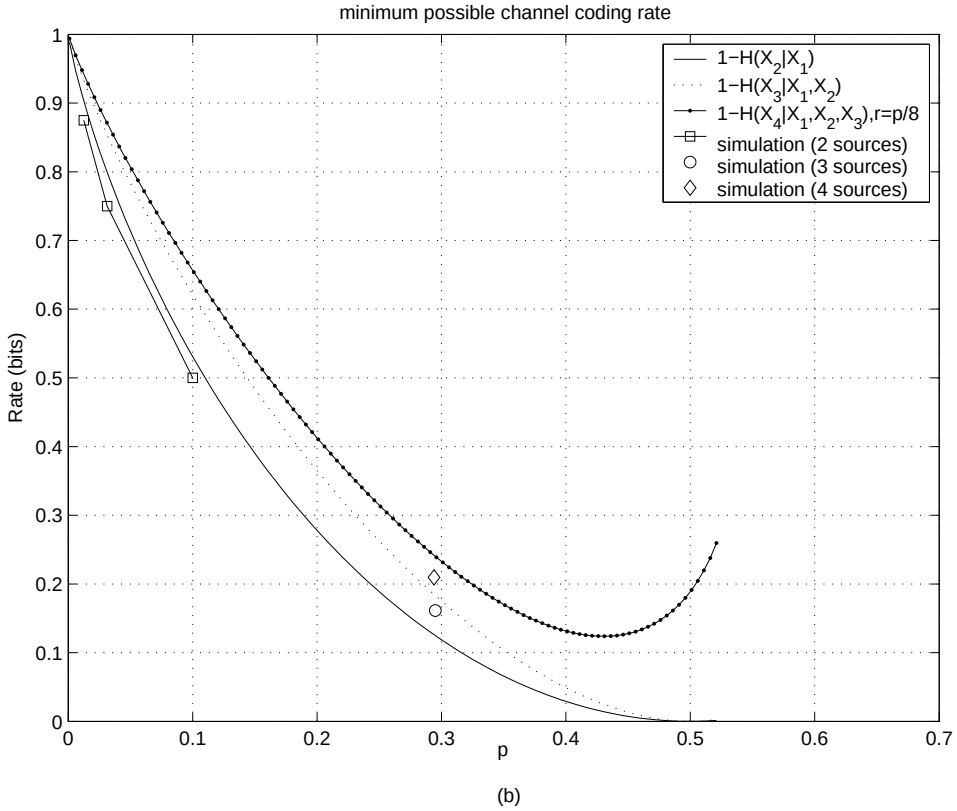

Fig. 25. The corresponding maximum channel code rates $1 - H(X_2|X_1)$, $1 - H(X_3|X_1, X_2)$ and $1 - H(X_4|X_1, X_2, X_3)$ as functions of p, when using syndromes [85] (or Wyner's scheme [44, 98]).

3. Binary LDPC Code Design with Density Evolution

Here we explain how to design LDPC codes for Slepian-Wolf coding of four sources. Taking C_4 as an example, from (50)-(57), we can see there are four possible input LLRs to the decoder, $\pm\log\frac{1-2p+r}{\frac{p}{2}-r}$ and $\pm\log\frac{\frac{p}{2}-r}{r}$, respectively. Therefore, the initial pdf of LLRs for X_4 given X_1, X_2 and X_3 is

$$\begin{aligned} f_{X_4}(t) &= \frac{3p}{8}\delta(t+\log\frac{\frac{p}{2}-r}{r})+\frac{9p}{8}\delta(t-\log\frac{\frac{p}{2}-r}{r}) \\ &\quad +\frac{3p}{8}\delta(t+\log\frac{1-2p+r}{\frac{p}{2}-r})+(1-\frac{15p}{8})\delta(t-\log\frac{1-2p+r}{\frac{p}{2}-r}). \end{aligned} \tag{75}$$

With f_{X_4} as the initial pdf in density evolution, the objective is to find the source code of minimum rate while assuming that the resulting bit error probability P_e is below 10^{-6}. For a given correlation model, i.e., in our setup for fixed p and r, the problem can be formulated as

$$(\lambda,\rho)=\arg\ \min_{S}\frac{\sum_i\frac{\rho_i}{i}}{\sum_i\frac{\lambda_i}{i}}, \tag{76}$$

where $S=\{(\lambda,\rho)|P_e(\lambda,\rho)<10^{-6},\sum_i\lambda_i=1,\ \sum_i\rho_i=1,\lambda_i\geq 0,\rho_i\geq 0\}$.

4. Design and Simulation Results

In Fig. 21 we also show the simulation results from [31] and [30], which correspond to the highest simulated compression at these rates given in the literature for distributed compression of two and three equiprobable binary sources. The same approach as in [31] and [30] can be used to design the component codes at all levels to get rates close to the theoretical limits. Therefore, we focus on showing how this approach can be extended to get R_l close to $H(X_l|X_1,X_2,...,X_{l-1})$ and obtain the rate savings predicted by the theoretical limit of Fig. 21. Take $l=4$ as an example. In addition

to the ideal case, errors in the outputs from lower levels are also taken into account in simulations.

Consider designing C_4. Using the discretized density evolution [17], the resulting optimized profiles of C_4 code for $p = 0.3$ and $r = \frac{p}{8}$ are

$$\begin{aligned}
\lambda(x) &= 0.2979x + 0.1764x^2 + 0.1157x^3 + 0.0793x^4 + 0.0550x^5 + 0.0376x^6 \\
&+ 0.0246x^7 + 0.0145x^8 + 0.0064x^9 + 0.0280x^{14} + 0.0352x^{15} + 0.0150x^{16} \\
&+ 0.0153x^{19} + 0.0131x^{22} + 0.0189x^{23} + 0..0296x^{24} + 0.0093x^{25} + 0.0116x^{28} \\
&+ 0.0059x^{30} + 0.0108x^{35}. \\
\rho(x) &= 0.6738x^3 + 0.3262x^5. \qquad (77)
\end{aligned}$$

The rate of the ensemble of LDPC codes with this profile is 0.2095, which corresponds to a compression rate of $1 - 0.2095 = 0.7905$. By Eq. (74) and (45), $H(X_4|X_1, X_2, X_3) = 0.8247$ and $H(X_4|X_1, X_2) = 0.7671$ for $p = 0.3$. Our result is only 0.0264 bits away from $H(X_4|X_1, X_2, X_3)$ and 0.0342 bits below $H(X_4|X_1, X_2)$. This is shown in the second row of Table VII. In the third row of the table, the correlation in terms of p is shown for fixed rate $R = 0.7905$, equal to the code rate of Eq. (77). So $p = 0.2793$ results in $H(X_4|X_1, X_2) = 0.7905$ from Eq. (44) and $p = 0.3168$ results in $H(X_4|X_1, X_2, X_3) = 0.7905$ from Eq. (45).

Table VII. Comparison of the LDPC code threshold with the theoretical limit in terms of both rate and correlation.

