University of California
Los Angeles

Universal Serially Concatenated Trellis Coded Modulations and Rate-Compatible High-Rate LDPC Codes

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Electrical Engineering

by

Wen-Yen Weng

2006

The dissertation of Wen-Yen Weng is approved.

Richard S. Elman

Gregory J. Pottie

Kung Yao

Richard D. Wesel, Committee Chair

University of California, Los Angeles

2006

To my family and friends

Table of Contents

List of Figures

List of Tables

Vita

August 17, 1975	Born, Taichung, Taiwan
1997	B.S., Electrical Engineering National Taiwan University Taipei, Taiwan
2001	M.S., Electrical Engineering University of California, Los Angeles Los Angeles, California
2001–2006	Graduate student researcher Electrical Engineering Department University of California, Los Angeles
2004	Summer intern Conexant Systems Inc. Los Angeles, California
2006	Ph.D., Electrical Engineering University of California, Los Angeles Los Angeles, California

Publications

C. Köse, W. Weng and R. D. Wesel, "Serially Concatenated Trellis Coded Modulation for the Compound Periodic Erasures Channel," ICC 2003, Anchorage,

Alaska, May 2003.

A. I. Vila Casado, W. Weng and R. D. Wesel, "Multiple Rate Low-Density Parity-Check Codes with Constant Block Length," Asilomar Conf. on Signals, Systems and Computers, Pacific Grove, CA, 2004.

W. Weng, A. Ramamoorthy and R. D. Wesel, "Lowering the Error Floors of High-Rate LDPC Codes by Graph Conditioning," IEEE VTC 2004, Los Angeles, California.

M. Griot, A. I. Vila Casado, W. Weng, H. Chan, J. Basak, E. Yablonovitch, I. Verbauwhede, B. Jalali, and R. D. Wesel, "Trellis Codes with Low Ones Density for the OR Multiple Access Channel," IEEE ISIT, Seattle, USA, July 2006.

A. I. Vila Casado, S. Valle, W. Weng, and R. D. Wesel, "Constant-Blocklength Multiple-Rate LDPC Codes for Analog-Decoding Implementations," Analog Decoding Workshop, June 2006.

A. I. Vila Casado, W. Weng, S. Valle and R. D. Wesel, "Multiple-Rate Low-Density Parity-Check Codes with Constant Blocklength," accepted by IEEE Transactions on Communications on April 24, 2006.

W. Weng, B. Xie and R. D. Wesel, "Universal Space-Time Serially Concatenated Trellis Coded Modulations," Globecom 2006, San Francisco, USA.

Abstract of the Dissertation

Universal Serially Concatenated Trellis Coded Modulations and Rate-Compatible High-Rate LDPC Codes

by

Wen-Yen Weng

Doctor of Philosophy in Electrical Engineering

University of California, Los Angeles, 2006

Professor Richard D. Wesel, Chair

This dissertation presents you universal serially concatenated trellis-coded modulations (SCTCMs) that perform consistently close to the available mutual information for the periodic erasure channel (PEC), the periodic fading channel (PFC) and the 2×2 compound matrix channel. For the PEC and PFC, the universal SCTCMs extend the near-capacity performance of serially concatenated convolutional codes under AWGN to periodically time-varying channels. We use both maximum-likelihood decoding criteria and iterative-decoding criteria to design these universal SCTCMs. Each component of the SCTCM including the constituent codes, constellation, labeling, and interleaver are carefully chosen to achieve universality.

For the space-time channel, by de-multiplexing the symbols across the antennas, universal SCTCMs for the period-2 PFC deliver consistent performance over the eigenvalue skew of the matrix channel. Within the family of channels having the same eigenvalue skew, a time-varying linear transformation (TVLT) is used to mitigate the performance variation over different eigenvectors. Because of

their consistent performance over all channels, the proposed codes will have good frame-error-rate (FER) performance over any quasi-static fading distribution.

A graph-conditioning algorithm called the approximate cycle extrinsic message degree (ACE) algorithm is used to construct high-rate ($R \geq 1/2$) irregular LDPC codes. For high-rate LDPC codes, due to the large number of degree-2 variable nodes in the optimal degree distribution, it is more difficult to condition the graph. By constraining the number of degree-2 nodes, the ACE algorithm can dramatically lower the error floor with little compromise of the threshold. The same design criteria are suitable for rate-compatible applications using information-nulling.

Another rate-compatible technique uses row-combining, which combines rows of a lower-rate parity-check matrix to form one or more higher-rate parity-check matrices according to predefined combining rules. The resulting LDPC codes are called Constant Blocklength Multiple-Rate (CBMR) LDPC codes. This row-combining approach fits well with an efficient hardware architecture known as the irregular partitioned permutation (IPP) LDPC code. We identified all the constraints that the IPP codes and row-combining placed on the parity-check matrix and designed the row-combined IPP (RC-IPP) codes. The implementation issues of encoding and decoding for RC-IPP codes are exploited. As a result, the RC-IPP codes have FER as good as that of the stand-alone codes at each rate.

CHAPTER 1

Universal Serially Concatenated Trellis Coded Modulation for Periodic Erasures and Periodic Fading

1.1 Introduction

Often, design of channel codes focuses on the optimization of performance on a specific channel such as the additive white Gaussian noise (AWGN) channel. Powerful error-correcting codes such as turbo codes [6] and Low-Density Parity-Check (LDPC) codes [7] have been shown to operate within a dB of the Shannon limit on the AWGN channel. However, the performance can degrade significantly over some specific channel realizations. Therefore, channel codes that extend this performance to other channels of practical interest are highly desirable.

Root and Varaiya's compound channel coding theory for linear Gaussian channels [8] indicates that a single code can reliably transmit information at R bits/symbol on each channel in the ensemble of linear Gaussian channels with mutual information (MI) larger than the attempted rate. In related work, Sutskover and Shamai [9] recently proposed a setting for decoding of LDPC codes jointly with channel estimation for transmission over memoryless compound channels.

In [10, 11] the term *universal* was used to describe a channel code that has a good bit-error-rate (BER) or frame-error-rate (FER) for every channel in a

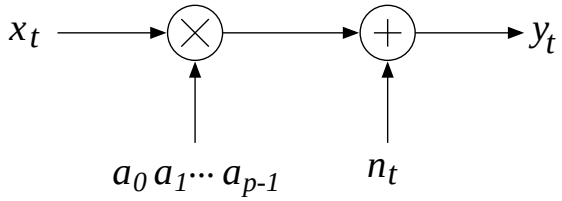


Figure 1.1: Channel model of the period-p PFC. The fading of the channel varies periodically over time with period p.

family, $\mathcal{H}$, of channels. The code is said to be universal over $\mathcal{H}$. These papers presented trellis codes that approach universal behavior on space-time channels with a proximity to the Shannon limit similar to that with which trellis codes approach capacity on the AWGN channel.

Wesel et al. [12], [13] identified universal trellis codes (TCM) for the periodic fading channel (PFC) as shown in Figure (1.1), where x_t is the transmitted symbol with average energy E_s, n_t is the additive white Gaussian noise (AWGN) with variance $N_0/2$ per dimension, and y_t is the received symbol. The vector $\underline{a} = [a_0\ a_1\ \cdots\ a_{p-1}]$ describes the nature of the time-varying attenuation behavior of the channel.

Such channels appear in frequency-hopped or multicarrier transmissions as well as diagonally-layered space-time architectures [14]. In a multi-carrier modulation system, the period of the fading is the number of sub-carriers. In a frequency-hopped spread spectrum system, the period of the fading is the period of the hopping pattern. Motivated by [12] and [13], it is natural to extend the universal TCM scheme to a universal SCTCM scheme and take advantage of the capacity-achieving capability of turbo codes to further improve the compound channel performance.

As a subset of the PFC, the periodic erasure channel (PEC) has the symbols

either transmitted without any attenuation or has it completely erased, i.e., $a_i \in \{0, 1\}$, for $i = 0, 1, \cdots, p-1$. The reason that the PEC is considered is because there are fewer instances of channels in the PEC family to be considered than that of the PFC and we have found that the erasures seem to impose the most critical conditions for the code that results in the worse case performance in the PFC family [14] [10].

In this chapter, serially concatenated trellis coded modulations (SCTCMs) are presented with universal behavior on the period-2 PEC and the period-2 PFC. The period-2 PFC is specified by

$$y_t = a_{(t \bmod 2)} x_t + n_t, \tag{1.1}$$

where the attenuation vector is $\underline{a} = [a_0 \ a_1]$. Assuming $a_0 \geq a_1$, the period-2 PFC, $\underline{a} = [a_0 \ a_1]$, can be normalized as the $[1 \ \mathbf{q}]$ channel with $0 \leq \mathbf{q} \leq 1$ where $\mathbf{q} = a_1/a_0$.

As a subset of the PFC, the period-2 PEC consist of two instances of the attenuation vector. When $\underline{a} = [1 \ 0]$, every other symbol is erased, and when $\underline{a} = [1 \ 1]$, there are no erasures, the channel is the standard AWGN channel. We have found (see [14] Section IV and [10] Theorem 3) that the performance on the normalized $[1 \ \mathbf{q}]$ channel is approximately bounded by the $[1 \ 1]$ and $[1 \ 0]$ channels. Hence, the problem of designing universal SCTCMs for the PFC can be simplified to designing SCTCMs for the PEC.

Section 1.2 defines the channel model and the performance measure for universality. Section 1.3 extends the existing AWGN design rules for SCTCM schemes under maximum-likelihood (ML) decoding to periodic erasure channels. Section 1.4 presents design criteria for universal SCTCMs under iterative decoding with particular emphasis on the inner TCM. The iterative decoding criteria includes the selection of constellation labeling, constituent code complexity, design

of interleaver and the inner code. Section 1.5 provides universal SCTCMS for PEC transmitting at 0.5, 1 and 1.5 bits/symbol and demonstrates their universality on the PFC. Section 1.6 concludes the chapter.

1.2 Channel Models and Performance Measure

As in [10], [12], and [14], we use the excess mutual information (EMI) to measure the proximity with which the code approaches the theoretical limit of the channel. For the period-2 PFC and PEC in Eq. (1.1), the mutual information (MI) between the input symbol x and output symbol y with an equal allocation of power, E_s, to each symbol in the period is

$$\mathrm{MI}\left(\underline{a}, E_s\right) = \frac{1}{2}\sum_{i=0}^{1}\log_2\left(1+\frac{|a_i|^2 E_s}{N_0}\right). \tag{1.2}$$

According to the compound channel coding theorem [8], if MI $\geq R$, the error probability of a universal code can decrease to zero in the limit as its blocklength goes to infinity.

In practice, we design a finite-blocklength code transmitting at R-bits per symbol that achieves its target BER, say 10^{-5}, at $\mathrm{SNR}^* = E_s^*/N_o$. The asterisks (*) indicate that this is the SNR at which the target BER is achieved. Then, we define the EMI of this code for the PFC as:

$$\mathrm{EMI}(\underline{a}) = \mathrm{MI}(\underline{a}, E_s^*) - R. \tag{1.3}$$

As in [10], [12], and [14], the MI we refer to is the unconstrained MI, i.e., the MI achieved using a Gaussian codebook. For example, a 64-state trellis code of [12] (Code 1) requires $E_s/N_0 = 4.3$ dB to achieve BER $= 10^{-5}$ on the AWGN channel. Using Eq. (1.2) and (1.3), this corresponds to an EMI of 0.88 bits. The

same code requires $E_s/N_0 = 9.4$ dB at BER $= 10^{-5}$ when every other symbol is erased. The EMI under periodic erasures is 0.64 bits.

In [1], the constrained MI, i.e., the MI achieved using a codebook with symbols from that constellation, is used as the reference because it is the true theoretic limit once we have selected the constellation. However, a desired throughput, R bits/symbol, can be achieved using different constellation sizes, n bits/symbol, with corresponding code rates R/n. Then, the EMIs computed using the constrained MI cannot be compared directly for different schemes. Therefore, we choose the unconstrained MI over the constrained MI for convenience in comparing SCTCMs transmitting at the same throughput but with different constellation sizes.

At a fixed BER and with a fixed input information blocklength, the universal channel coding problem is a multicriterion optimization problem. Our goal is to find an SCTCM that is a Pareto optimal over all channels in the family. Pareto optimality means that no SCTCM performs better on every channel in the family. Typically, there will be several Pareto optimal SCTCMs. So, an objective function is required for selecting among the Pareto optimal codes.

One possible objective function is the maximum EMI requirement over all the channels in $\mathcal{H}$, and hence the problem becomes a minimax problem. Although Root and Varaiya [8] showed that EMI goes to zero over each of the channels in the ensemble when the blocklength goes to infinity, it is still an open problem how fast the EMI decreases as a function of the blocklength and whether the EMI are the same for all channels. Empirical evidence thus far [10], [12], [14] indicates that universal codes tend to have relatively constant EMIs when the unconstrained MI is chosen as the reference. As a result of this constant EMI

property, it is natural to use the minimax criteria.

$$\min \max_{H_i \in \mathcal{H}} \mathrm{EMI}(H_i). \tag{1.4}$$

Another possible objective function is the average EMI per channel.

$$\overline{\mathrm{EMI}} = \frac{1}{|\mathcal{H}|} \sum_i \mathrm{EMI}(H_i), \tag{1.5}$$

This is equivalent to the objective function, J_{MI}, in [12] to minimize the sum of the MIs over the channel family. Since a universal code performs consistently well on every channel in the family, we would expect that a universal code also has minimum average EMI. Furthermore, average EMI criteria remains a good objective function whether constrained or unconstrained MI is used. As a result, we choose unconstrained MI as the reference and seek the code that minimizes the average EMI as the universal channel code. Note that when unconstrained MI is used as the reference, minimax and average EMI criteria yield almost the same universal SCTCM performance.

To verify the universality of the proposed codes, codes are simulated over all possible realizations of the channel ensemble. For the period-2 PEC, there are only two instances. As for the period-2 PFC, which has infinite channel realizations, we use a fine sampling of the channel space under the assumption of continuity. For the normalized [1 $\mathbf{q}$] PFC, the sampling is done over $\mathbf{q}$ between 0 and 1.

With turbo codes, it is possible to achieve lower EMI figures than the trellis codes at similar BERs. The serially concatenated scheme is especially attractive in designing for the compound channel because universality can be imposed on the inner code which directly interfaces the channel. However, if we pick an binary serially concatenated convolutional codes (SCCC) in the literature [3] and map it to an 8PSK constellation, the SCTCM (SC-8, see Table 1.3) performs well

on AWGN, but it suffers under the periodic erasures. As seen in Figure 1.2 on the period-2 erasure channels, a universal SCTCM that we designed, SC-5 (see Table 1.3), requires slightly more EMI (0.012 bits, or equivalently 0.07 dB) on the [1 1] channel to achieve BER $= 10^{-5}$, but its [1 0] channel performance is much better (requiring 0.182 bits less, or 1.28 dB less than SC-8). The average EMI requirement of SC-5 is 0.280 bits compared to 0.365 bits of SC-8. This example illustrates the importance of code design to achieve universality.