	$H(X_4\|X_1, X_2)$	LDPC (77)	$H(X_4\|X_1, X_2, X_3)$
$p = 0.3$	R=0.8247	R=0.7905	R=0.7671
$R = 0.7905$	$p = 0.2793$	$p = 0.3000$	$p = 0.3168$

The simulated performance of C_3 [30] and this LDPC code with codeword length $5 \cdot 10^5$ has been plotted in Fig. 26. Each simulation point in the figure corresponds to 30 frame errors after 200 decoding iterations. Convergence of the LDPC code is achieved for about $p = 0.294$ for which $H(X_4|X_1, X_2) = 0.8167$ and $H(X_4|X_1, X_2, X_3) = 0.7597$, i.e., this LDPC code can come about 0.0308 bits away from the theoretical limit and exceed the three sources theoretical limit by about 0.0262 bits. Considering the correlation model changes due to errors in side information, the new correlation model is

$$
\begin{aligned}
\Pr[X_1 = 0 = X_2 = X_3 | X_4 = 0] &= (1 - \frac{15p}{8})(1 - \epsilon_1)(1 - \epsilon_2) + \frac{p}{8}\epsilon_1\epsilon_2 + \frac{3p}{8}\epsilon_1 + \frac{3p}{8}\epsilon_1 \\
\Pr[X_1 = 0 = X_2 \neq X_3 | X_4 = 0] &= \frac{3p}{8}(1 - \epsilon_1)(1 - \epsilon_2) + \frac{p}{8}\epsilon_1 + (1 - \frac{15p}{8})\epsilon_2 + \frac{3p}{8}\epsilon_1\epsilon_2 \\
\Pr[X_1 = 0 = X_3 \neq X_2 | X_4 = 0] &= \frac{3p}{8}(1 - \epsilon_1)(1 - \epsilon_2) + \frac{p}{8}\epsilon_2 + (1 - \frac{15p}{8})\epsilon_1 + \frac{3p}{8}\epsilon_1\epsilon_2 \\
\Pr[X_2 = 0 = X_3 \neq X_1 | X_4 = 0] &= \frac{3p}{8}(1 - \epsilon_1)(1 - \epsilon_2) + \frac{p}{8}\epsilon_1 + \frac{p}{8}\epsilon_2 + \frac{3p}{8}\epsilon_1\epsilon_2 \\
\Pr[X_1 = 1 = X_2 = X_3 | X_4 = 0] &= \frac{3p}{8}(1 - \epsilon_1)(1 - \epsilon_2) + \frac{p}{8}\epsilon_1 + \frac{3p}{8}\epsilon_2 + \frac{p}{8}\epsilon_1\epsilon_2 \\
\Pr[X_1 = 1 = X_2 \neq X_3 | X_4 = 0] &= \frac{p}{8}(1 - \epsilon_1)(1 - \epsilon_2) + \frac{3p}{8}\epsilon_1 + \frac{3p}{8}\epsilon_2 + \frac{p}{8}\epsilon_1\epsilon_2 \\
\Pr[X_1 = 1 = X_3 \neq X_2 | X_4 = 0] &= \frac{p}{8}(1 - \epsilon_1)(1 - \epsilon_2) + \frac{3p}{8}\epsilon_1 + \frac{3p}{8}\epsilon_2 + \frac{p}{8}\epsilon_1\epsilon_2 \\
\Pr[X_2 = 1 = X_3 \neq X_1 | X_4 = 0] &= \frac{p}{8}(1 - \epsilon_1)(1 - \epsilon_2) + \frac{3p}{8}\epsilon_1 + \frac{3p}{8}\epsilon_2 + (1 - \frac{15p}{8})\epsilon_1\epsilon_2
\end{aligned}
$$

where $r = \frac{p}{8}$, ϵ_1 and ϵ_2 are error rates in side information caused by decoders at the previous two stages and we assume no errors in X_1.

Similar correlation models with errors from decoders at lower levels can be derived easily in the case of three sources. Both the design and the simulation results are also shown in Fig. 21. In Fig. 26, the simulated performances of LDPC codes with perfect and imperfect side information are shown with p for three and four sources. Each simulation point in Fig. 26 corresponds to 30 frame errors after 200

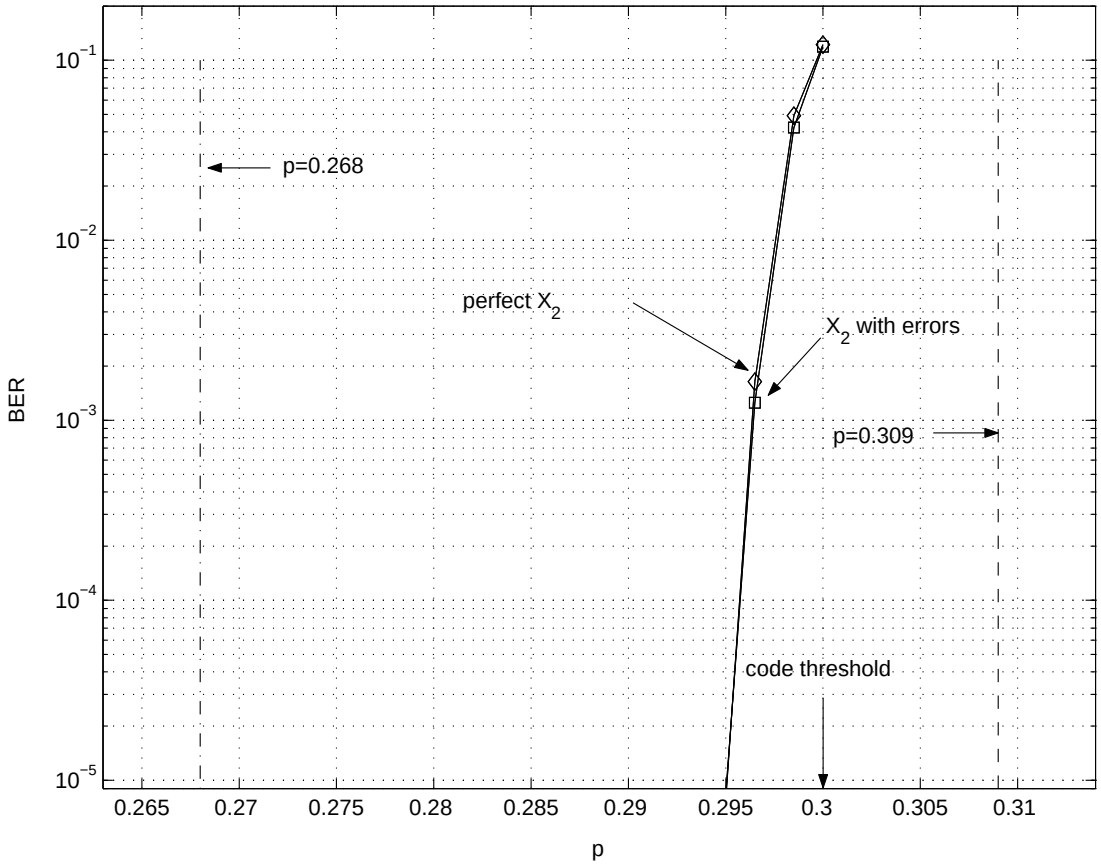

Fig. 26. Simulation results with perfect and imperfect side information for the LDPC code (77) as a function of p. The $p = 0.268$ for $H(X_2|X_1) = H(X_3|X_1) = 0.8386$, the $p = 0.309$ for $H(X_3|X_1, X_2) = 0.8386$ and the LDPC code threshold $p = 0.300$ are also shown.