1.3 Code Design Criteria under Maximum-Likelihood decoding

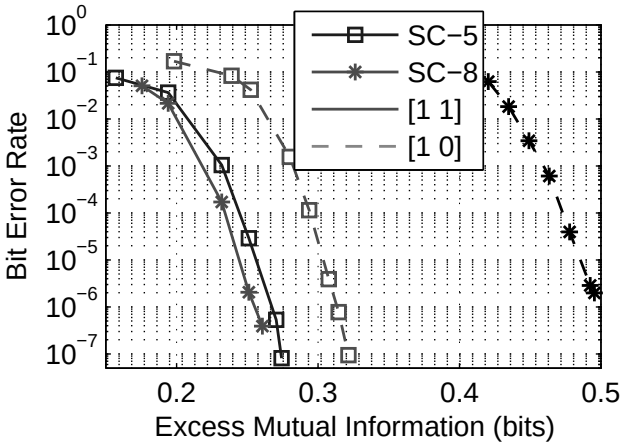
As shown in Figure 1.3, an SCTCM consists of a rate-R_o outer convolutional encoder (C_o), an interleaver, a rate-R_i inner convolutional encoder (C_i), and a two-dimensional 2^n-point constellation mapper. The throughput of the overall scheme is nR_oR_i bits per symbol. The uniform interleaver analysis [3] has shown that the BER under ML decoding decreases exponentially with the effective free Euclidean distance of the inner code. For a large interleaver size of N_{IL} bits

$$-\log \text{BER} \propto \lfloor (d_o+1)/2 \rfloor \log N_{IL} + \delta^2 \frac{E_s}{4N_0} \tag{1.6}$$

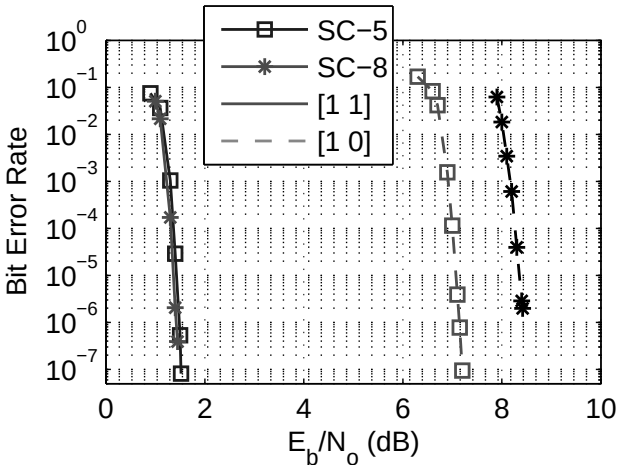
where d_o is the free Hamming distance of the outer code, and

$$\delta^2 = \begin{cases} \frac{1}{2} d_o d_i^2 & \text{for } d_o \text{ even} \\ \frac{1}{2}(d_o-3)d_i^2 + (h_m^3)^2 & \text{for } d_o \text{ odd} \end{cases} \tag{1.7}$$

where d_i denotes the effective free Euclidean distance of the inner code, defined to be the minimum Euclidean distance of error events caused by two information bit errors, and $h_m^{(3)}$ is the minimum Euclidean distance of error events caused by three information bit errors.



(a)



(b)

Figure 1.2: BER of universal SCTCM (SC-5) and non-universal SCTCM (SC-8) on the [1 1] and [1 0] channels as a function of (a) EMI and (b) SNR (E_b/N_o).

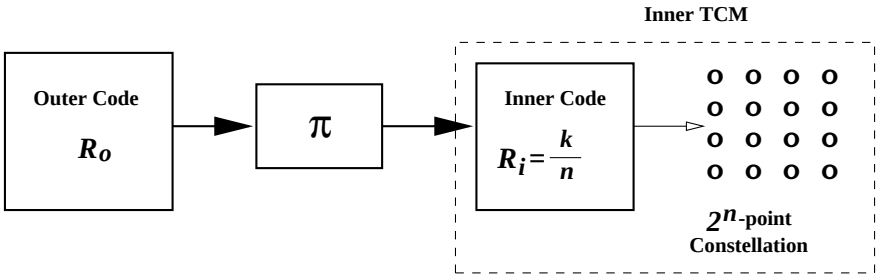


Figure 1.3: The SCTCM scheme.

The BER approximation in Eq. (1.6) extends to the period-2 PEC when the input to the channel is considered to be a vector of two consecutive symbols. Two necessary conditions for universal performance over the compound period-2 PEC are:

1. The concatenated code should have positive redundancy under periodic erasures, i.e., $2R_oR_i < 1$.

2. The inner TCM should have nonzero effective free Euclidean distance under periodic erasures. In particular, for a given number of states, the rate of the inner TCM should be small enough to avoid parallel transitions. This condition results from an extension of the uniform interleaver analysis of [3].

The specific requirement on rate, expressed as a bound on k, to avoid parallel transitions is described by the following lemma:

Lemma: For $k > 1$, an S-state rate-k/n inner encoder mapped to a 2^n-point constellation has *zero* minimum effective free Euclidean distance under erasure if

$$\binom{k}{2} > S. \tag{1.8}$$

Proof: For $k > 1$, the encoder trellis has $\binom{k}{2}$ transitions per state to describe the inputs with Hamming weight 2. If $\binom{k}{2} > S$, there are parallel transitions with

input Hamming weight 2 between states and shortest error events are 1-symbol long, and that symbol is subject to erasure. ■

For a 4-state inner code, the maximum k for nonzero d_i on the [1 0] channel is less than or equal to 3 and for an 8-state inner code, the maximum k for nonzero d_i is 4.

1.4 Code Design Criteria under iterative decoding

Unlike the ML-decoding design criteria which focus on maximizing the minimum distance, the iterative decoding design criteria focus on finding codes with low pinch-off thresholds. In general, iterative decoding design criteria employ Extrinsic Information Transfer (EXIT) charts [15] or density evolution [16] to predict the pinch-off thresholds of the SCTCMs. The height of the error floor depends on the minimum distance of the code and cannot be predicted from these techniques. Although not rigorously proved, it is generally observed by [7], [3], [17] that for turbo codes and LDPC codes, there exists a tradeoff between low pinch-off threshold and low error floor.

EXIT chart analysis predicts the waterfall region in the BER curve as the blocklength goes to infinity. The technique analyzes iterative decoding by tracking the density (probability distribution) of the extrinsic information of soft-input soft-output (SISO) *a posteriori* probability (APP) modules as this density evolves from iteration to iteration. In [18], El Gamal and Hammons showed that when the noise is Gaussian, the extrinsic information in the process of iterative decoding can be closely approximated by symmetric Gaussian distribution, where the variance is twice the mean. Using Gaussian approximation, the threshold of convergence of turbo codes can be predicted efficiently.

The threshold can indeed be predicted accurately in this way on the [1 1] channel. However, when the SCTCM is working on the [1 0] channel, the density of the extrinsic information at the output of the inner SISO doesn't follow a symmetric Gaussian distribution. This is because the erased bits tend to have lower log-likelihood ratios (LLRs) than that of the unerased bits. So, the overall pdf resembles a bi-modal, not necessarily symmetric, Gaussian mixture distribution. This effect makes it difficult to analyze the outer and inner SISO separately. The simple extrinsic information calculation resulting from the symmetric Gaussian approximation becomes invalid. The exact densities must be manipulated to analyze the SISOs. As a result, there is no computational advantage for this exact density evolution over explicit simulation. Hence, our approach of finding the threshold is to simulate the candidate SCTCMs and take the SNR with BER = 10^{-1} as its pinch-off threshold.

1.4.1 Design of the outer encoder

We use convolutional codes with large free Hamming distance as the outer code. For design simplicity and because it is the inner TCM that directly interfaces the channel, we fix the outer code and find the inner TCM such that the SCTCM has the lowest pinch-off threshold. Table 1.1 lists the rate-1/2, 2-, 4- and 8-state maximal free Hamming distance outer codes $C_{o,1}$, $C_{o,2}$, and $C_{o,3}$ respectively.

1.4.2 Design of the inner TCM

Although we cannot use EXIT chart analysis with Gaussian approximation to design SCTCMs with low thresholds on the [1 0] channel, some design guidelines developed from EXIT chart analysis still help. They are described as follows:

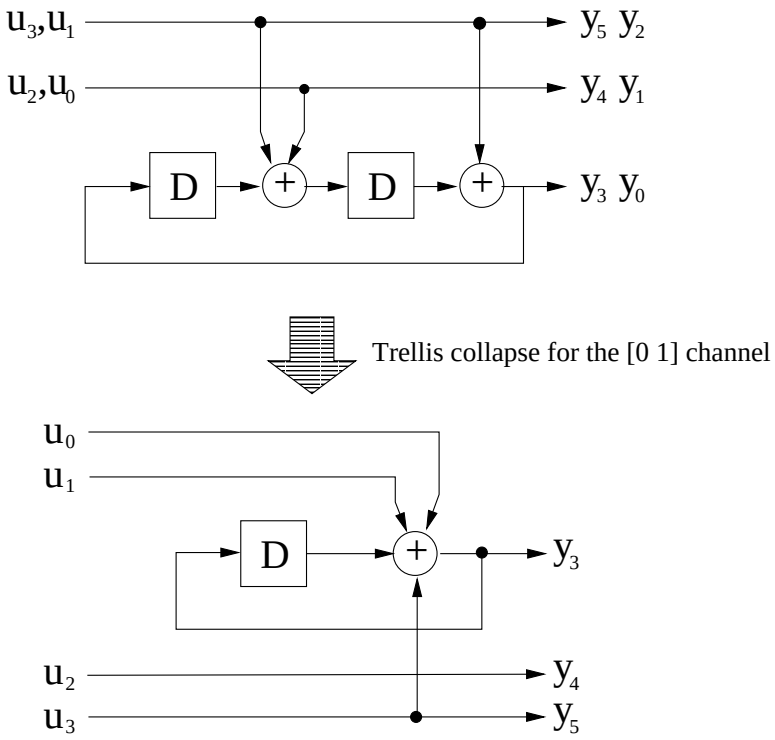


Figure 1.4: An example of state reduction. C_5^i, a 4-state, rate-2/3 inner code and its trellis-collapsed equivalent encoder under the [0 1] erasure channel.

- Complexity of constituent codes: It has been found that SCTCMs with lower complexity constituent codes tend to have lower pinch-off thresholds. However, the SCTCMs may suffer a high error floor because of the tradeoff. So, low complexity convolutional codes are preferred if the corresponding error floor is low enough to meet the BER requirement.

- Trellis-collapse check: An SCTCM that works on the [1 0] channel is equivalent to another SCTCM with twice the code rate working on the AWGN channel. In other words, because the erased symbols can't transmit any information at all, pairs of consecutive trellis stages collapse into a single stage. Although in general Gaussian Density Evolution works for SCTCMs on AWGN channel, it is not applicable for this equivalent trellis-collapsed

SCTCM because the LLR of inputs bits is exactly the same as that of the original encoder under the [1 0] channel and the pdf doesn't follow a symmetric Gaussian distribution. In other words, the trellis collapse is just another way of looking at the problem. By checking the equivalent trellis-collapsed inner codes, we found the following properties useful to reduce the population of candidate codes.

1. State reduction: The number of states of the collapsed inner code may become fewer than that of the original encoder because some registers simply never affect the transmitted symbol and are thus useless. For example, a 4-state, rate-2/3 inner code, $C_{i,5}$ (see Table 1.2), on the [0 1] channel, becomes equivalent to a trellis-collapsed 2-state, rate-4/3 code operating on the AWGN channel, as shown in Figure 1.4. Here equivalent means having exactly the same input-output behavior on the unerased symbols and the trellis collapsed equivalent encoder does not produce the symbols that will ultimately be erased. By the iterative decoding design criteria, constituent encoders with fewer states tend to yield lower pinch-off thresholds. This is especially important for the [0 1], [1 0] channels.
2. Non-transmitted input bits: After collapsing the inner code, some portion of the input bits to the inner code may not be transmitted at all. Figure 1.5 gives an example of an inner code whose [0 1] channel equivalent encoder has one non-transmitted bit per symbol. With this encoder, no extrinsic information for this non-transmitted bit can be extracted from the inner SISO. This structural defect is not desired and it results in high error floors (higher than BER $= 10^{-4}$).

In essence, we search for inner codes whose trellis-collapsed equivalent

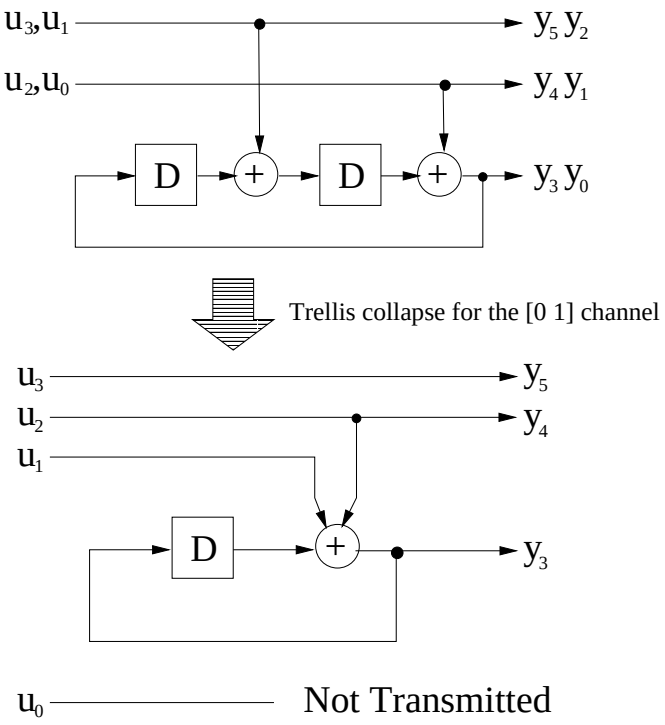


Figure 1.5: An example of non-transmitted bits after trellis collapse on the [1 0] channel. Note that no information of u_0 can be transmitted through the [1 0] channel for this encoder.

codes have state reduction and without non-transmitted bits. The qualified codes are then simulated to find the best code that has the lowest pinch-off threshold.

- Constellation labeling

  Constellation labeling affects the performance of SCTCM on both [1 1] and [1 0] channels. As ten Brink [15] shows, the systematic bits play an important role in SCTCM design. The EXIT chart analysis indicates that systematic inner codes have higher initial extrinsic information in their transfer curves than non-systematic codes and thus are more likely to have low pinch-off thresholds. Using a similar concept, our constellation labeling is selected such that the systematic bits of the inner code are best protected.

  Here we take two extreme cases of 8-PSK labelings as examples to illustrate this idea. Assume that we have a systematic rate-2/3 inner code with the two MSB as systematic bits. As shown in Figure 1.6, a Gray-labeled 8-PSK provides the best protection on the two systematic bits while natural labeling offers very poor protection. Figure 1.7 is the EXIT chart of an SCTCM with the above two 8-PSK labelings working on the AWGN with E_s/N_o=1.1 dB. The Gray-labeled inner TCM has already opened up the decoding tunnel, while the tunnel of the Natural-labeled inner TCM is still closed at this E_s/N_o. The latter will require the E_s/N_o=3.2 dB to open the tunnel.

  When the inner TCM consists of a rate-3/4 code mapped to 16-QAM, it is not possible as in 8-PSK to have maximal protection on each of the systematic bits because there are three input bits but the constellation is 2-dimensional. In this case, a Gray-labeled 16-QAM in raster order using hexadecimal [3, 1, 5, 7; 2, 0, 4, 6; a, 8, c, e; b, 9, d, f] provides the best

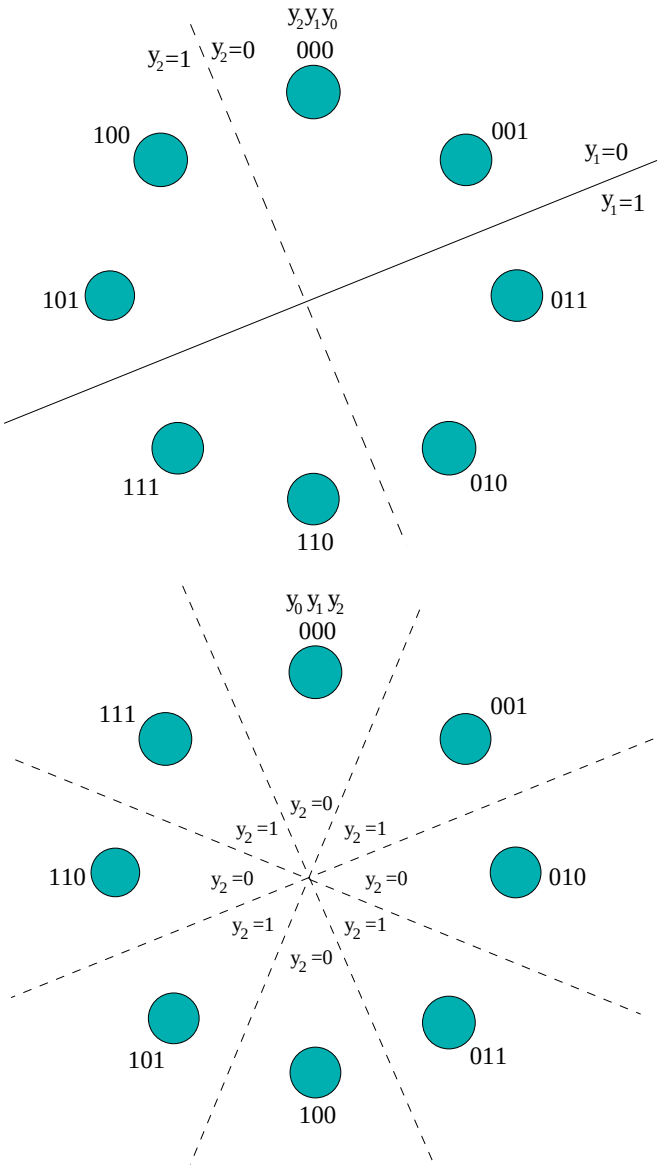


Figure 1.6: A Gray-labeled and a Natural-labeled 8-PSK constellations with y_2 and y_1 as systematic bits. Note that the Gray labeling provides maximum protection on systematic bits while for Natural labeling, the protection is the worst on y_2 and moderate on y_1.

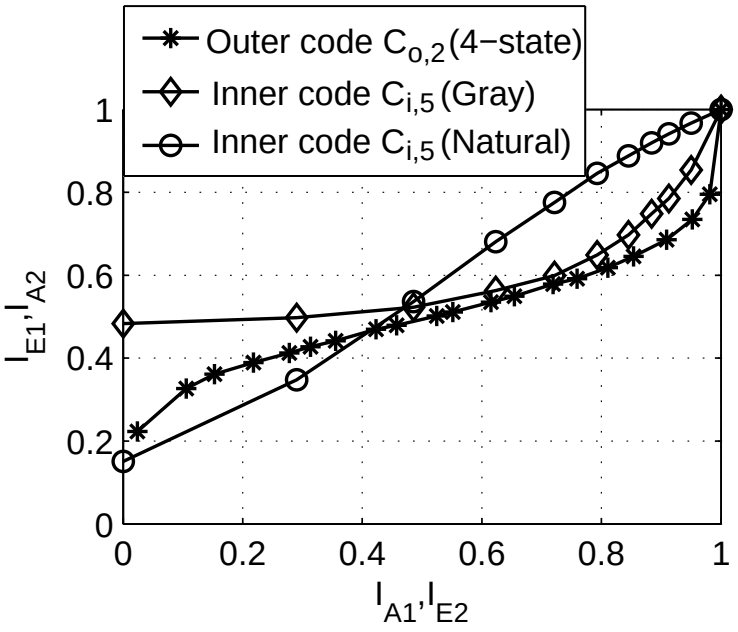


Figure 1.7: EXIT chart showing the effect of constellation labeling. The inner TCM with Gray-labeled 8-PSK has a significantly higher initial extrinsic information compared to the same inner TCM with Natural-labeled 8-PSK at the same $E_s/N_o = 1.1$ dB because of better protection of systematic bits. The Gray-labeled inner TCM has already opened up the decoding tunnel, while the tunnel of the Natural-labeled inner TCM is still closed at this E_s/N_o. The latter will require the $E_s/N_o = 3.2$ dB to open the tunnel.

possible protection of the systematic bits.

Table 1.2 lists the proposed inner convolutional codes with code rate-1/3, -2/3, and -3/4, and the corresponding design criteria used. The mapping is Gray-labeled 8PSK or 16QAM. The ML-optimal inner TCM has the largest d_i. As for the iterative-decoding optimal inner TCM, we perform an exhaustive search for inner TCMs whose trellis-collapsed equivalent codes have state reduction and without non-transmitted bits. The qualified candidate codes are then simulated to determine which one has the lowest threshold.

1.4.3 Interleaver design

The interleaver proposed is the extended spread interleaver [19] with a further constraint depending on the outer code and the periodic erasure pattern. Intuitively, if the intermediate bits, i.e., the output bits of C_o, that contain the information of a single input information bit are all interleaved to erased symbols, it becomes very difficult for the decoder to decode this input bit since only a small amount of information is preserved through the parity bits. So, the interleaver design aims to prevent the *all-erased* cases from happening for any of the input bits.

For example, let's design an interleaver for an SCTCM consisting of the rate-1/2 ($k_o = 1, n_o = 2$) $C_{o,2}$ from Table 1.2 and a rate-3/4 systematic C_i ($k_i = 3, n_i = 4$) with a 16-QAM constellation. The impulse response of $C_{o,2}$ is $h = [11\ 01\ 11\ 00 \cdots 00]$ indicating that five of the intermediate bits are associated with a single input bit. A bit-interleaver of length N_{IL}, is defined as $\{\pi(i) = j,\ i = 0, \cdots, N_{IL} - 1\}$ which interleaves the i^{th} interleaver input to the j^{th} interleaver output. Then the further constraint of the interleaver considering the periodic-2

erasure patterns, [1 0] and [0 1], is:

$$1 \leq \sum_{v \in S_h} \{ \lfloor \pi(n_o \cdot l + v)/k_i \rfloor (\text{modulo } 2) \} \leq |S_h| - 1 = 4, \ \forall \ l = 0, \cdots, N_i - 1. \quad (1.9)$$

where N_i is the input blocklength, $S_h = \{ w : h(w) = 1 \} = \{ 0, 1, 3, 4, 5 \}$, $n_o = 2$, and $k_i = 3$ in this example. Essentially, Eq. (1.9) guarantees that at least one of these five bits will not have its inner-code output-symbol erased.

Compared with spread interleavers without this further constraint, the new interleaver can improve the threshold of this SCTCM by about 0.05 bits of EMI (0.4 dB of E_s/N_o) on the [1 0] and [0 1] channels while maintaining almost the same performance on the [1 1] channel.

1.5 Example SCTCM Designs and Simulation Results

1.5.1 SCTCM Design of 0.5 bits per symbol

We used a rate-1/2 outer encoder and a rate-1/3 inner encoder mapping an 8-PSK constellation to provide 0.5 bits/symbol overall throughput. The inner TCM still has redundancy under period-2 erasures. An exhaustive search over the 8-state, rate-1/3 systematic feedback encoders using the trellis-collapse check yielded several inner encoders with reduced complexity and among these encoders, $C_{i,1}$ has the lowest threshold.

The proposed SCTCMs of 0.5 bits per symbol, SC-1, SC-2, and SC-3, are listed in Table 1.3 using constituent codes in Table 1.1 and Table 1.2. SC-1 uses a 4-state maximum free-distance outer code, $C_{o,2}$, with $C_{i,1}$. This scheme has an error floor around BER = 10^{-5} under the [1 0] channel. Two other codes, SC-2 and SC-3, were designed to have lower error floors and their BER curves are shown in Figure 1.8.

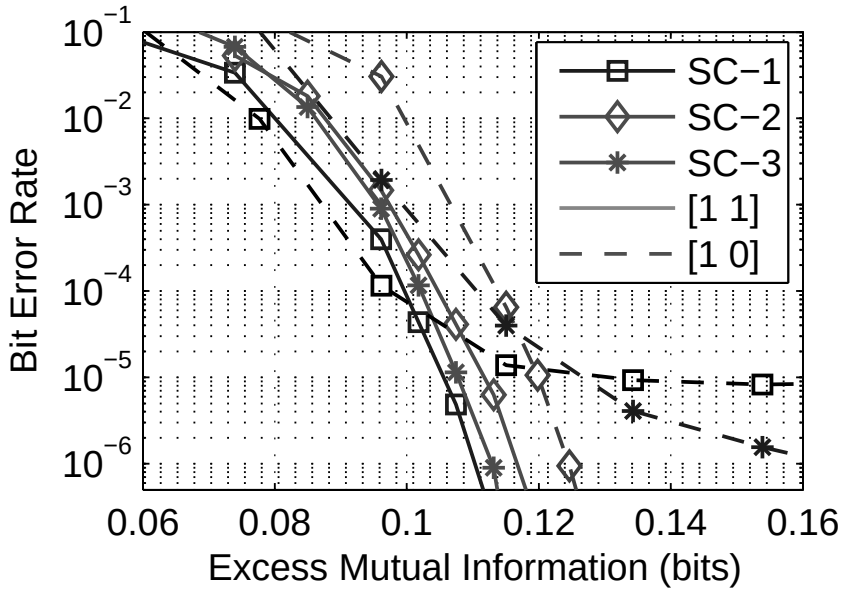


Figure 1.8: Simulations of BER versus EMI for 0.5 bits/symbol SCTCMs under the [1 1] and [1 0] period-2 PEC. SC- Block length=10,000 bits, 12 iterations.

One method to lower the error floor is to use an outer encoder with larger free distance at the expense of increased complexity. Using an 8-state maximal free-distance outer code, $C_{o,3}$, SC-3 lowered the error floor under periodic erasures to BER=10^{-6}.

SC-2 is another alternative scheme whose inner code is ML-optimal and doesn't have complexity reduction after trellis collapse. SC-2, which concatenates $C_{i,2}$ and $C_{o,2}$, is able to lower the error floor as expected because of its larger free distance in the inner code. The performance difference between the SCTCM designed by ML decoding criteria (SC-2) and iterative decoding criteria (SC-3) at BER=10^{-5} is quite small in this case. SC-2 requires the least average EMI of 0.116 bits at BER = 10^{-5} and SC-3 requires 0.117 bits. If a lower BER is required, SC-2 obviously is a better choice.

1.5.2 SCTCM Design of 1.0 bit per symbol

Our 1.0 bit per symbol SCTCM uses a rate-1/2 outer code and an inner TCM consisting of a rate-2/3 linear systematic recursive encoder with an 8-PSK constellation. With a rate-2/3 inner encoder, the inner TCM has *negative redundancy* and therefore zero minimum distance under period-2 erasures. However, it is still possible to find an inner code with nonzero *effective* free distance. ML decoding criteria suggest to choose the inner code with maximal effective free distance while iterative decoding criteria concentrate on SCTCM with low pinch-off threshold and allow the effective free distance to be zero. The proposed SCTCMs, SC-4 through SC-10, are listed in Table 1.3.

Figure 1.9 shows the BER performance of two SCTCM schemes (SC-4, SC-5) using the same inner TCM ($C_{i,5}$) with a Gray-labeled 8-PSK found by iterative decoding criteria. The difference here is the complexity of the outer code, SC-4

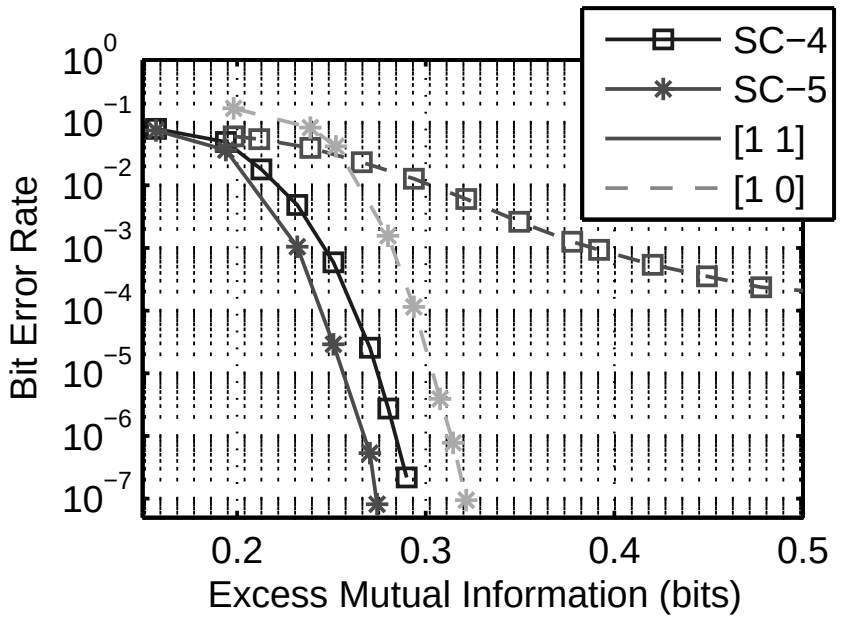


Figure 1.9: Simulations of BER versus EMI for two 1.0 bit/symbol SCTCMs which are only different in the complexity of the outer codes. SC-4 used a 2-state outer code while SC-5 uses a 4-state outer code. Blocklength=10,000 bits, 12 iterations at the decoder.

uses a 2-state outer code while SC-5 uses a 4-state outer code. The comparison shows that we need at least a 4-state C_o to deliver the desired error floor.

Figure 1.10 shows a set of simulations with a fixed outer code, $C_{o,2}$, and three different 4-state inner codes. SC-7, using a maximal-d_i (for both [1 1] and [1 0] channels) inner code, $C_{i,6}$. On the other hand, the inner codes of SC-5 and SC-6 both have reduced complexity without non-transmitted bits after the [0 1] channel trellis collapse. SC-5, SC-6 and SC-7 have similar performance on the [1 1] channel. On the [1 0] channel, SC-5 and SC-6, which are designed according to iterative decoding criteria, converge earlier than ML-optimal code, SC-7. SC-6 has the lowest threshold but its error floor is observed at BER = 10^{-6}. Among

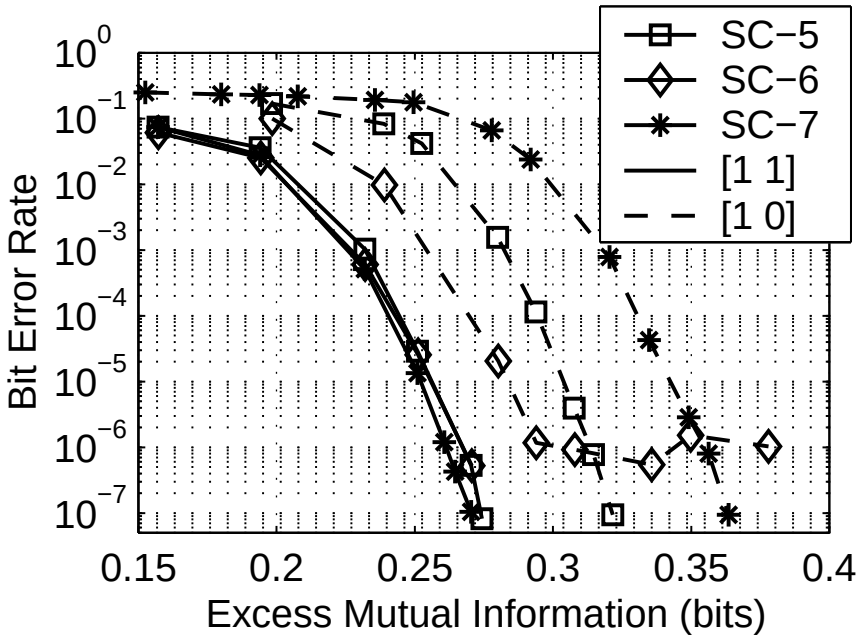


Figure 1.10: Simulations of BER versus EMI for three 1.0 bit/symbol SCTCMs with different 4-state inner codes. Solid lines for the [1 1] channel and dashed lines for the [1 0] channel. All three SCTCMs perform similarly on the [1 1] channel, but on the [1 0] channel, the tradeoff between low pinch-off thresholds (Iterative-designed SC-5,6) and low error floor (ML-designed SC-7) is observed. Blocklength=10,000 bits, 12 iterations.

the three codes, if the objective BER is 10^{-5}, SC-6 will be a better choice whose average EMI requirement is 0.270 bits. However, if an error floor lower than 10^{-6} is required, SC-5 can trade a little bit on pinch-off threshold for lower error floor which is not observed till 10^{-7}.