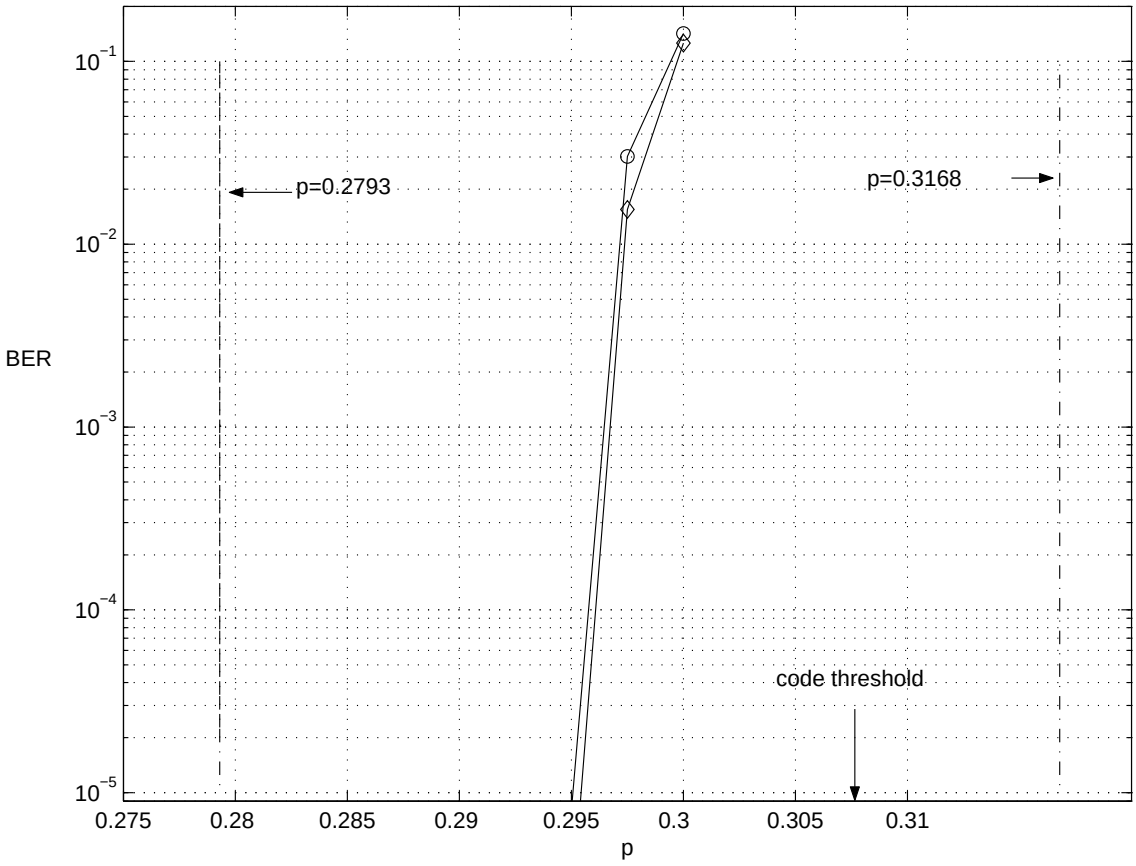

Fig. 27. Simulation with perfect and imperfect side information results for the LDPC code Eq. (77) as a function of p. The $p = 0.2793$ for $H(X_4|X_1, X_2) = H(X_3|X_1, X_2) = 0.7905$, the $p = 0.3168$ for $H(X_3|X_1, X_2) = 0.7905$ and the LDPC code threshold $p = 0.300$ are also shown.

decoding iterations. It is shown in Fig. 27 that the source coding system suffers from performance loss due to imperfect side information in the case of coding four sources.

E. Coding of Two Ternary Sources

1. System Model

We consider Slepian-Wolf coding of two ternary sources. As the correlation models in the binary case, symmetry is assumed. Let X_1 and X_2 denote two ternary sources, the conditional probability matrix $\mathbf{P}$ is given as

$$\begin{bmatrix} 1-2q & q & q \\ q & 1-2q & q \\ q & q & 1-2q \end{bmatrix} \tag{78}$$

where $P_{i,j} = p(X_2 = j | X_1 = i)$ for $i, j = 1, 2, 3$.

2. Ternary LDPC Codes

When the sources are not binary, one can use m-ary LDPC codes as component codes in a multilevel code and compute the syndrome. An m-ary LDPC code is defined by its parity matrix whose entries are taken from a finite Galois field, GF(m). As binary LDPC codes, m-ary LDPC codes can be decoded with message-passing algorithms. Before the algorithm is given, a few of notations are defined. $V(c)$ denotes the set of variable nodes that participate in the check node c. Similarly, $C(v)$ denotes the set of variable nodes that depend on the variable node v. q_{cv}^a denotes the probability that variable node v is a given the information obtained via check other than c. r_{cv}^a denotes probability that check c is satisfied if the variable node v is fixed at a. Then the update rule at variable nodes of the decoding algorithm for m-ary LDPC codes

[97] is given by

$$q_{cv}^{a} = \alpha_{cv} f_{v}^{a} \prod_{j \in V(C) \backslash v} r_{jv}^{a} \tag{79}$$

and the update rule at check nodes is given by

$$r_{cv}^{a} = \sum_{X' : X'_v = a} \Pr[Z_c | X'] \prod_{j \in C(V) \backslash c} q_{cj}^{x'_j}. \tag{80}$$

3. Ternary LDPC Code Design with EXIT Charts

The density evolution used to design binary LDPC codes can still be applied to non-binary cases. However, since the input LLRs are random vectors, designing a m-ary LDPC code with density evolution is very time-consuming, which simply prohibits us to do so. A rough estimation of the complexity of density evolution based on a look-up table [16] gives us an idea how complex the design is. The look-up table, $R(a, b)$, is for the tanh operation. Given the outgoing message $c = R(a, b)$ and two incoming messages a and b, the probability mass function p_c of c is given by $p_c[k] = \sum_{(i,j):k\Delta = R(i\Delta, j\Delta)} p_a[i] p_b[j]$. Considering the case of ternary codes, a, b and c in the above expression are random vectors of dimension 3. If a summation over n terms is required for a $p_c[k]$ in the binary case, a summation over $n_1 n_2 n_3$ terms is going to be needed in the ternary case, where n_i is the number of terms in summation for the ith entry of a given vector c. Therefore, another approach based on EXIT charts is used to design m-ary LDPC codes in this chapter. Exit charts was first proposed by [96] to analyze the convergent behavior of decoders for turbo codes. The idea is to track the extrinsic information for each iteration in the decoding process. It turns out that this technique can be applied to LDPC codes as well. Define $\Pr_A(i)$ as the input probability of i^{th} symbol at variable nodes and similarly $\Pr_E(i)$ for check nodes. Let $f(I_A)$ and $g(I_E)$ be the extrinsic information transfer functions for variable and

nodes, respectively, where

$$I_E = \log_2 m + \sum_{i=0}^{m-1} \Pr_E(i) \log_2 \Pr_E(i), \tag{81}$$

and

$$I_A = \log_2 m + \sum_{i=0}^{m-1} \Pr_A(i) \log_2 \Pr_A(i). \tag{82}$$

A property of capacity-approaching codes is that the reliability of extrinsic information increases with iterations for their rates smaller than capacity, which implies $f(I_A)$ should be greater than $g^{-1}(I_E)$. For an irregular LDPC codes, the transfer functions of the extrinsic information for variable and check nodes of different degree should be available before we can proceed to design the degree-profiles. Unfortunately, there is a lack of explicit expressions for the transfer functions of the extrinsic information for general channels, except for Gaussian channels. Therefore, we have to resort to simulation for the channels (correlation models) considered in this chapter. Once the extrinsic information transfer curves is obtained, we are able to setup an optimization problem to solve for λ_i and ρ_i as follows:

$$\min \quad \frac{\sum_i \frac{\rho_i}{i}}{\sum_i \frac{\lambda_i}{i}} \tag{83}$$

subject to $\sum_i \lambda_i f_i(I_A) \geq \sum_i \rho_i g_i^{-1}(I_A)$, $1 \geq \lambda_i \geq 0$ and $1 \geq \rho_i \geq 0$. where $f_i(I_A)$ and $g_i(I_E)$ are the extrinsic information transfer functions for variable and check nodes of degree i and j respectively.