SC-9 and SC-10 use 8-state inner TCMs designed to be iterative decoding optimal and ML-optimal respectively. The BER performance are shown in Figure 1.11. Compared to the 4-state inner TCM schemes, SC-5, SC-6 and SC-7, the increased complexity did not help reduce the pinch-off thresholds via increased

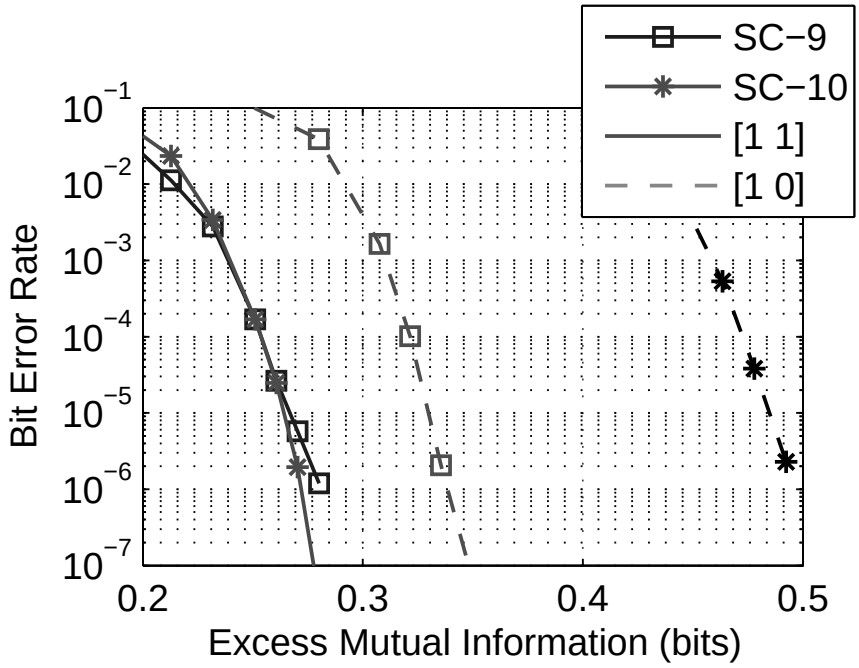


Figure 1.11: Simulations of BER versus Excess MI for two 1.0 bit/symbol SCTCMs with 8-state inner code. SC-9 has lowest pinch-off threshold while SC-10 has maximal inner effective free distance. Blocklength=10,000 bits, 12 iterations.

minimum distances. Divsalar [16] and ten Brink [15] also reported this kind of effect. Again, the iterative decoding optimal code outperforms ML-optimal code on the [1 0] channel (0.155 bits less EMI) without sacrificing on the [1 1] channel (0.003 bits more EMI).

As a result, the search for universal SCTCMs over compound periodic erasure channel should start from low complexity constituent encoders, applying the trellis-collapse checks, and avoiding high error floors. Because of the tradeoff between pinch-off threshold and error floor, the optimal code depends on the operating BER.

1.5.3 SCTCM Design of 1.5 bits per symbol

Using the same criteria as in previous searches, we designed 1.5 bits/symbol SCTCMs by concatenating a rate-1/2 outer code with an inner TCM consisting of a linear recursive systematic rate-3/4 inner code with a 16-QAM constellation. We see a very similar situation as in the 1.0 bit/symbol design. As shown in Figure 1.12, SC-11 and SC-12 designed by iterative decoding criteria outperformed ML-optimal SC-13. Note that SC-11 and SC-12 use 4-state inner codes while SC-13 must use an 8-state inner code to guarantee nonzero d_i. At BER $= 10^{-5}$, SC-11 requires 0.375 bits of average EMI and the difference between [1 1] and [1 0] channel performances is less than 0.01 bits. SC-12 performs slightly better than SC-11 but its error floor is observed at BER $= 10^{-7}$ on the [1 0] channel. SC-13 has similar performance as SC-11 and SC-12 on the [1 1] channel but degrades significantly on the [1 0] channel.

1.5.4 SCTCM on AWGN channel with inner code rate greater than One

A surprising discovery of this work is that an SCTCM which has zero effective free distance in the inner code can perform well as long as the overall rate is less than 1. For the negative redundancy inner TCM, not only is the free distance certainly zero, the effective free distance may also be zero. Such an SCTCM definitely violates the ML decoding design criteria, which seeks to maximize d_i. However, in some cases a high error floor is not observed which means that the overall free distance of the SCTCM is still nonzero and large enough. This makes sense because the uniform interleaver analysis of the ML criteria is only an average performance over all the possible interleavers and considers the worst case distance of the SCTCM by concatenating all the weight-2 input error events to-

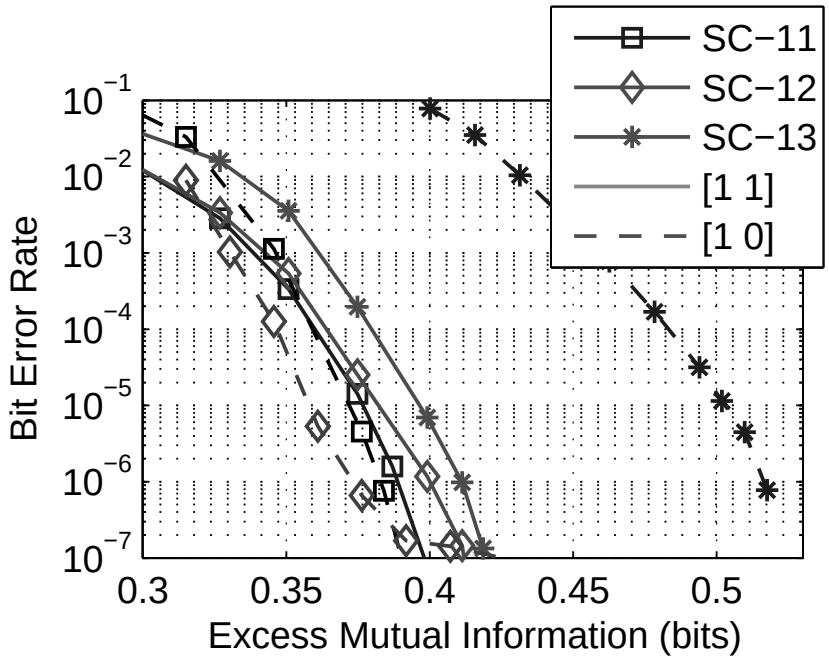


Figure 1.12: Simulations of BER versus EMI for three 1.5 bit/symbol SCTCMs. Solid lines for the [1 1] channel and dashed lines for the [1 0] channel. All three SCTCMs perform similarly on the [1 1] channel, but on the [1 0] channel, the iterative-optimal SCTCMs, SC-11 and SC-12, perform better than the ML-optimal SC-13. No error floor is observed for these SCTCMs until BER=10^{-7}. All simulations are run with blocklength of 10,000 bits and 12 iterations at the decoder.

gether. But the concatenated weight-2 error event may never happen with a suitable interleaver. The ML criterion provides some helpful insight to the design of STCTM but doesn't necessarily guarantee that SCTCMs that violate the design criterion are bad codes. For example, SC-5 under [1 0] channel can be seen as a rate-2/3 code with zero effective free distance (rate-1/2 $C_{o,2}$ concatenated with rate-4/3 C^i in Figure 1.4) mapping 8-PSK under AWGN with throughput equals 2.0 bits/symbol. We simulated the equivalent SCTCM on AWGN and the BER curve is, of course, exactly the same as the original SCTCM on the [1 0] channel. Thus, this is an example of an SCTCM performing well despite having an inner code with zero d_i.

Let's compare this SCTCM whose inner code rate has negative redundancy and zero d_i with a conventionally designed SCTCM in the literature. Divsalar [20] proposed an SCTCM with rate-1 inner code on 8-PSK which performs better than our SCTCM by less than half a dB at the BER = 10^{-5} with the input blocklength of 10,000 bits. Although our punctured SCTCM performs worse, it is surprising that the difference is quite small which means that an SCTCM with negative redundancy and even zero d_i in the inner code can still be a reasonably good code. However, it is worth noting that it takes careful interleaver design and constellation labeling selection for those punctured SCTCMs to perform closely to conventional SCTCMs. The remaining performance degradation after all the design efforts could be due to the rate allocation of a rate-1 inner code is better than a rate-4/3 inner code. However, a more likely reason is that the erasure channel forces the erased bits and the unerased bits to have different reliabilities. So, in the equivalent collapsed encoder, the LLR's of the bits corresponding to the erased bits in the original scheme are inevitably lower. The unerased bits must pass their information to the erased bits through the code structures so that eventually all the bits can be correctly decoded. This is the inherent defect

of punctured codes compared to un-punctured codes.

1.5.5 Periodic fading channel performance

As mentioned in the introduction, we are interested in how a universal SCTCM designed for the PEC performs on the PFC normalized as [1 **q**] channels where $0 \leq \mathbf{q} \leq 1$. Figure 1.13 shows the EMI requirements of SC-2, SC-5, and SC-11 (the best code at each throughput with an error floor $< 10^{-7}$) on [1 **q**] channels for **q**=0.2, 0.4, 0.6 and 0.8. For all three codes, the EMI requirements are approximately bounded by that of [1 1] and [1 0] channels. This result is consistent with [14] section IV and [10] theorem 3. Also, the MI requirement is relatively flat over **q** for all three cases indicating consistently good performance for all channels. Therefore, these universal SCTCMs designed for periodic erasures are also universal for periodic fading.

LDPC codes are found to have universal property on the period-2 PFC without much special design effort [1]. Comparing these two capacity-approaching coding schemes, a well-designed universal SCTCM for period-2 channels can yield comparable or even slightly smaller EMI requirements than LDPC codes. As shown in Figure 1.15, at 1.0 bit/symbol, the average EMI requirement between SC-5 and a comparable LDPC code designed using approaches proposed in [1] differ by only 0.006 bits of EMI at BER $= 10^{-3}$.

However, when the period of channel increases, it becomes more and more difficult to design a universal SCTCM because the number of channels to be considered grows exponentially with the length of the period and it is prohibitively complex to apply the proposed design rules on each of the channels. To compare with the LDPC code on the four period-256 fading channels in [1] (See Figure 5 and 6), we designed a period-2 universal SCTCM, SC-14, transmitting at rate-1/3

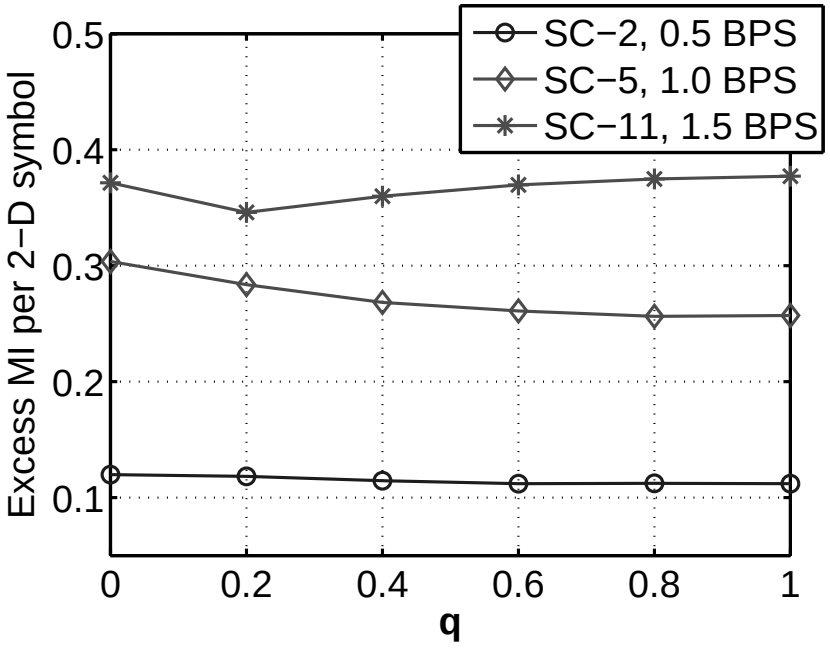


Figure 1.13: Periodic fading channel performances of SC-2, SC-5, and SC-11. $\mathbf{q}$ is the attenuation factor in the normalized period-2 fading channel denoted as [1 $\mathbf{q}$] channel. EMI is computed for target BER of 10^{-5}.

on QPSK. SC-14 does not have the universal performance on these period-256 channels because the short constraint length of the inner code makes it vulnerable to long deep fades. For the first three channels, the faster the channel varies, the less likely are these problematic long deep fades. So, the performance on a channel with more ISI taps is better. (See Figure 1.14, the dotted lines.) As expected, the SCTCM cannot work at all if 125 consecutive tones are erased. However, LDPC code works well in this severe case. If we use a random channel interleaver to break up the long deep fades, SC-14 then performs uniformly well on the first three channels (Figure 1.14, the solid lines) but still loses to the LDPC code by about 0.11 bits of EMI. As for the fourth channel, random channel interleavers result in performance difference of more than 10dB depending on how well the deep fades are separated. The best performance is given by a 2-by-128 block interleaver resulting in a channel with every other symbol erased which then becomes similar to a period-2 channel. Since our SCTCM is designed to have optimal performance under the [1 0] channel, it is not surprising that the SCTCM slightly outperforms the LDPC code by 0.04 bits of EMI for this particular interleaver. However, this is impractical because for a random fading channel, it is impossible to design a "proper" channel interleaver for the SCTCM. Hence, LDPC codes seem to be a more reasonable choice than turbo codes to achieve universality for periodic fading with longer periods.

1.6 Conclusions

The design of universal SCTCMs in this chapter aims to maintain turbo performance under periodic erasures with as little compromise as possible on the AWGN performance. At BER = 10^{-5}, the proposed SCTCMs transmitting at 0.5 bits/symbol, 1 bit/symbol and 1.5 bits/symbol require an average EMI of

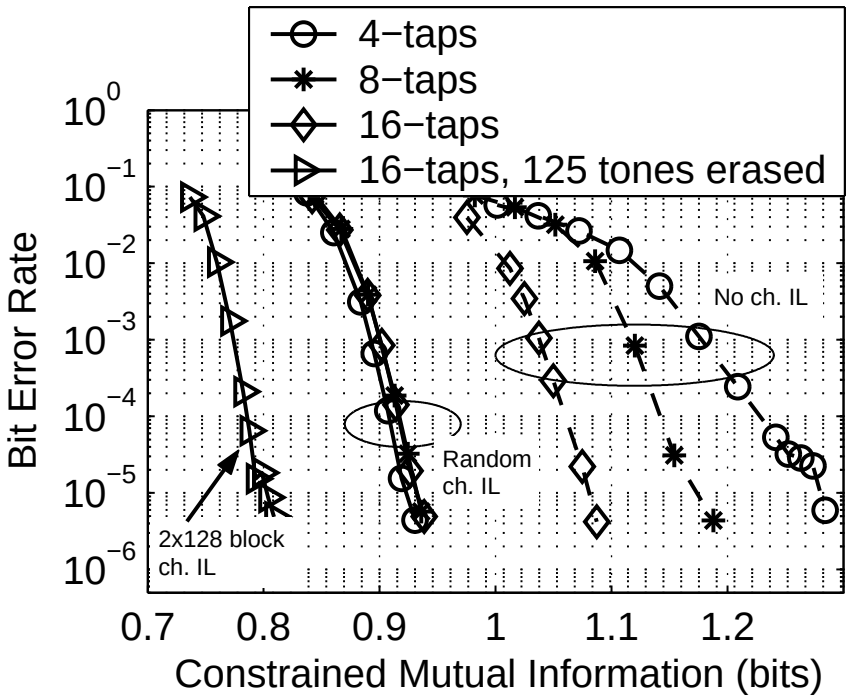


Figure 1.14: The performance of SC-14, the universal SCTCM on period-2 channels, on the four period-256 channels (See [1], Figure 5). Dotted lines are simulation results without channel interleavers. Note that the code doesn't work at all on the fourth channel where 125 consecutive erasures happen. Solid lines show that the performances are improved when channel interleavers are used. These results can be compared directly to LDPC code results (See [1], Figure 6).

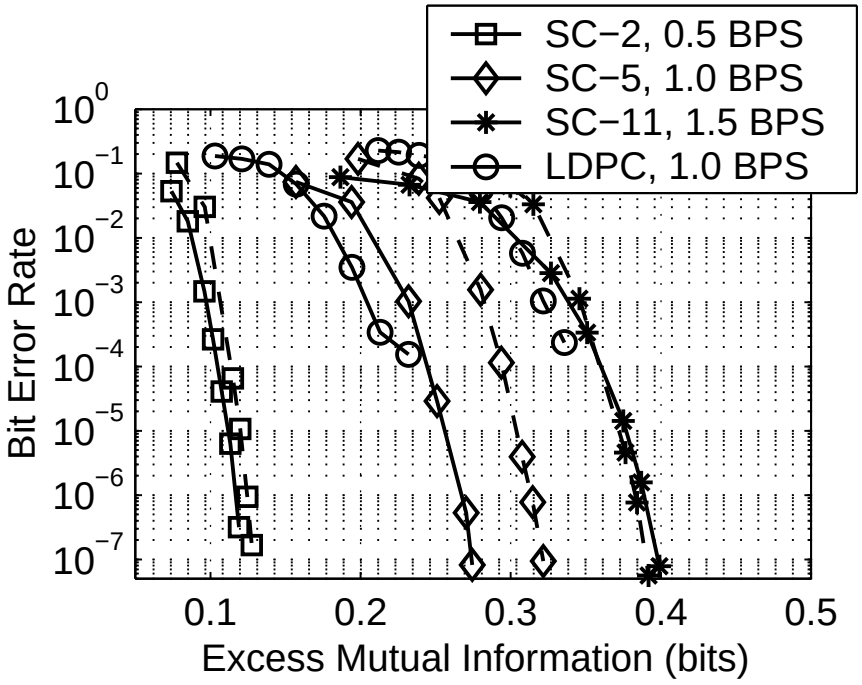


Figure 1.15: The EMI requirement for the best codes proposed whose error floors are lower than 10^{-7}. SC-2, SC-5 and SC-11 transmit at 0.5, 1.0, and 1.5 bit/symbol respectively. A 1.0 bit/symbol LDPC code is also shown for comparison. Solid lines for the [1 1] channel and dashed lines for the [1 0] channel.

0.116, 0.270, 0.370 bits respectively on the compound period-2 periodic erasure channels.

Typical 1 bit/symbol SCTCMs [21] at similar complexity require 0.350 bits of EMI at BER $= 10^{-5}$ under AWGN. The proposed 1 bits/symbol SCTCM requires 0.094 and 0.066 bits less EMI on the [1 1] and [1 0] channels respectively. Fig. 1.15 shows the EMI requirement for the best three codes proposed whose error floors are lower than 10^{-7}. Note that SC-3, SC-6 and SC-12 are also competitive SCTCMs but with higher error floors.

An interesting outcome of our experiments is that it is possible to work under periodic erasures with an inner TCM that has negative redundancy when every other symbol is erased even if the effective free Euclidean distance under periodic erasures is zero. Although zero effective free distances on the inner code may result in very high error floors, the trellis-collapse checks help avoid the defect by disallowing non-transmitted bits.

The maximum-likelihood decoding criterion and iterative decoding criterion yield SCTCMs with very different performances. The ML-optimal SCTCMs have larger effective free distances and thus lower error floors but their pinch-off thresholds are usually higher. On the other hand, SCTCMs which are designed solely based on low pinch-off thresholds may have higher error floors. It is still an open issue to determine the position of error floor of a specific SCTCM. In this chapter, we proposed criteria to avoid very high error floors. The trade-off between low pinch-off thresholds and low error floors is more evident under periodic erasures when most of the redundancy is removed.

The universal SCTCMs we designed for periodic-2 erasures are also universal over periodic-2 fading channels. As a result of careful design, SCTCMs can have comparable performance to LDPC codes on period-2 channels. But when

the period of the channel goes higher, it becomes difficult to design a universal SCTCM and a channel interleaver is required. An SCTCM can perform as good as an LDPC code on a period-256 channel only if the channel interleaver is properly designed according to the fading pattern. On the other hand, LDPC codes stay universal without much special design effort or channel interleavers, which makes LDPC codes a better choice than turbo codes for universal channel coding on periodic fading channels with large periods.

Table 1.1: Outer convolutional encoders of rate-$\frac{k}{n}$ with memory ν

Encoder	Crit.	k	n	ν	$G(D)$
$C_{o,1}$	ML	1	2	1	$[1+D \quad D]$
$C_{o,2}$	ML	1	2	2	$[1+D^2 \quad 1+D+D^2]$
$C_{o,3}$	ML	1	2	3	$[1+D^2+D^3 \quad 1+D+D^2+D^3]$
$C_{o,4}$	ML	2	3	2	$\begin{bmatrix} 1+D & D & 1+D \\ D & 1 & 1 \end{bmatrix}$

Table 1.2: Inner convolutional encoders of rate-$\frac{k}{n}$ with memory ν

Encoder	Crit.	k	n	ν	$G(D)$
$C_{i,1}$	I	1	3	3	$[1 \quad \frac{D}{1+D^2} \quad 1+D]$
$C_{i,2}$	ML	1	3	3	$[1 \quad \frac{D^2}{1+D+D^2} \quad \frac{1}{1+D}]$
$C_{i,3}$	ML	2	3	2	$\begin{bmatrix} 1 & 0 & \frac{1+D^2}{1+D+D^2} \\ 0 & 1 & \frac{1+D}{1+D+D^2} \end{bmatrix}$
$C_{i,4}$	I	2	3	2	$\begin{bmatrix} 1 & 0 & \frac{D}{1+D^2} \\ 0 & 1 & \frac{D}{1+D^2} \end{bmatrix}$
$C_{i,5}$	I	2	3	2	$\begin{bmatrix} 1 & 0 & \frac{1+D}{1+D^2} \\ 0 & 1 & \frac{D}{1+D^2} \end{bmatrix}$
$C_{i,6}$	ML	2	3	2	$\begin{bmatrix} 1 & 0 & \frac{D}{1+D+D^2} \\ 0 & 1 & \frac{1+D}{1+D+D^2} \end{bmatrix}$
$C_{i,7}$	I	2	3	3	$\begin{bmatrix} 1 & 0 & \frac{1}{1+D^3} \\ 0 & 1 & \frac{D}{1+D^3} \end{bmatrix}$
$C_{i,8}$	ML	2	3	3	$\begin{bmatrix} 1 & 0 & \frac{D}{1+D^2+D^3} \\ 0 & 1 & \frac{1+D+D^2}{1+D^2+D^3} \end{bmatrix}$
$C_{i,9}$	I	3	4	2	$\begin{bmatrix} 1 & 0 & 0 & \frac{D}{1+D^2} \\ 0 & 1 & 0 & \frac{D}{1+D^2} \\ 0 & 0 & 1 & \frac{1+D}{1+D^2} \end{bmatrix}$
$C_{i,10}$	I	3	4	2	$\begin{bmatrix} 1 & 0 & 0 & \frac{D}{1+D^2} \\ 0 & 1 & 0 & \frac{D}{1+D^2} \\ 0 & 0 & 1 & \frac{D}{1+D^2} \end{bmatrix}$
$C_{i,11}$	ML	3	4	3	$\begin{bmatrix} 1 & 0 & 0 & \frac{1}{1+D+D^3} \\ 0 & 1 & 0 & \frac{D^2}{1+D+D^3} \\ 0 & 0 & 1 & \frac{1+D+D^2}{1+D+D^3} \end{bmatrix}$
$C_{i,12}$	I	1	2	1	$[1 \quad \frac{1}{1+D}]$

Table 1.3: SCTCM schemes SC-1 to SC-13 transmitting at 0.5, 1.0 and 1.5 bits/symbol resulted from the searches by either ML decoding criteria or iterative decoding criteria. The operating E_s/N_o are measured at BER $= 10^{-5}$ on the [1 1] and [1 0] channels to compute the corresponding EMI. Simulations of BERs are run down to 10^{-7}. None of these SCTCMs has error floor observed on the [1 1] channel, so only error floors on the [1 0] channel are reported. Note that SC-8 is originally a BPSK, AWGN code designed according to ML criteria [3] but here it is mapped to an 8-PSK as an example of non-universal SCTCMs. SC-1 to SC-13 have input blocklength equals 10,000. SC-14 is designed to compare with the LDPC code in [1] which is rate-1/3 mapping QPSK with block length 5,000.

SC	C_o	C_i	ν_o	ν_i	R	C_i Crit.	EMI			E_s/N_o(dB)		Err. floor ([1 0])
							[1 1]	[1 0]	Avg.	[1 1]	[1 0]	
1	2	1	2	3	0.5	I	0.106	0.120	0.113	-2.82	1.35	10^{-5}
2	2	2	2	3	0.5	ML	0.112	0.120	0.116	-2.76	1.35	$< 10^{-7}$
3	3	1	3	3	0.5	I	0.107	0.126	0.117	-2.80	1.42	10^{-6}
4	1	5	1	2	1.0	I	0.274	N/A	N/A	1.52	N/A	10^{-3}
5	2	5	2	2	1.0	I	0.256	0.304	0.280	1.43	7.07	$< 10^{-7}$
6	2	4	2	2	1.0	I	0.256	0.284	0.270	1.42	6.92	10^{-6}
7	2	6	2	2	1.0	ML	0.252	0.442	0.347	1.41	8.05	$< 10^{-7}$
8	2	3	2	2	1.0	ML*	0.244	0.486	0.365	1.36	8.35	$< 10^{-7}$
9	2	7	2	3	1.0	I	0.267	0.330	0.299	1.48	7.26	$< 10^{-7}$
10	2	8	2	3	1.0	ML	0.264	0.485	0.375	1.47	8.35	$< 10^{-7}$
11	2	9	2	2	1.5	I	0.377	0.372	0.375	4.27	10.93	$< 10^{-7}$
12	2	10	2	2	1.5	I	0.382	0358	0.370	4.29	10.84	10^{-7}
13	2	11	2	3	1.5	ML	0.396	0.603	0.500	4.35	12.42	$< 10^{-7}$
14	4	12	2	1	2/3	I	0.176	0.258	0.217	3.06	6.98	10^{-6}

CHAPTER 2

Universal Serially Concatenated Trellis Coded Modulation for Space-Time Channels

2.1 Introduction

As mentioned in Chapter 1, design of channel codes often focuses on the optimization of performance on a specific channel such as the additive white Gaussian noise (AWGN) channel. Powerful error-correcting codes such as turbo codes [6] and Low-Density Parity-Check (LDPC) codes [7] have been shown to operate within a dB of the Shannon limit on the AWGN channel. Motivated by Root and Varaiya's compound channel coding theory [8], we take a different approach to design universal channel codes that perform consistently on not just one channel but a family of channels. Designing criteria of universal SCTCMs for the periodic erasure channels (PEC) and periodic fading channels (PFC) are discussed in Chapter 1.

In this chapter, we extend results in Chapter 1 to design universal SCTCMs for the space-time channel or the Multiple-Input Multiple-Output (MIMO) channels

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}, \tag{2.1}$$

where $\mathbf{H}$ denotes the $N_r \times N_t$ channel matrix, $\mathbf{x}$ is the $N_t \times 1$ vector of transmitted symbols, one for each transmit antenna, $\mathbf{n} \sim \mathcal{N}(0, N_o I_{N_r})$ denotes the additive white Gaussian noise vector, and $\mathbf{y}$ is the $N_r \times 1$ vector of received symbols. N_t

and N_r are the number of transmit antennas and receive antennas respectively in the MIMO system.

Numerous turbo TCMs [22][23] [24] [25] have been designed to optimize average performance for space-time Rayleigh fading channels. Typically, recursive space-time trellis codes that satisfy the slow-fading criteria are used as constituent codes. These codes perform well on the intended channel or distribution. However, the performance can degrade significantly over some specific channel realizations.

In [26, 27], Zheng et al. examine the trade-off between diversity and multiplexing in MIMO systems. This trade-off turns out to be related to universal behavior. In [28], Tivildar and Viswanath give a precise characterization of approximately universal codes. These universal codes are also universally optimal in their trade-off between diversity and multiplexing. Tse and Viswanath provide an excellent overview of universal codes and their role in the diversity-multiplexing tradeoff in [29].

The proposed space-time SCTCM scheme uses an SCTCM designed for the PFC and de-multiplexes the output symbols into N_t symbol streams (as designed in [11, 30]). With this de-multiplexing scheme, the PFC is equivalent to diagonal MIMO channels and thus is as a subset of the MIMO channels. The PFC-universal SCTCMs designed in Chapter 1 show their universality over eigenvalue skew. However, nearly singular channels still display a variation due to the particular eigenvectors. The performance difference over eigenvectors can be largely mitigated, but not completely eliminated, by a time-varying linear transformation (TVLT) technique introduced in [4]. The current scheme employs a generalized form of TVLT, which uniformly sweeps three phase parameters over $[0, 2\pi)$.

Using the universal SCTCM of 0.5, 1.0 and 1.5 bits per symbol for the PFC, we

propose SCTCMs which transmits 1, 2, and 3 bits per channel use with consistent performance on the 2×2 space-time channel.

The rest of the chapter is organized as follows. Section 2.2 defines the channel model and the parameters used to sample the channel space. Section 2.3 presents the MIMO-SCTCM system and the corresponding SCTCMs used. Section 2.4 analyzes the TVLT technique. Section 2.5 presents the consistent excess mutual information (EMI) requirement of universal SCTCMs and their performance on the quasi-static Rayleigh fading channel. Section 2.6 concludes the chapter.