4. Design and Simulation Results

To get $f_i(I_A)$ and $g_i(I_E)$, simulations are conducted to collect different points on the transfer curves while fixing λ_i or ρ_i at 1. Curve-fitting tools can be used to

approximate $f_i(I_A)$ and $g_i(I_E)$ with polynomial or exponential curves (to reduced approximation errors). Then we can evaluate the approximated version of $f_i(I_A)$ and $g_i(I_E)$ at desired values of I_A and I_E. Therefore, Eq. 83 can be set-up with a set of sampled linear constraints and solved with standard optimization tools. For coding two ternary sources, $q = 0.1$ as assumed in the correlation model, using EXIT charts the optimized degree profiles are obtained with (83)

$$\begin{aligned} \lambda(x) &= 0.2281x^1 + 0.1231x^2 + 0.1149x^3 + 0.1001x^5 + 0.0177x^6 + 0.4162x^{23}. \\ \rho(x) &= 0..0015x^3 + 0.5592x^6 + 0.4393x^7. \end{aligned} \tag{84}$$

The rate of this codes is 0.3846 and the capacity for the correlation model is 0.4183, which is only 0.0337 bits away from the theoretical limit. The extrinsic information transfer curves for variable and check nodes of different degrees are shown in Fig. 28. The extrinsic information transfer curves averaged over λ_i and ρ_i are shown Fig. 29. The simulated performance of the length 5×10^5 ternary code is 4.43×10^{-4} (symbol error rate) at $q = 0.085$, for which the capacity is 0.4778. The loss is about 0.0932 in channel coding rate (or $0.0932 \log_2 3 = 0.1477$ bit in compression performance). The larger loss in this ternary case agrees with what has been observed for the AWGN channel, i.e., it is harder to achieve the capacity with non-binary signaling/coding.

F. Summary

We have shown how LDPC codes can be used for distributed compression of multiple m-ary correlated sources. Using the equivalent channels, the Slepian-Wolf coding problems can be treated as channel coding problems. LDPC codes were designed for such a setup with threshold 0.0264 and 0.0337 bits away from the theoretical limit

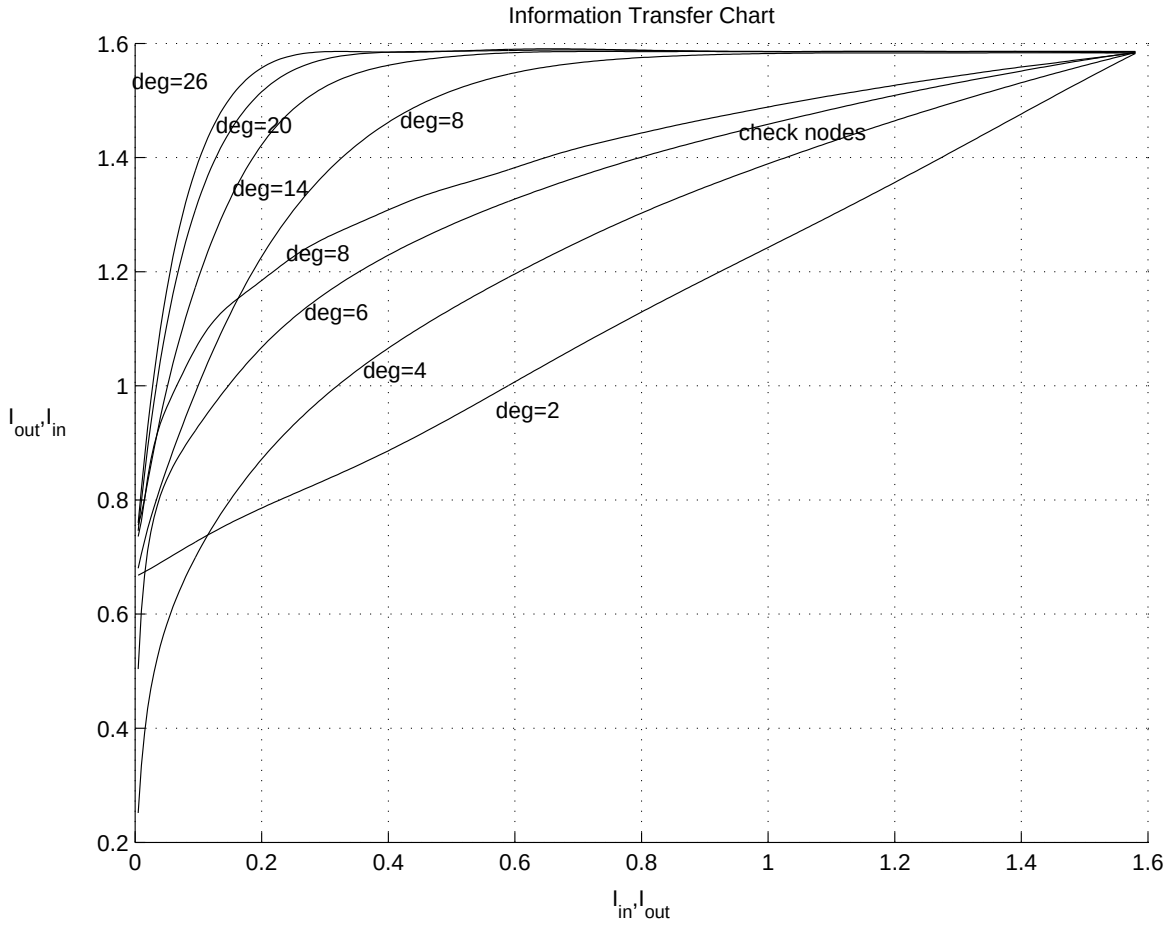

Fig. 28. Extrinsic information transfer curves for variable and check nodes of different degrees.

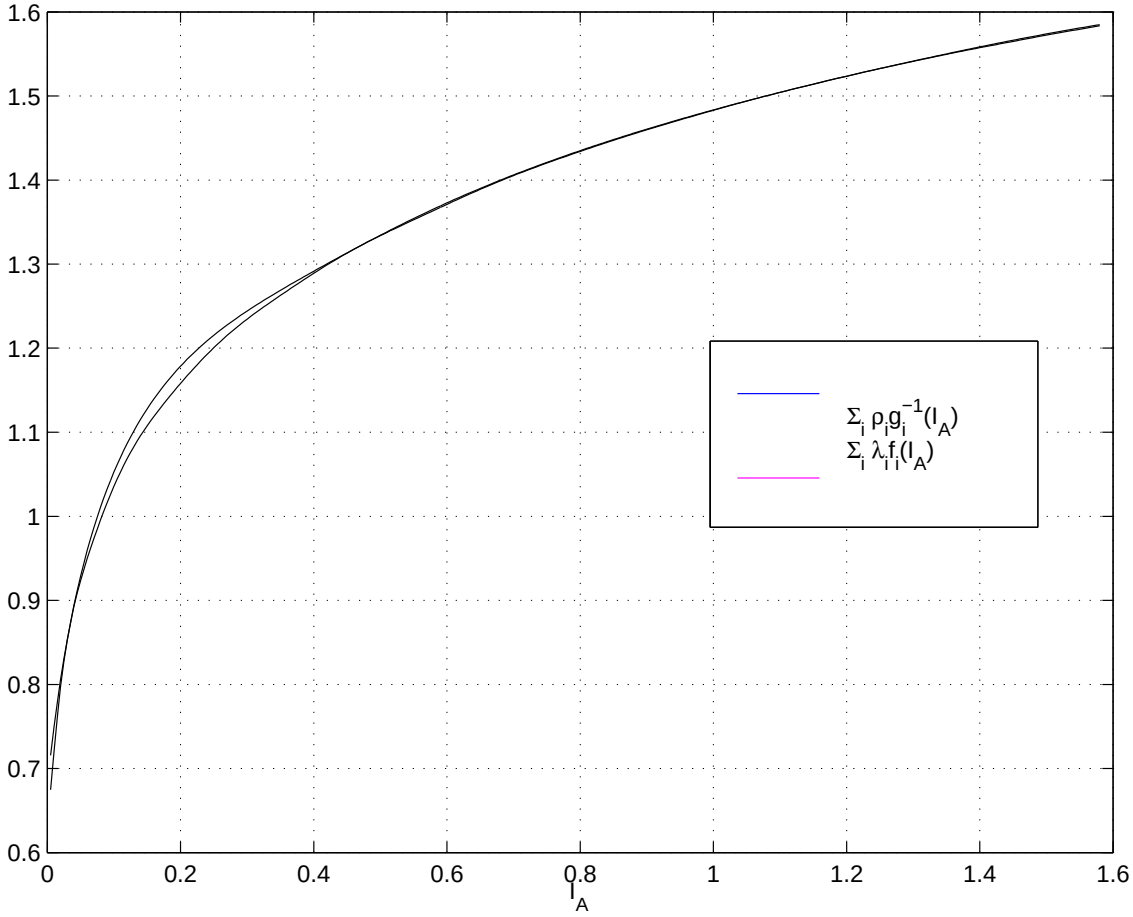

Fig. 29. $\sum_i \lambda_i f_i(I_A)$ and $\sum_i \rho_i g_i^{-1}(I_E)$, averaged extrinsic information transfer functions for variable and check nodes.

in the case coding of four binary and two ternary sources. Based on the chain rule for the joint entropy, the same approach can be extended to multiple sources. For non-binary sources, m-ary LDPC codes can also approach the theoretical limit.

CHAPTER VI

CONCLUSION

This dissertation is dedicated to the design of a class of graphed-based error-correcting codes, which can be represented in Tanner graphs and can be decoded using soft iterative message-passing algorithms, and their applications to scalable image and video transmission, lossless distributed data compression.

The first part of the dissertation is to design and analysis of rate-compatible IRA codes for scalable image transmission systems. In a scalable image transmission system, rate-compatible channel codes are required to provide unequal error protection for bits of different importance. This part include the analysis with density evolution the effect of puncturing applied to IRA codes and asymptotic analysis of the performance of the systems. We have presented a JSCC system for scalable image and video transmission based on state-of-the-art scalable source coders and rate-compatible IRA codes. The set of IRA codes obtained by puncturing achieves rate-compatibility while maintaining a high performance and lower decoding complexity. By concatenating a rate-optimal solution with a modified local search algorithm, the error control scheme achieves compatible performance to the Viterbi algorithm used in [28], while the complexity is lower. We have applied this joint design to the transmission of scalable image and video transmission over BSC channels, using state-of-the-art source coders. Simulations show uniformly better results over those reported in previous works.

In the second part of the dissertation, we consider designing source-optimized IRA codes. The idea is to take advantage of the capability of unequal error protection (UEP) of IRA codes against errors because of their irregularities. In video and image transmission systems, the ene-to-end performance is measured by peak signal to noise

ratio (PSNR). We propose an approach to design IRA codes optimized for such a criterion. Based on this approach, we have presented a method of designing IRA codes for sources characterized by cost functions. By assuming a cost function and independence and incorporating this model into channel code design, we obtain IRA codes that is designed for certain source which is characterized by cost function. The procedure to optimize IRA codes automatically decides the best UEP scheme (degree profiles) when the cost function is distortion-rate function. We justify this idea by designing IRA codes for SPIHT-coded image sources. Simulations show that our scheme outperforms conventional UEP scheme and our results are $0.40 \sim 0.73$ dB away from the theoretical limit. Our design achieves source-optimized channel coding or true joint source-channel coding. This is different from usual combined source-channel coding designs.

The third part of the dissertation is to investigate the Slepian-Wolf coding problem using LDPC codes. The problems which have been addressed include coding problem involving multiple sources and non-binary sources, and coding using multi-level codes and non-binary codes. We have shown how LDPC codes can be used for distributed compression of multiple m-ary correlated sources. Using the equivalent channels, the Slepian-Wolf coding problems can be treated as channel coding problems. LDPC codes were designed for such a setup with threshold 0.0264 and 0.0337 bits away from the theoretical limit in the case coding of four binary and two ternary sources. Based on the chain rule for the joint entropy, the same approach can be extended to multiple sources. For non-binary sources, m-ary LDPC codes can also approach the theoretical limit.

Through study, the understanding of coding theory and the practice of designing good codes are achieved by employing and designing codes in real applications. The approach (density evolution) is found useful in a variety of applications and yields

fruitful results.

REFERENCES

[1] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J*, vol. 27, pp. 379 - 423, 1948.

[2] T. M. Cover and J. A. Thomas, *Elements of Information Thoery*, New York: John Wiley & Sons, 1991.

[3] D. Slepian and J.K. Wolf, "Noiseless coding of correlated information sources", *IEEE Trans. Inform. Theory*, vol. IT-19, pp. 471 - 480, Mar. 1973.

[4] S. B. Wicker, *Error Control Systems for Digital Communication and Storage* Englewood Cliffs, NJ: Prentice Hall, 1995.

[5] G. D. Forney, Jr., *Concatenated Codes*, Cambridge, MA: MIT Press, 1966.

[6] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: Turbo-codes," *IEEE Trans. Commun.*, vol. 44, pp. 1261 - 1271, Oct. 1996.

[7] P. D. Alexander, A. J. Grant and M. C. Reed, "Iterative detection in code-division multiple access with error control coding," *European Trans. Telecommun.*, vol. 9, no. 5, pp. 419 - 429, Sep./Oct. 1998.

[8] C. Douillard, M. Je'ze'quel and C. Berrou, "Iterative correction of intersymbol interference: Turbo-Equalization," *European Trans. Telecommun.*, vol. 6, no. 5, pp. 507 - 511, Sep./Oct. 1995.

[9] J. R. Barry, A. Kavcic, S. W. McLaughlin, A. Nayak and W. Zeng, "Iterative timing recovery" *IEEE Signal Processing Magazine*, vol.. 21 , pp. 89 - 102, Jan. 2004.

[10] S. Benedetto, G. Montorsi, D. Divsalar and F. Pollara, "Serial concatenated of interleaved codes: Performance analysis, design and iterative decoding," *JPL TDA Progress Report*, vol. 42 - 126, Aug. 1996.

[11] D. Divsalar and F. Pollara, "Hybrid and serial concatenated codes with applications, "In *Proc. Intl. Symp. on Turbo Codes and Related Topics*, Brest, France, pp. 80 - 87, Aug. 1997.

[12] R. G. Gallager, *Low-Density Parity-Check Code*, Cambridge, MA: MIT Press, 1963.

[13] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Trans. Inform. Theory*, vol. IT-45, pp. 399 - 431, Mar. 1999.

[14] T. Richardson and R. Urbanke, "The capacity of Low-Density Parity-Check codes under message-passing decoding," *IEEE Trans. Inform. Theory*, vol. IT-47, pp. 599 - 618, Feb. 2001.

[15] T. Richardson, A. Shokrollahi, and R. Urbanke, "Design of capacity approaching irregular Low-Density Parity-Check codes," *IEEE Trans. Inform. Theory*, vol. IT-47, pp. 619 - 637, Feb. 2001.

[16] S. Y. Chung, G. D. Forney, Jr., T. J. Richardson and R. Urbanke, "On the design of Low-Density Parity-Check codes within 0.0045dB of the Shannon limits," *IEEE Commun. Letter*, vol. 5, No. 2, pp. 58 - 60, Feb. 2001.

[17] S.-Y. Chung, On the construction of some capacity-approaching coding schemes, Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, MA, 2000.

[18] D. Divsalar, H. Jin and R. J. McEliece,“ Coding theorems for “turbo-like codes” codes,”” in *Proc. Allerton Conf. Commun. and Control*, Monticello, IL, pp. 201 - 210, Sep. 1998.

[19] H. Jin, A. Khandekar and R. McEliece, “Irregular repeat accumulate codes,” in *Proc. Intl. Symp. on Turbo Codes and Related Topics*, Brest, France, pp. 1 - 8, Sep. 2000.

[20] J. Li, K. R. Narayanan and C. N. Georghiades,“ A class of linear-complexity, soft-decodable, high-rate, ‘goog codes’: Construction, properties and performance,” in *Proc. Intl. Symp. Information Theory*, Washington, D.C., pp. 122, Jun. 2001.