2.2 The Space-Time Channels

For the space-time channel in Eq. (2.1), the MI between the input vector $\mathbf{x}$ and output vector $\mathbf{y}$ is

$$\begin{aligned} \mathrm{MI}\left(\mathbf{H}, E_s\right) &= \log_2 \ \det\left(I_{N_r} + \frac{E_s}{N_o}\mathbf{H}\mathbf{H}^\dagger\right) && (2.2) \\ &= \sum_{i=1}^{\min(N_t,N_r)} \log_2\left(1 + \frac{E_s}{N_o}\lambda_i\right). && (2.3) \end{aligned}$$

where E_s is the energy per symbol per transmit antenna, $\mathbf{H}^\dagger$ is the Hermitian matrix of $\mathbf{H}$ and $\lambda_1, \lambda_2, \cdots, \lambda_{\min(N_t,N_r)}$ are the eigenvalues of $\mathbf{H}\mathbf{H}^\dagger$. From this point on, we will assume that $N_t = N_r = N$. For $N_t \neq N_r$, the same analysis applies with $N = \min(N_r, N_t)$ since the smallest $|N_r - N_t|$ eigenvalues of $\mathbf{H}\mathbf{H}^\dagger$ will be zero.

The search of universal SCTCM in this chapter will focus on the 2×2 matrix channels. Assume that $\lambda_1 \geq \lambda_2$ and define the eigenvalue skew $\kappa = \frac{\lambda_2}{\lambda_1}$, then the MI is given by

$$\mathrm{MI}\left(\mathbf{H}, E_s\right) = \log_2\left(1 + \frac{E_s}{N_o}\lambda_1\right)\left(1 + \frac{E_s}{N_o}\kappa\lambda_1\right), \qquad (2.4)$$

To verify the universality of the proposed codes, codes are simulated over all possible realizations of the channel ensemble. Since the 2×2 space-time channel has infinite channel realizations, we use a fine sampling of the channel space under the assumption of continuity.

The 2×2 space-time channel is parameterized as follows. First, by singular value decomposition (SVD), $\mathbf{H}$ can be written as

$$\mathbf{H} = U \Lambda V^{\dagger}, \tag{2.5}$$

where U and V are unitary matrices. Using the following notations:

$$\Lambda \triangleq \begin{bmatrix} \sqrt{\lambda_1} & 0 \\ 0 & \sqrt{\lambda_2} \end{bmatrix}, \; M(\psi) \triangleq \begin{bmatrix} \cos\psi & \sin\psi \\ -\sin\psi & \cos\psi \end{bmatrix}, \; D(\psi) \triangleq \begin{bmatrix} 1 & 0 \\ 0 & e^{j\psi} \end{bmatrix}, \tag{2.6}$$

$V^{\dagger}$ can be written as

$$V^{\dagger} = e^{j\mu} D(\omega) M(\phi) D(\theta). \tag{2.7}$$

At the receiver, we multiply on the left by a unitary matrix

$$P = e^{-j\mu} D(-\omega) U^{\dagger} \tag{2.8}$$

so that the equivalent channel becomes

$$\begin{aligned} \mathbf{H}_{eq} &= \Lambda M(\phi) D(\theta) \\ &= \begin{bmatrix} \sqrt{\frac{1}{1+\kappa}} & 0 \\ 0 & \sqrt{\frac{\kappa}{1+\kappa}} \end{bmatrix} \begin{bmatrix} \cos\phi & \sin\phi \cdot e^{j\theta} \\ -\sin\phi & \cos\phi \cdot e^{j\theta} \end{bmatrix}. \end{aligned} \tag{2.9}$$

Note that $\kappa = \frac{\lambda_2}{\lambda_1}$ and $\lambda_1 + \lambda_2$ is normalized to 1. We will use Eq. (2.9) as the channel model to evaluate the performance of SCTCMs on the 2×2 channel space by sampling over κ, ϕ and θ.

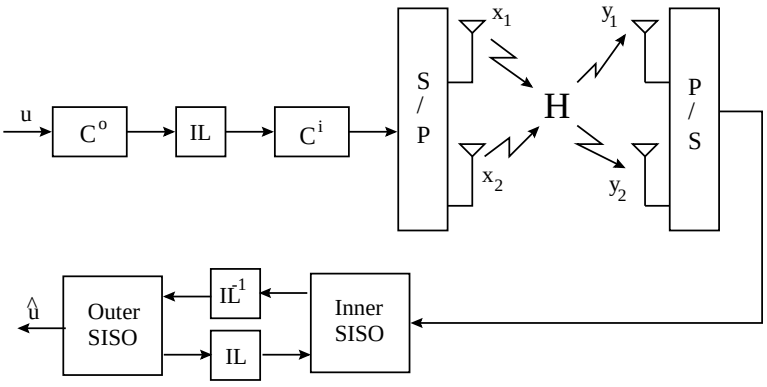


Figure 2.1: The de-multiplexed SCTCM MIMO scheme

2.3 Space-Time SCTCM Scheme

The proposed space-time SCTCM scheme, see Figure 2.1, consists of a rate-R^o outer convolutional encoder, C^o, an interleaver, a rate-R^i inner convolutional encoder, C^i, and a two-dimensional 2^n-point constellation mapper. The symbols are de-multiplexed into N_t symbol streams. If TVLT is used, the N_t-symbol vector is multiplied by a time-varying N_t-by-N_t unitary matrix before transmission. Therefore, the throughput of the overall scheme is $nR^oR^iN_t$ bits per transmission. On the decoder side, it is assumed that the decoder has perfect channel state information and the decoder also knows exactly the TVLT unitary matrices. Therefore, the LLR of the symbols can be calculated and passed to the turbo decoder. Note that the received symbols form a $N_r \times 1$ vector and each received symbol is a superposition of N_t transmitted symbols. So, the inner SISO is based on the *collapsed trellis* which combines N_t trellis stages together into a super-trellis.

In Chapter 1, universal SCTCMs were designed for the period-2 fading channels with fading pattern [1 $\mathbf{q}$] where $\mathbf{q}$ is a real number and $0 \leq \mathbf{q} \leq 1$. Note that a 1-dimensional SCTCM over the scalar [1 $\mathbf{q}$] period-2 fading channels is equiv-

Table 2.1: Universal SCTCMs for period-2 PFC proposed in Chapter 1 transmitting 0.5, 1.0 and 1.5 bits per symbol.

SC	C^o	C^i	Constellation	BPS
2	$[1+D^2\ 1+\mathrm{D}+D^2]$	$[1\ \frac{D^2}{1+D+D^2}\ 1+D]$	8PSK	0.5
5	$[1+D^2\ 1+\mathrm{D}+D^2]$	$\begin{bmatrix} 1 & 0 & \frac{D}{1+D^2} \\ 0 & 1 & \frac{1+D}{1+D^2} \end{bmatrix}$	8PSK	1.0
11	$[1+D^2\ 1+\mathrm{D}+D^2]$	$\begin{bmatrix} 1 & 0 & 0 & \frac{D}{1+D^2} \\ 0 & 1 & 0 & \frac{D}{1+D^2} \\ 0 & 0 & 1 & \frac{1+D}{1+D^2} \end{bmatrix}$	16QAM	1.5

alent to the proposed de-multiplexed space-time SCTCM system over the 2×2 matrix channels in (2.9) when $\kappa = \mathbf{q}^2$ and $\phi = \theta = 0$, i.e., the diagonal channels. The performance of a diagonal channel is usually the best one among the set of channels with the same eigenvalue skew. The universal SCTCMs over the $[1\ \mathbf{q}]$ channel are good starting point since they deliver universal performances over κ, but there is still the issue of non-diagonal channels.

The code design criteria under maximum-likelihood (ML) decoding and iterative decoding are proposed in Section 1.3 and 1.4. Using these criteria, the best SCTCMs found for 0.5, 1.0 and 1.5 bits per symbol with BER error floors lower than 10^{-7} are listed in Table 2.1. The same codes will be used in the corresponding 1.0, 2.0 and 3.0 bits per transmission space-time SCTCM systems.

2.4 Time-Varying Linear Transformation (TVLT)

The 2×2 space-time channel is parameterized using κ and the channel angles ϕ and θ as shown in Eq. (2.9). The angle ϕ determines the amount of interference between the two antennas. When $|\cos(\phi)|$=1 or 0, the channel is diagonal with

no interference. On the other hand, the interference is the largest when $|\cos\phi| = |\sin\phi| = \frac{1}{\sqrt{2}}$. The other parameter, θ, represents the phase difference between the constellations of the two antennas. Our experiments showed that although the MI is only a function of κ, different ϕ and θ can result in very different performances for SCTCM especially when κ is close to zero (the singular channel case).

We need to characterize EMI as a function of ϕ and θ but this is computationally difficult. Similar to the constellation labeling experiments in Sec. 1.4, we propose an approximate method to predict the EMI performance by observing the correlation between the EMI and the initial extrinsic information (IEI). The higher the IEI of the inner SISO is, the more likely the whole inner-SISO EXIT curve stays above the outer-SISO EXIT curve and thus lower convergence threshold. Note that the IEI can be easily calculated by numerically computing the parallel independent decoding capacity of the systematic bits of the inner code.

Figure 2.2 shows the EMI requirement and the corresponding inner-SISO IEI of SC-11 at $E_b/N_o = 6.35$ dB as a function of ϕ where $\theta = \kappa = 0$. It is clear that the EMI requirement and the IEI are closely related. Based on the above observations, achieving a uniform IEI over ϕ and θ is way to enhance consistent EMI behavior and thereby approach universal performance. We tested three very different approaches to achieve a uniform IEI over ϕ and θ.

Our first idea was to design a combination of constellation and labeling that delivers a more uniform IEI over ϕ and θ than the conventional schemes using PSK/QAM and Gray labeling. Different constellation shapes were tested. However, their performances were still as dependent on ϕ and θ as those of the PSK/QAM constellations. For example, we tried three different 16-point

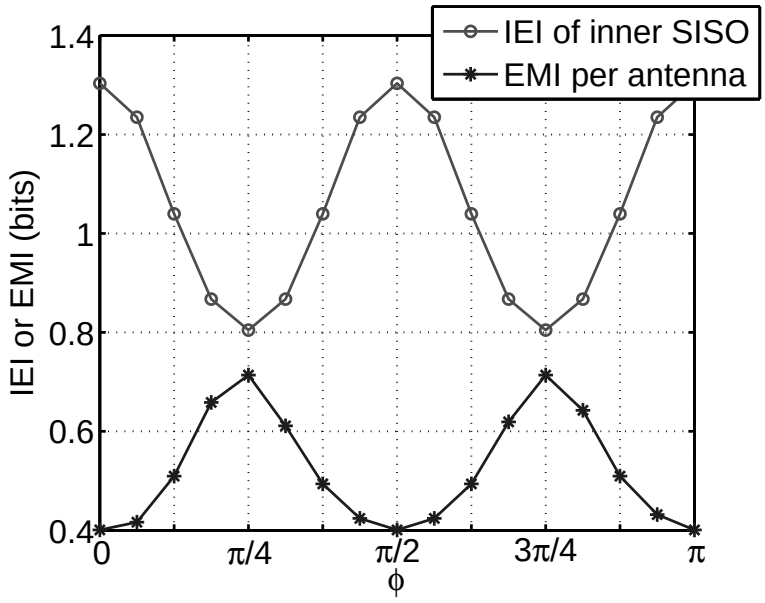


Figure 2.2: The inner-SISO IEI at $E_b/N_o = 6.35$ dB and corresponding EMI per antenna for SC-11 as a function of ϕ when $\kappa = \theta = 0$. It is observed that IEI is highly correlated with EMI and thus can be used as a tool to facilitate the code search and to understand and approximate the TVLT performance.

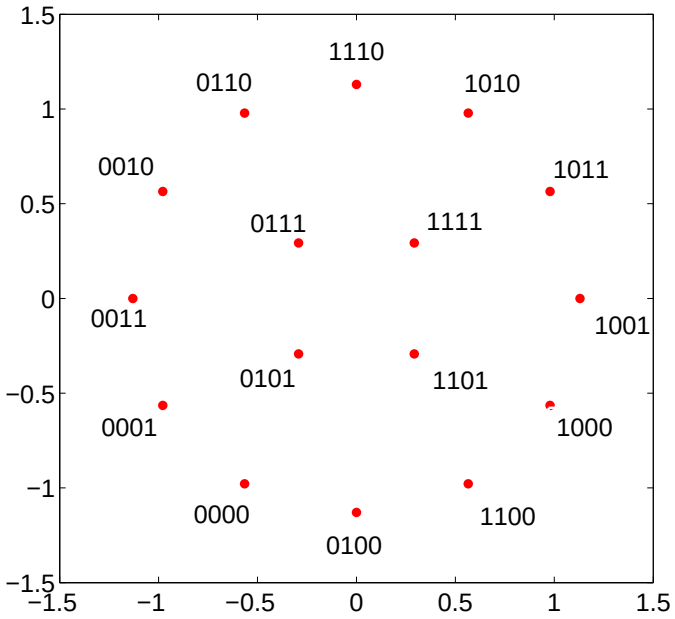


Figure 2.3: (12,4)-PSK constellation and labeling.

constellations including Gray-labeled 16QAM, (12,4)-PSK (see Figure 2.3), and Gray-labeled 16PSK. Figure 2.4 shows the IEI of SC-11 using these constellations as a function of ϕ when $\kappa = 0$ and $\theta = 0$. In all three cases, the IEI depends on ϕ in a similar way that the IEI has maximum when there is no interference and has minimum when the interference is the largest. Therefore, the shape of the constellation only has marginal effect on the IEI, and the universality of the SCTCM is not improved by changing the shape of the constellation.

As for the labeling, no other labeling is found to have the IEI that dominates the IEI of the Gray labeling. For example, Figure 2.5 shows the IEI for the 16QAM constellation using the Gray labeling, natural labeling, mixed labeling and set partition labeling as shown in Figure 2.6.

The Gray labeling dominates the other labelings in this example because it

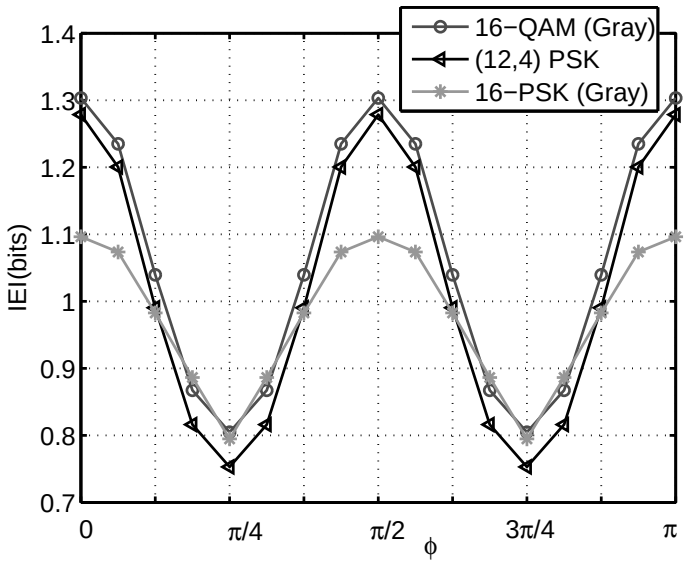


Figure 2.4: IEI of three different 16-point constellation including Gray-labeled 16-QAM, (12,4)-PSK and Gray-labeled 16-PSK as a function of ϕ when $\kappa = 0$ and $\theta = 0$.

has the best protection on the systematic bits. It is possible though to design a labeling which yields higher IEI for certain (ϕ, θ) angles, but at the same time, it also decreases the IEI at other angles. As a result, labelings other than the Gray labeling yield a less uniform IEI and an equal or inferior worst-case performance.

Motivated by [31], our second idea was to make the two constellations of the two antennas correlated. Instead of mapping the n coded bits to a 2^n-point 2D constellation, the general form of a correlated constellation considers the two transmitted antennas together and uses $2n$ consecutive bits to map a 2^{2n}-point 4D constellation (2D per antenna). However, the performance becomes worse because increasing the number of constellation points actually decreases the IEI.

The third idea, which fortunately does work, uses a time-varying linear transformation (TVLT) to rotate the channel to different angles, hoping that the

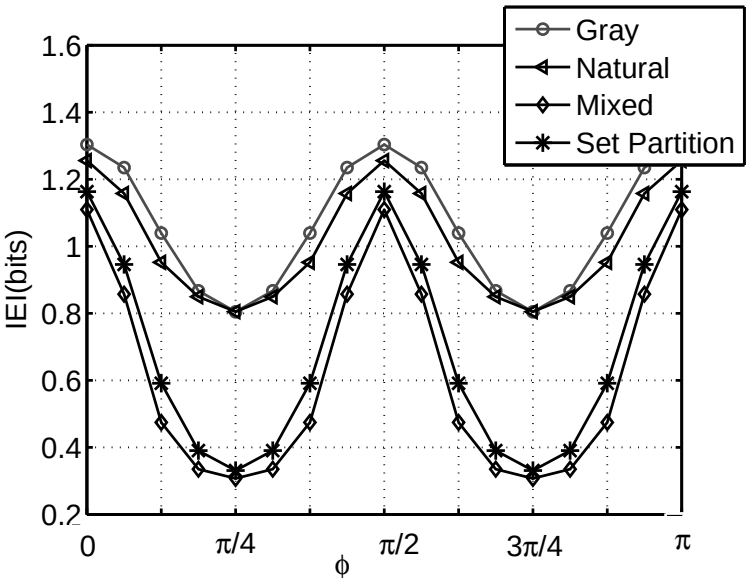


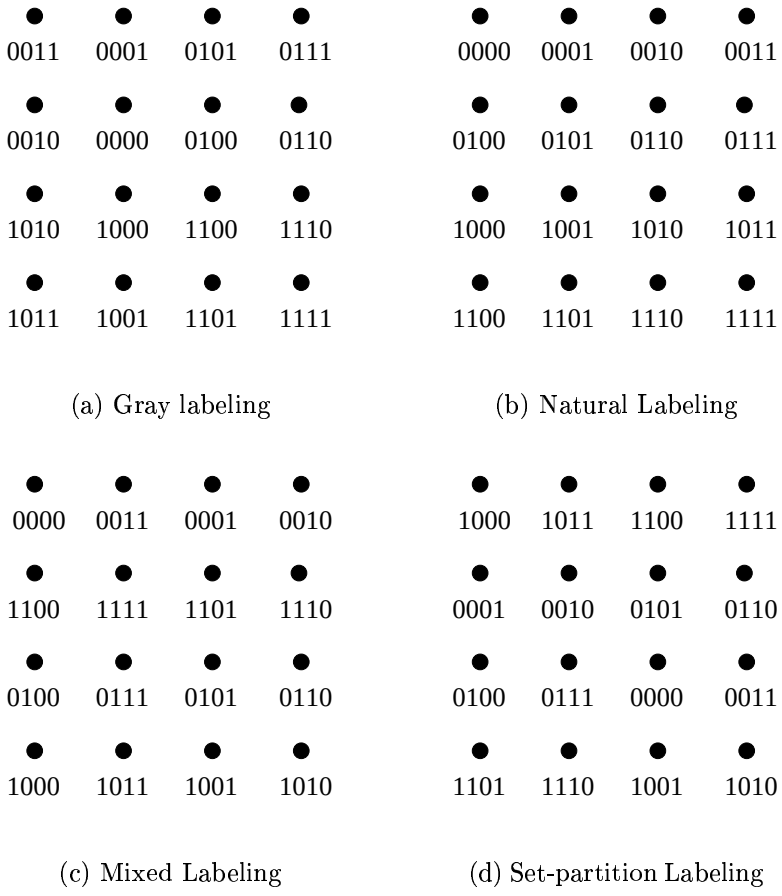
Figure 2.5: IEI of four different labelings for 16-QAM including Gray labeling, natural labeling, mixed labeling and set-partition labeling. The IEI is shown as a function of ϕ when $\kappa = 0$ and $\theta = 0$.

dependence on ϕ and θ can be "averaged out". This idea was first proposed in [4]. In the current work, this concept is generalized so that a time-varying unitary matrix,

$$Q_t = D(\beta_t)M(\alpha_t)D(\gamma_t), \tag{2.10}$$

is multiplied to the signal vector, $\mathbf{x}$, before it is transmitted (see Eq. (2.6) for definitions of $M(\cdot)$ and $D(\cdot)$). By controlling the three parameters, this technique creates a time-varying equivalent channel $\tilde{\mathbf{H}} = \mathbf{H}Q_t$. Using (2.9) and suppressing the subscript t,

$$\begin{aligned} \tilde{\mathbf{H}} &= \mathbf{H}Q \\ &= [\Lambda M(\phi)D(\theta)]\,[D(\beta)M(\alpha)D(\gamma)] \\ &= \Lambda\,[M(\phi)D(\theta+\beta)M(\alpha)]\,D(\gamma). \end{aligned} \tag{2.11}$$



(a) Gray labeling

(b) Natural Labeling

(c) Mixed Labeling

(d) Set-partition Labeling

Figure 2.6: 4 different labelings for 16-QAM

Because $M(\phi)D(\theta+\beta)M(\alpha)$ is also unitary, it can be decomposed into

$$M(\phi)D(\theta+\beta)M(\alpha) = e^{j\zeta}D(\tilde{\theta}_L)M(\tilde{\phi})D(\tilde{\theta}_R), \tag{2.12}$$

where

$$\tilde{\phi} = \frac{1}{2}\cos^{-1}\left(\cos(2\phi)\cos(2\alpha) - \sin(2\phi)\sin(2\alpha)\cos(\theta+\beta)\right), \tag{2.13}$$

$$\zeta = \measuredangle\left\{\cos\phi\cos\alpha - \sin\phi\sin\alpha\cos(\theta+\beta) - j\sin\phi\sin\alpha\sin(\theta+\beta)\right\}, \tag{2.14}$$

$$\tilde{\theta}_L = \measuredangle\left\{\sin(2\phi)\cos(2\alpha) + \cos(2\phi)\sin(2\alpha)\cos(\theta+\beta) + j\sin(2\alpha)\sin(\theta+\beta)\right\}, \tag{2.15}$$

$$\tilde{\theta}_R = \measuredangle\left\{\cos(2\phi)\sin(2\alpha) + \sin(2\phi)\cos(2\alpha)\cos(\theta+\beta) + j\sin(2\phi)\sin(\theta+\beta)\right\}. \tag{2.16}$$

Note that $e^{j\zeta}$ and $D(\tilde{\theta}_L)$ can be canceled out in the receiver. Hence the equivalent channel, represented in the same form as that in Eq. (2.9), becomes $\tilde{\mathbf{H}}_{eq} = \Lambda M(\tilde{\phi})D(\tilde{\theta})$, where $\tilde{\phi}$ is given by Eq. (2.13) and $\tilde{\theta} = \tilde{\theta}_R + \gamma$.

The ideal TVLT would vary α, β, and γ such that $\tilde{\phi}$ and $\tilde{\theta}$ both sweep uniformly over $[0, 2\pi)$. In that case, the SCTCM sees the same set of channels regardless of the actual channel angles ϕ and θ. Without any information of ϕ and θ in the transmitter, the best strategy is to sample uniformly over the 3-dimensional (α, β, γ) space. For fixed α and β values, if γ sweeps uniformly over $[0, 2\pi)$, $\tilde{\theta}$ also varies uniformly. However, due to the nonlinearity in Eq. (2.13), $\tilde{\phi}$ is not swept uniformly and still depends on ϕ and θ. Therefore, the performance of the SCTCM cannot be completely universal using TVLT.

Here we provide an analysis of the approximate performance of an SCTCM with TVLT. From the simulation results, we propose a model for EMI as a function of ϕ and θ as

$$\mathrm{EMI}(\phi,\theta) \approx \mathrm{EMI}_{\min} + K_1[1-\cos(4\phi)][K_2+\cos(2\theta)] \tag{2.17}$$

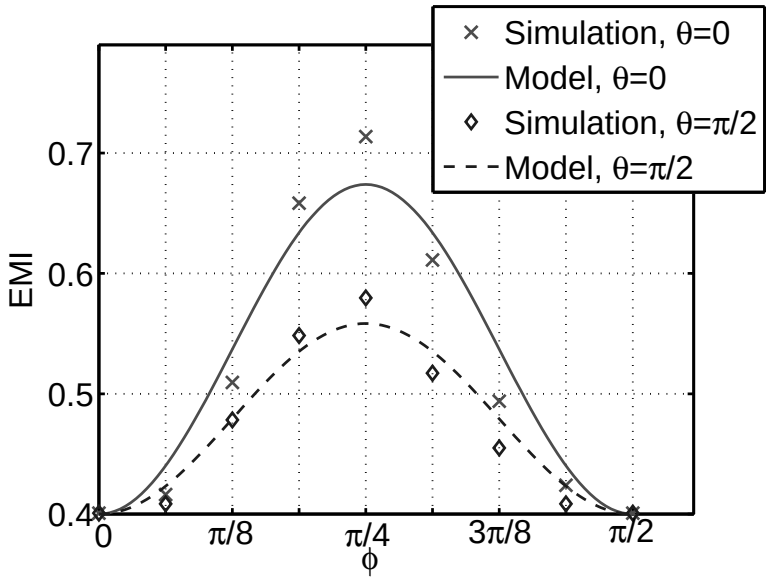


Figure 2.7: Simulated EMI requirement for SC-11 on singular channels ($\kappa = 0$) as a function of ϕ when $\theta = 0$ and $\theta = \pi/2$. The proposed model for EMI in Eq. (2.17) with $\text{EMI}_{\text{min}} = 0.40$, $K_1 = 0.0289$ and $K_2 = 3.74$ which minimizes the mean square error corresponds well to the simulation results.

where $\text{EMI}_{\min}$, K_1 and K_2 depend only on κ and are found to minimize the mean square error. Figure 2.7 shows that the model corresponds well to the simulated EMI for SC-11 on singular channels ($\kappa = 0$) with $\text{EMI}_{\min} = 0.40$, $K_1 = 0.0289$ and $K_2 = 3.74$. To obtain the approximation of EMI with TVLT, we assume that the EMI is the average of the instantaneous EMIs. Further, we assume a TVLT in which α, β, and γ are uniformly distributed, the average EMI with such a TVLT is given by

$$\begin{aligned} \text{EMI}^*(\phi, \theta) &\approx \frac{1}{N_\alpha N_\beta N_\gamma} \sum_\alpha \sum_\beta \sum_\gamma \text{EMI}(\tilde{\phi}, \tilde{\theta}) \\ &\approx \text{EMI}_{\min} + \frac{K_1 K_2}{4} \cdot [5 - \cos(4\phi)] \,. \end{aligned} \tag{2.18}$$

where N_α, N_β and N_γ are the number of α, β and γ swept over $[0, 2\pi)$ respectively. The full derivation of Eq. (2.18) is in the Appendix. Note that the dependence on θ is eliminated but the EMI still depends on ϕ, although with a smaller variance.

A more relaxed sufficient condition than uniformly distributed α, β, and γ for Eq. (2.18) to hold is

$$\sum_\alpha e^{j4\alpha} = 0, \;\; \sum_\beta \cos(2\theta + 2\beta) = 0, \;\; \sum_\gamma \cos(2\gamma) = 0. \tag{2.19}$$

Therefore, We propose two TVLT schemes:

- Fine-sampled TVLT (F-TVLT): Sweep α, β and γ over $[0, 2\pi)$ uniformly with a step size of $\frac{2\pi}{\sqrt[3]{N_s}}$, where N_s denotes the number of symbols per block.
- Simplified TVLT (S-TVLT): By Eq. (2.19), use $\alpha = 0$ or $\pi/4$, $\beta = 0$ or $\pi/2$, and $\gamma = 0$ or $\pi/2$ which yields a total of 8 different unitary matrices compared with N_s matrices in F-TVLT.

As shown in Figure 2.8, TVLT makes the EMI requirements more concentrated and the overall performance is close to universal. Figure 2.9 shows the

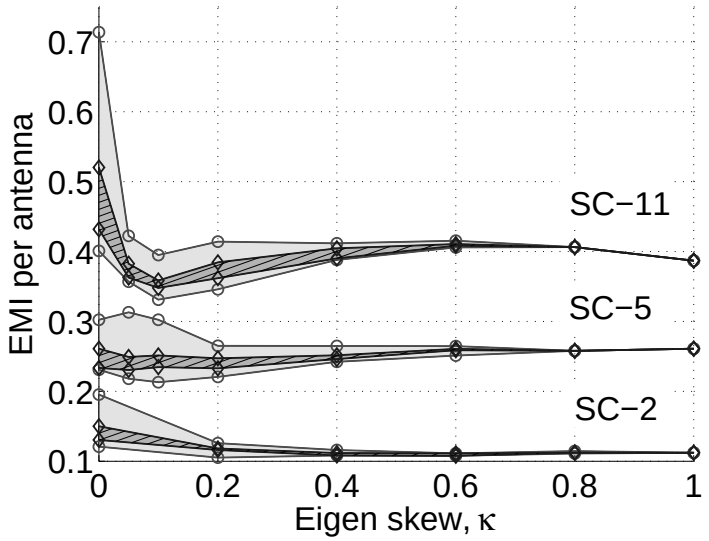


Figure 2.8: EMI requirements of SC-2, SC-5 and SC-11 over eigenvalue skew and eigenvectors. The gray area is the EMI region without TVLT while the shaded area is the EMI region with F-TVLT. The proposed SCTCMs perform consistently close to channel capacity for any of the 2×2 channels.

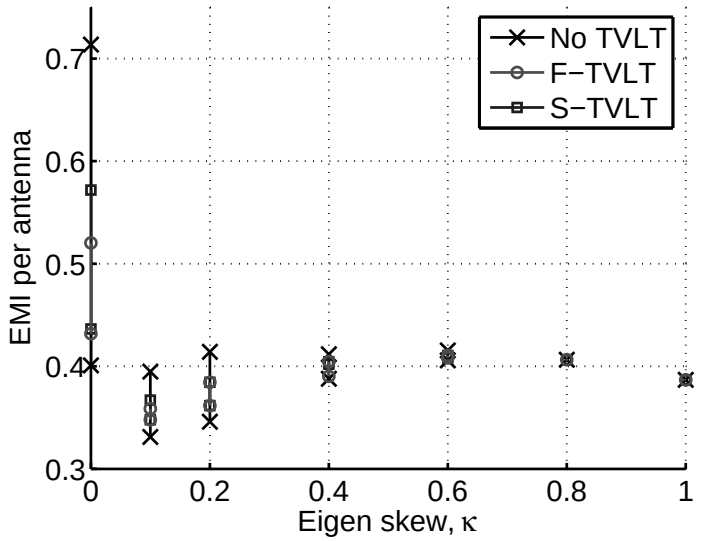


Figure 2.9: EMI requirements of SC-11 without TVLT and with F-TVLT and S-TVLT over eigenvalue skew and eigenvectors. It is found that both TVLT scheme deliver similar averaging effect which makes the code more universal.