[21] J. Li, K. R. Narayanan and C. N. Georghiades, “Product accumulate codes: A class of capacity-approaching, low-complexity codes, ” *IEEE Trans. Inform. Theory*, vol. IT-50, pp. 31 - 46, Jan. 2004.

[22] A. Said and W. A. Pearlman, “A new, fast, and efficient image code based on set partitioning in hierarchical trees,” *IEEE Trans. Circuits and Systems for Video Technology*, vol. 6, pp. 243 - 250, Jun. 1996.

[23] D. Taubman and M. Marcellin, *JPEG2000: Image Compression Fundamentals, Standards, and Practice*, Boston: Kluwer Academic Publishers, 2001.

[24] B. Kim, Z. Xiong, and W. Pearlman, “Very low bit-rate embedded video coding with 3-D set partitioning in hierarchical trees (3-D SPIHT),” *IEEE Trans. on Circuits and Systems for Video Tech.*, vol.. 10, pp. 1374 - 1387, Dec. 2000.

[25] A. Secker and D. Taubman, “Lifting-based invertible motion adaptive transform (LIMAT) framework for highly scalable video compression,” *IEEE Trans. Image*

Processing, vol. 12, pp. 1530 - 1542, December 2003.

[26] J. Shapiro, "Embedded image coding using zero trees of wavelet coefficients," *IEEE Trans. Signal Processing*, vol. 41, pp. 3445 - 3463, Dec. 1993.

[27] Y. He, F. Wu, S. Li, Y. Zhong and S. Yang, "H.26L-based fine granularity scalable video coding," in *ISCAS 2002*, Scottsdale, AZ, May 2002.

[28] B. Banister, B. Belzer and T. Fischer, "Robust image transmission using JPEG2000 and Turbo-codes," in *Proc. ICIP'00*, Vancouver, BC, Canada, pp. 375 - 378, Sep. 2000.

[29] P. G. Sherwood and K. Zeger, "Progressive image coding for noisy channels", *IEEE Signal Processing Letters*, vol. 4, pp. 189 - 191, Jul. 1997.

[30] A.D. Liveris, C.-F Lan, K. R. Narayanan, Z. Xiong and C.N. Georghiades, "Slepian-Wolf coding of three binary sources using LDPC codes," in *Proc. Intl. Symp. on Turbo Codes and Related Topics*, Brest, France, pp. 63 - 66, Sep. 2003.

[31] A. D. Liveris, Z. Xiong and C. N. Georghiades, "Compression of binary sources with side information at the decoder using LDPC codes", *IEEE Commun. Letters*, vol. 6, pp. 440 - 442, Oct. 2002.

[32] M. Yang, W. E. Ryan, and Y. Li, "Design of efficiently encodable moderate-length high-rate irregular LDPC codes," *IEEE Trans. Commun.*, vol.. 52, pp. 564 - 571, Apr. 2004.

[33] T. J. Richardson and R. L. Urbanke, "Efficient encoding of Low-Density Parity-Check codes", *IEEE Trans. on Information Theory*, vol. IT-47 , pp. 638 - 656, Feb. 2001.

[34] J. Li and K. R. Narayanan, "Rate-Compatible low Density Parity Check codes for capacity-approaching ARQ schemes in packet data communications," in *Proc. Intl. Conf. on Communcations, Internet and Information Technology* (CIIT), Virgin Islands, US, pp. 201 - 206, Nov. 2002.

[35] R. Narayananswami, Coded modulation with Low-Density Parity-Check codes, M.S. thesis, Texas A&M University, College Station, 2001.

[36] D. Heckeman and M. P. Wellman, "Bayesian network," *Commun. of the ACM,* vol.. 38, pp. 27 - 30, 1995.

[37] J. H. Kim and J. Pearl, "A computational model for combined causal and diagnostic reseasoning in inference systems," in *Proc. 8th Intl. Joint Conf. AI*(IJCAI83), Karlsruhe, Germany, pp 190 - 193, 1983.

[38] J. Pearl, "Reverend Bayes on inference engines: A distributed hierarchical approach," in *Proc. Conf. Nat. Conf. AI*, Pittsburgh, PA, pp. 133 - 136, 1982.

[39] J. Pearl, "Fusion, propogation, and structuring in belief network", *Artif. Intell.*, vol.. 29, pp. 241 - 288, 1986.

[40] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, San Mateo, CA: Morgan Kaufman, 1988.

[41] N. Wiberg, "Codes and decoding on general graph," Ph.D. dissertation, Linkoping University, Linkoping, Sweden, 1996.

[42] S. Chung, T. Richardson, and R. Urbanke, "Analysis of sum-product decoding of Low-Density Parity-Check codes using a Gaussian approximation," *IEEE Trans. Inform. Theory*, vol. IT-47, pp. 657 - 670, Feb. 2001.

[43] S. ten Brink, "Design of serially concatenated codes based on iterative decoding convergence", in *Proc. of International Symposium on Turbo Codes and Related Topics*, Brest, France, pp. 319 - 322, Sep. 2000.

[44] A.D. Wyner, "Recent results in the Shannon theory," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 2 - 10, Jan. 1974.

[45] R. Hamzaoui, V. Stankovic, and Z. Xiong, "Rate-based versus distortion-based optimal joint source-channel coding," in *Proc. DCC'02*, Snowbird, UT, pp. 63 - 72, Apr. 2002.

[46] B. Haskell, A. Puri, and A. Netravali, *Digital Video: An Introduction to MPEG-2*, New York: Chapman & Hall, 1997.

[47] MPEG-4 Video VM, ver. 13.0, ISO/IEC JTC 1/SC29/WG11 N2687, Mar. 1999.

[48] ITU-T Q.16/SG16, "H.26L Test Model Long Term Number 9 (TML-9) draft0,", Doc. VCEG-N83D1, Dec. 2001.

[49] V. Chande and N. Farvardin, "Progressive transmission of images over memoryless noisy channels," *IEEE J. Selected Areas in Commun.*, vol. 16, pp. 850 - 860, Jun. 2000.

[50] A. Mohr, E. Riskin, and R. Ladner, "Graceful degradation over packet erasure channels through forward error correction," in *Proc. DCC'99*, Snowbird, UT, pp. 92 - 101, Mar. 1999.

[51] P. Cosman, J. Rogers, P. Sherwood, and K. Zeger, "Combined forward error control and packetized zerotree wavelet encoding for transmission of images over varying channels," *IEEE Trans. Image Processing*, vol. 9, pp. 982 - 993, Jun. 2000.

[52] P. A. Chou, A. E. Mohr, A. Wang, and S. Mehrotra, "Error control for receiver-driven layered multicast of audio and video," *IEEE Trans. on Multimedia*, vol. 3, pp. 108 - 122, Mar. 2001.

[53] J. Hagenauer, "Rate-compatible punctured convolutional codes (RCPC Codes) and their applications," *IEEE Trans. Commun.*, vol. 36, pp. 389 - 400, Apr. 1988.

[54] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near shannon limit error-correcting coding and decoding: Turbo-codes," in *Proc. IEEE Int. Conf. Commun.*, Geneva, Switzerland, pp. 1064 - 1070, May 1993.

[55] B. Hochwald and K. Zeger, "Tradeoff between source and channel coding," *IEEE Trans. Inform. Theory*, vol. IT-43, pp. 1412 - 1424, Sep. 1997.

[56] University of California at Los Angeles (UCLA) Image Commun. Lab, "Wavlet image coding: PSNR results," Web site http://www.icsl.ucla.edu/~ipl/psnr_results.html, 1998.

[57] D. Taubman, "High performance scalable image compression with EBCOT," *IEEE Trans. Image Processing*, vol. 9, pp. 1158 - 1170, Jul. 2000.

[58] I. Witten, R. Neal, and J. Cleary, "Arithmetic coding for data compression," *Commun. of the ACM*, vol. 30, pp. 520 - 540, Jun. 1987.

[59] ITU-T Recommendation H.263, version 2, *Video coding for low bitrate communication,* Jan. 1998.

[60] D. Divsalar and F. Pollara, "Multiple Turbo Codes for Deep-Space Commun.," *TDA Progress Report 42 - 121.* Jet Propulsion Laboratory, California Institute of Technology, 1995.

[61] P. Roberston, E. Villebrun, and P. Hoeher, "A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain," in *Proc. ICC*, Vancouver, Canada, pp. 1009 - 1013, Jun. 1995.

[62] J. L. Massey, "Joint source channel coding", in *Communication Systems and Random Process Theory*, J. K. Skwirzynski, ed., pp. 279 - 293, Amsterdam, Netherlands: Sijthoff and Nordhoff, 1978.

[63] Joachim Hagenauer, "Rate-compatible punctured convolutional codes (RCPC codes) and their applications," *IEEE Trans. Commun.*, vol. COM-36, pp. 389 - 400, Apr. 1988.

[64] Special issue on error-resilient image and video transmission, *IEEE J. Selected Areas in Commun.*, vol. 18, Jun. 2000.

[65] Special issue on wireless video, *IEEE Trans. Circuits and Systems for Video Tech.*, vol. 12, Jun. 2002.

[66] Special issue on streaming media, *IEEE Trans. Multimedia*, vol. 6, Apr. 2004.

[67] Q. Chen and T. R. Fischer, "Image coding using robust quantization for noisy digital transmission", *IEEE Trans. Image Processing*, vol.. 7, pp. 496 - 505, Apr. 1998.

[68] T.-T. Lam, G. P. Abousleman and L. J. Karam, "Image coding with robust channel-optimized trellis-coded quantization", *IEEE Journal on Selected Areas in Commun.*, vol. 18, pp. 940 - 951, Jun. 2000.

[69] A. Kurtenbach, P. Wintz, "Quantizing for noisy channels", *IEEE Trans. Commun.*, vol. COMM-17 , pp. 291 - 302, 1969.

[70] N. Farvardin and V. Vaishampayan, "Optimal quantizer design for noisy channels: An approaches to combined source-channel coding", *IEEE Trans. Inform. Theory*, vol.: IT-33, pp. 827-838, Nov. 1987.

[71] N. Farvardin and V. Vaishampayan, "On the performance and complexity of channel-optimized vector quantizers", *IEEE Trans. Inform. Theory*, vol: IT-37, pp. 155 - 160, Nov. 1991.

[72] H. Kumazawa, M. Kasahara and T. Namekawa," A construction of vector quantizers for noisy channels", *Electron. Eng. JPN.*, vol. 67-b, pp. 39 - 47, 1984.

[73] K. Zeger and A. Gersho, "Vector quantizer design for memoeyless noisy channels", in *Proc. ICC*, Philadelphia, PA, pp. 1593 - 1597, 1988.

[74] A. Ayanoğlu and R. M. Gray, "The design of joint source and channel trellis waveform coders", *IEEE Trans. Inform. Theory*, vol. 33, pp. 855 - 865, Nov. 1987.

[75] M. Wang and T. R. Fischer, "Trellis coded quantization designed for noisy channels", *IEEE Trans. Inform. Theory*, vol. 40, pp. 1792 - 1802, Nov. 1994.

[76] A. Huebner, J. Freudenberger, R. Jordan, and M. Bossert, "Irregular turbo codes and unequal error protection," in *Proc. ISIT'01*, Washington D.C., pp. 24 - 29, Jun. 2001.

[77] G. Caire and E. Biglieri, "Pararrel concatenated codes with unequal error protection," *IEEE Trans. Commun.*, vol. 46, pp. 565 - 567, May 1998.

[78] B. Vasic, "Combinatorial construction of low-density parity check codes for iterative decoding," submitted to *IEEE Trans. Inform. Theory.*

[79] B. Vasic, A. Cvetkovic, S. Sankaranarayanan and M. Marcellin "Adaptive error protection Low-Density Parity-Check codes for joint source-channel coding", in *Int. Symp. on Information theory*, pp. 267, Yokohama, Japan, Jun. 2003.

[80] C. Lan, K. R. Narayanan and Z. Xiong, "Scalable image transmission using rate-compatible irregular repeat accumulate (IRA) Codes," in *Proc. ICIP'02*, pp. 717 - 720, Rochester, New York, Sep. 2002.

[81] C. Lan, T. Chu, K. R. Narayanan and Z. Xiong, "Scalable image and video transmission using irregular repeat accumulate (IRA) codes with fast algorithm for optimal unequal error protection," *IEEE Trans. Commun.*, vol. 52, pp. 1092 - 1101, Jul. 2004.

[82] S. Mallat, F. Falzon, "Analysis of low bit rate image tranform coding," *IEEE Trans. Signal Processing*, vol. 46, No. 4, pp. 1027 - 1042, Apr. 1998.

[83] L. Lastras and T. Berger, "All sources are nearly successively refinable", *IEEE Trans. Inform. Theory*, vol. IT-47 , pp 918 - 926, Mar. 2001.

[84] J. K. Wolf, "Data reduction for multiple correlated sources," in *Proc. 5th Colloquium Microwave Communication*, Budapest, Hungary, pp. 287 - 295, Jun. 1973.

[85] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): Design and construction", in *Proc. IEEE DCC'99*, Snowbird, UT, pp. 158 - 167, Mar. 1999.

[86] J. Garcia-Frias and Y. Zhao, "Compression of correlated binary sources using turbo codes," *IEEE Commun. Letters*, vol. 5, pp. 417 - 419, Oct. 2001.

[87] A. Aaron and B. Girod, "Compression with side information using turbo codes", in *Proc. IEEE DCC'02*, Snowbird, UT, pp. 252 - 261, Apr. 2002.

[88] C. Lan, K. R. Narayanan and Z. Xiong, "Source-optimized irregular repeat accumulate codes with inherent unequal error protection capabilities and their application to progressive image transmission," in *Proc. 37th Asilomar Conference on Signals, Systems, and Computers*, pp. 1505 - 1509, Pacific Grove, CA, Nov. 2003.

[89] J. Bajcsy and P. Mitran, "Coding for the Slepian-Wolf problem with turbo codes," in *Proc. IEEE GLOBECOM*, San Antonio, Tx, pp. 1400–1404, Nov. 2001.

[90] T. Murayama, "Statistical mechanics of the data Compression theorem," *J. Phys. A: Math. Gen.*, vol. 35, pp. L95 - L100, Mar. 2002.

[91] A.D. Liveris, Z. Xiong and C.N. Georghiades, "Distributed compression of binary sources using conventional parallel and serial concatenated convolutional codes," in *Proc. IEEE DCC'03*, Snowbird, UT, pp. 193–202, Mar. 2003.

[92] D. Schonberg, K. Ramchandran and S.S. Pradhan, "LDPC codes can approach the Slepian-Wolf bound for general binary sources", in *Proc. Allerton Conf. Commun. and Control*, Monticello, IL, Oct. 2002.

[93] S. S. Pradhan and K. Ramchandran, "Distributed code constructions for the entire Slepian-Wolf rate region for arbitrarily correlated sources," in *Proc. DCC'04*, Snowbird, UT, pp 303 - 311, Mar. 2004.

[94] V. Stanković, A. D. Liveris, Z. Xiong and C. N. Georghiades, "Design of Slepian-Wolf codes by channel code parttioning," in *Proc. DCC'04*, Snowbird, UT, pp

303 - 311, Mar. 2004.

[95] S. S. Pradhan and K. Ramchandran, "Distributed source coding: Symmetric rates and applications to sensor networks," in *Proc. DCC'00*, Snowbird, UT, pp 363 - 372, Mar. 2000.

[96] S. ten Brink "Design of low-Density-Parity-Check codes for modulation and detection, submitted to " *IEEE Trans. Inform. Theory*, Jun. 2002, Available at http://cm.bell-labs.com/who/gkr/pub.html.

[97] M. C. Davey and D. J. C. MacKay, "Low Density Parity Check Codes over GF(q)," *IEEE Commun. Letters*, pp. 165 - 167, Jun. 1998.

[98] R. Zamir, S. Shamai (Shitz) and U. Erez, "Nested linear/lattice codes for structured multiterminal binning," *IEEE Trans. Inform. Theory*, vol. IT-48, pp. 1250 - 1276, Jun. 2002.

APPENDIX[1]

Fast Algorithm for Optimal UEP

Generally speaking, the objective of optimal UEP for multimedia data transmission is to minimize the average MSE under a transmission rate constraint and known channel condition. Because of the huge number of all candidate solutions, such an optimization problem is generally very time-consuming. Exhaustive search is thus prohibitive and various algorithms have been developed. With fixed source block size and variable channel codeword length, Chande and Farvardin showed how to solve this JSCC problem exactly using dynamic programming [49]. It can also be converted into a Lagrangian minimization problem and solved using an iterative descent algorithm [52]. On the other hand, when the channel codeword length is fixed, a Viterbi algorithm is used in [28] to find the optimal rate allocation.

These algorithms have proved to be efficient at the price of relatively high complexity. In [45], however, the authors showed that a rate-optimal solution concatenated with a fast local search algorithm can be used to achieve near-optimal PSNR performance, while its time complexity was lower than that of all previously proposed algorithms. Consequently, we adopt the rate-optimal solution in [49] and combine it with a modified local search algorithm.

After source and channel coding, each packet has a fixed length of 4136 bits (517

---

[1]©2004 IEEE. Reprinted with permission, from *IEEE Transactions Communications*, vol. 52, pp. 1092 - 1101, July 2004.

[2]A JSCC solution is called rate-optimal if it maximizes the expected number of correctly received source bits. Correspondingly, a distortion-optimal solution minimizes the expected MSE.

[3]PSNR is defined as PSNR $= 10\log_{10}(255^2/\text{MSE})$ and measured in decibels (dB).

bytes), with the ith packet consisting of R_i bit from the source bit stream, 8 bits for specifying next packet's channel coding rate and 16 CRC bits for error detection, and C_i parity bits. So we have $R_i + C_i = 4112$ for $1 \leq i \leq N$, with N being the total number of packets. If the decoder detects an error in any packet, the decoding process is stopped and the source is reconstructed from the correctly received source bits. We assume that all errors can be detected.

Borrowing notation from [45], now we have m (in this case $m = 7$) channel codes $c_1, c_2, ..., c_m$ which are constructed in the above section. Let $\mathcal{R}$ be the set of corresponding code rates $r_1 < r_2 < ... < r_m$. For $i = 1, 2, ..., m$, let $p(r_i)$ denote the probability of a decoding error in a packet protected by code c_i, which is given in Fig. 8 as a function of the channel BER. Suppose now that we want to send N packets of $L = 4136$ bits with 24 bits overhead in each packet. We have a rate allocation vector $\mathbf{R} = (r_{k_1}, r_{k_2}, ..., r_{k_N})$, which assigns to each packet i, $i = 1, 2, ..., N$, a channel code rate $r_{k_i} \in \mathcal{R}$. The number of source bits in the ith packet is given by $v(r_{k_i}) = Lr_{k_i} - 24$. For $i = 1, ..., N-1$, $P_i(\mathbf{R}) = \prod_{j=1}^{i}(1 - p(r_{k_j}))p(r_{k_{i+1}})$ is the probability that no errors occur in the first i packets but with an error in the next one, $P_0(\mathbf{R}) = p(r_{k_1})$ is the probability of an error in the first packet, and $P_N(\mathbf{R}) = \prod_{j=1}^{N}(1 - p(r_{k_j}))$ is the probability that all N packets are correctly received. Then the expected distortion is given by

$$E_N\{d\}(r_{k_1}, r_{k_2}, ..., r_{k_N}) = \sum_{i=0}^{N} P_i(\mathbf{R})d_i(\mathbf{R}), \tag{A-1}$$

where $d_0(\mathbf{R}) = d_0$ is the source variance, and for $i \geq 1$, $d_i(\mathbf{R})$ is the reconstruction distortion using the first i packets.

Direct minimization of the expected distortion given in (A-1) is relatively complex a problem to solve. Alternatively, according to the analysis in [45], we can first get a rate-optimal solution as the starting point and then use a local search algorithm to

find a local minimum of (A-1).

The rate-optimal solution maximizes the expected number of correctly received source bits

$$E_N\{r\}(r_{k_1}, r_{k_2}, ..., r_{k_N}) = \sum_{i=0}^{N} P_i(\mathbf{R})V_i(\mathbf{R}), \tag{A-2}$$

where $V_0(\mathbf{R}) = 0$ and for $i \geq 1$, $V_i(\mathbf{R}) = \sum_{j=1}^{i} v(r_{k_j})$ with $v(r_{k_j}) = Lr_{k_j} - 24$ being the number of source bits in the jth packet, then a rate-optimal solution can be easily computed in $O(N)$ time [49].

Starting from this rate allocation, the next step is to perform local search to find the distortion-optimal solution. By modifying the fast local search algorithm in [45], we search the minimum expected distortion as following.

Modified local search algorithm

1. Set $k = 1$ and $n = 1$. Let $\mathbf{R} = \mathbf{R}_c = \mathbf{R}_0$, where $\mathbf{R}_0$ is the rate-optimal solution.

2. If $n > 3$, stop. Otherwise let r be the kth highest rate used by $\mathbf{R}_c$. Find j that is the index of the first packet of rate r in $\mathbf{R}_c$.

3. If $r = r_1$, stop. Otherwise, let $r_c \in \mathcal{R}$ be the highest rate smaller than r, set the jth component of $\mathbf{R}_c$ to r_c.

4. If $E_N\{d\}(\mathbf{R}_c) < E_N\{d\}(\mathbf{R})$, set $\mathbf{R} = \mathbf{R}_c$ and goto step 2.

5. If $j = N$ and r_c is the equal to the rate of packet $j - 1$, set $k = k + 1$ and $\mathbf{R}_c = \mathbf{R}$; if $j = 1$ and $r_c = r_1$, set $k = 1$, $n = n + 1$ and $\mathbf{R}_c = \mathbf{R}$; go to step 2.

The resulting vector $\mathbf{R}$ contains a distortion-optimal solution. Experiments show that this local search algorithm achieves comparable performance to the Viterbi algorithm but with much lower complexity.

VITA

Ching Fu Lan received the B.S. degree in electrical engineering from National Central University, Taiwan, in 1994, and the M.S. degree in communication engineering from National Chiao-Tung University, Taiwan, in 1996.

From 1996 to 1998, he served as Second Lieutenant in Army, Taiwan. From 1998 to 2000, he was an Assistant Researcher with Chung-Hwa Telecommunication Labs, Taiwan. He was a Research Assistant with Department of Electrical Engineering at Texas A&M University from 2001 to 2004. He received the Ph.D. degree in electrical engineering from Texas A&M University in 2004.

His research interest falls in the general area of communication engineering and digital signal processing, with specific focus on multi-carrier communication systems, equalization, channel coding and coded modulation for wireline/wireless communications.

He was also the recipient of the Texas Telecommunication Engineering Consortium (TxTEC) Scholarship (2001). He can be reached at:

No. 9 Hua-Yuan 2nd Rd. Sec 3, Hsin-Tien, Taipei, Taiwan.

The typist for this thesis was Ching Fu Lan.