EMI requirement of SC-11 without TVLT and with F-TVLT and S-TVLT respectively. The S-TVLT performance is almost identical to that of the F-TVLT.

2.5 Simulation Results

Figure 2.8 shows the EMI for SC-2, SC-5 and SC-11 with and without F-TVLT over eigenvalue skew. For the same eigenvalue skew, the maximum and minimum EMIs are marked and the shaded region represents the region of operation. All the simulations are done with an input blocklength of 10,000 bits, 12 iterations at the decoder, and BER of 10^{-5}. The TVLT technique improves the universality of the SCTCMs. As a result, SC-2 and SC-5 provide uniform EMI requirements of no more than 0.15 bits and 0.26 bits per antenna respectively. SC-11 provides a uniform EMI of no more than 0.41 bits except for the sudden increase to 0.53

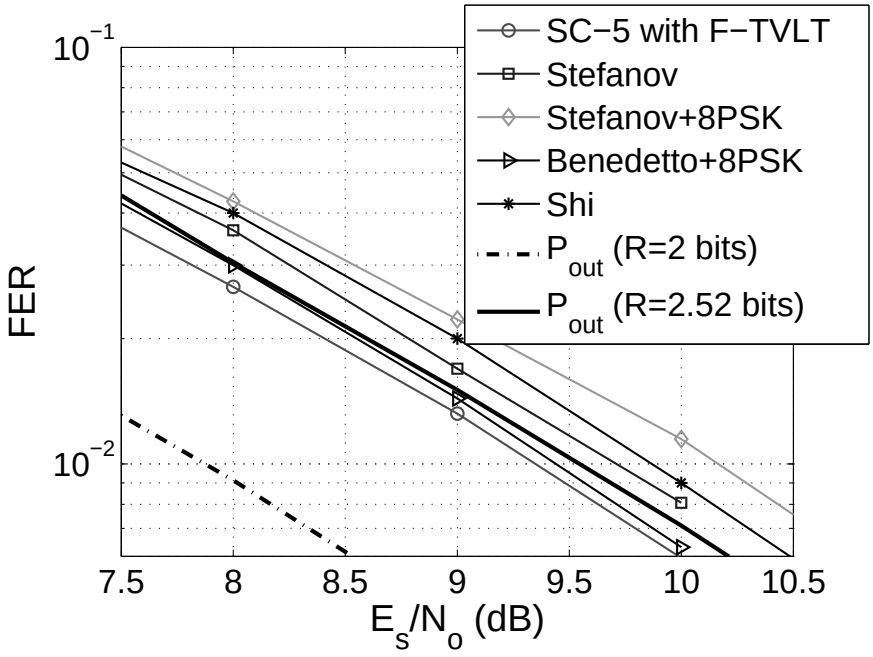


Figure 2.10: FER comparison of 2 bits/s/Hz codes including the SC-5 with TVLT, two PCCC-BICMs (by Stefanov [2]), a binary SCCC (by Benedetto [3]) mapped to 8PSK (SC-8), and a PCTCM (by Shi [4]) over quasi-static Rayleigh fading channel.

bits for the worst-case singular channel.

Next, we will compare the proposed universal SCTCMs with other coding schemes designed specifically for Rayleigh fading environment in terms of both average performance and universality (channel to channel performance). Among them, the best Rayleigh fading performance comes from Stefanov's [2] parallel concatenated convolutional code with bit-interleaved coded modulation (PCCC-BICM) which is 1.8 dB from the outage probability at FER=10^{-2}. Turbo-TCMs [24], [4], [32] are reported to operate at 2.0 to 2.2 dB from the outage probability. The universal SC-5 with F-TVLT is only 1.5 dB from the outage probability. SC-8, which maps a binary SCCC designed for AWGN channel by Benedetto [3]

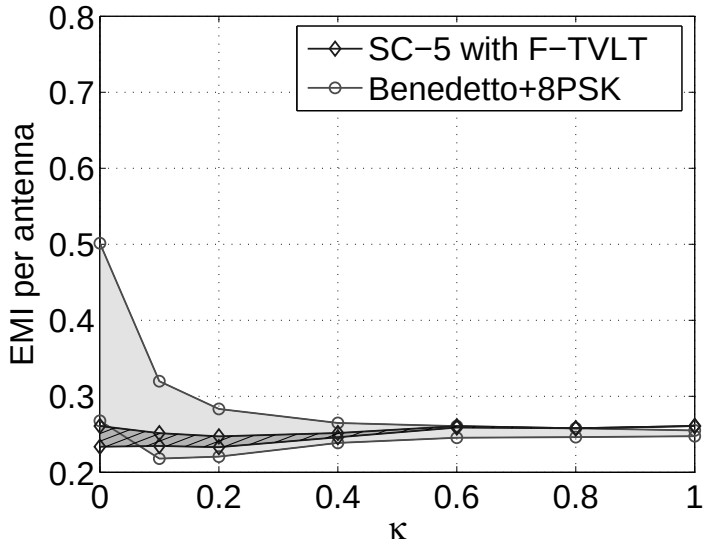


Figure 2.11: EMI requirement regions for SC-5 with F-TVLT and a binary SCCC (by Benedetto [3]) mapped to 8PSK (SC-8) transmitting at 2 bits/s/Hz. SC-5 with F-TVLT provides a more consistent channel-by-channel performance than SC-8, and a slightly better (0.1 dB) Rayleigh fading performance as shown in Figure 6.

directly to 8PSK, is 1.6 dB from the outage probability. For the Rayleigh fading performance, SC-5 and SC-8 were simulated with input blocklength of 260 bits, which is the same as that in [2] and [24]. [4] and [32], however, have blocklengths of 8,196 and 1,024 bits respectively. Moreover, the number of iterations in these papers are not exactly the same. So a fair conclusion would be that these codes all perform similarly on the Rayleigh fading channel.

On the other hand, Figure 2.11 and Figure 2.12 shows the EMI requirement region for SC-5, SC-8 and two PCCC-BICMs, all with blocklength of 10,000 bits. SC-5 clearly has a much narrower EMI region and hence is more universal than others, especially after applying TVLT. Note that the PCCC-BICM in [2] was

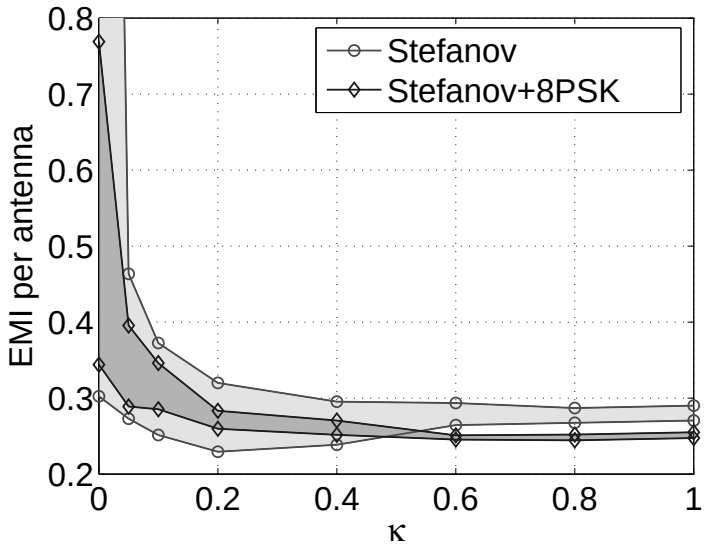


Figure 2.12: EMI requirement regions for two PCCC-BICMs (by Stefanov [2]) transmitting at 2 bits/s/Hz over the 2×2 matrix channel in Eq. (2.9). Using a larger constellation (8PSK) improves the universality, but both codes are still not as universal as SC-5.

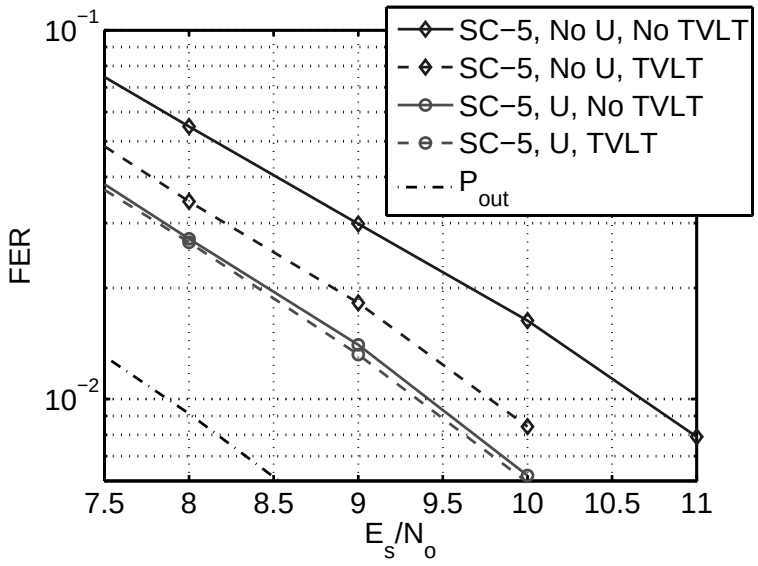


Figure 2.13: The effect of TVLT and $U^\dagger$ multiplication on the Rayleigh fading performance of SC-5. The $U^\dagger$ matrix improves the FER significantly. The TVLT helps more in the case of no $U^\dagger$ and helps little when $U^\dagger$ is applied.

originally rate-1/2 and mapped to a QPSK constellation to transmit 2 bits/s/Hz over two antennas. The system, in the worst condition of $\kappa = 0, \phi = 0$, becomes uncoded and thus requires a very large EMI. Therefore, we also designed a rate-1/3 PCCC-BICM mapped to 8-PSK. The worst case EMI is reduced to 0.8 bits per antenna but it is still not as universal as SC-5.

In Sec. IV-C in [4], Shi found that by multiplying $U^\dagger$ (see Eq. (2.5)), i.e. the R matrix in Shi's notation, at the receiver, the FER improves significantly but the effect of TVLT is invisible with or without $U^\dagger$. As shown in Figure 2.13, we also found that $U^\dagger$ is indeed crucial for the SCTCM performance on the Rayleigh fading channel. However, unlike Shi, who only uses *real* unitary matrices in their experiments, our generalized TVLT matrices scan the whole *complex* unitary matrix space and provide a better averaging effect. As a result, for SC-5, TVLT

improves the FER by 0.9 dB at FER=10^{-2} in the case of no $U^{\dagger}$ and merely 0.1 dB when $U^{\dagger}$ is applied. In general, the averaging over channels in a Rayleigh simulation hides the channel-by-channel consistency provided by the TVLT.

Since SC-5 requires a uniform EMI no more than 0.26 bits per antenna (i.e. 0.52 bits per transmission), SC-5 should be able to transmit 2 bits successfully over any channel which provides an MI of more than 2.52 bits. Figure 2.10 shows that the FER of SC-5 with F-TVLT is close to the outage probability of MI=2.52 bits. The beauty of the universal SCTCM is that SC-5 can perform close to the outage probability of a MI=2.52 bits not only for Rayleigh fading but *any* quasi-static fading channels.

2.6 Conclusions

Designs of universal space-time SCTCM schemes are proposed to deliver close-to-capacity performance on any of the quasi-static 2×2 channels. Using the de-multiplexing MIMO scheme, universal SCTCMs over PFC show their universality over the eigenvalue skew because the PFC is equivalent to the diagonal channels. Then for non-diagonal channels, it is found that the performance of SCTCM may degrade significantly due to the convergence behavior of the iterative decoder. We observed that the relation between EMI and IEI can be used to facilitate the comparison of different constellations and labelings. Experiments with fixed constellations and labelings could not achieve universal IEI. Hence, TVLT is used to rotate the channel matrix over time and is shown to effectively mitigate the dependency of EMI on the eigenvectors. As a result, the proposed SCTCMs of 1.0, 2.0 and 3.0 bits per channel use require a consistent EMI of 0.11-0.15, 0.23-0.26 and 0.35-0.53 bits per antenna and are 1.9, 2.2, and 2.4 dB respectively from the outage probability at FER=10^{-2} on the quasi-static Rayleigh fading

channels. However, the main point of designing such codes is that they are not designed specifically for Rayleigh fading, and will work close to the theoretical limit on any quasi-static fading channel.

CHAPTER 3

Lowering the Error Floors of Irregular High-Rate LDPC Codes by Graph Conditioning

3.1 Introduction

Low density parity check (LDPC) codes have generated much research interest because of their capacity approaching performance. Most of the work in the literature focuses on low-rate and very long codes. The goal of this chapter is to design high-rate LDPC codes with low encoding complexity and low error floors. Yang *et* al. [33] proposed the specialized class of LDPC codes called the extended irregular repeat-accumulate (IRA) codes that have low complexity encoders. Richardson *et* al. [34] created the density evolution technique to optimize the degree distributions in cycle-free bipartite graphs as the block length and the number of iterations go to infinity. For finite block length, Tian *et* al. [17] proposed an efficient graph conditioning algorithm called the approximate cycle extrinsic message degree (ACE) algorithm to lower the error floor by avoiding harmful short cycles. The ACE algorithm is effective for the construction of low-rate LDPC codes. In this chapter, we explore the effectiveness of ACE algorithm for high-rate ($R \geq 1/2$) code designs.

Following the common notation in the literature, k and n denote the input block length and codeword length respectively. Thus the code rate $R = k/n$.

The parity check matrix, H, is an $(n-k) \times n$ sparse matrix which can be divided into

$$H = [H_1 \mid H_2], \tag{3.1}$$

where the two submatrices, H_1 and H_2, have k and $n-k$ columns respectively as defined in [33].

The H_2 matrix must have $n-k-1$ weight-2 columns plus one weight-1 column with a bi-diagonal structure [33], i.e.,

$$H_2 = \begin{bmatrix} 1 & & & & & & \\ 1 & 1 & & & & & \\ & 1 & 1 & & & & \\ & & 1 & \ddots & & & \\ & & & \ddots & 1 & & \\ & & & & 1 & 1 \end{bmatrix}. \tag{3.2}$$

Then H_2^{-T} is a matrix with all the upper-diagonal elements equal 1. Its generating function is $\frac{1}{1+D}$ and can be easily implemented using an accumulator. Therefore, the generator matrix, G, of the extended IRA codes can be written as

$$G = [I \mid H_1^T H_2^{-T}], \tag{3.3}$$

which only requires a sparse matrix multiplication followed by an accumulator and yields low encoding complexity as depicted in Figure 3.1.

The LDPC code is decoded by passing message on the bi-partite graph associated with the parity-check matrix of the code. Richardson et al. [34] developed density evolution technique to analyze and optimized the degree distributions of the variable nodes and check nodes. The decoding threshold can be found for infinite blocklength code and assuming the associated graph is tree-like. The degree distribution can be defined from the edge's point of view or from the nodes

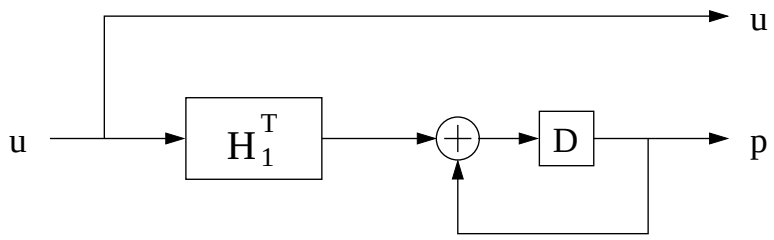


Figure 3.1: Low complexity encoder structure of the extended IRA LDPC code which consists of a sparse matrix multiplication followed by an accumulator.

point of view. We will use the edge's point of view following Richardson's [34] notations. $\lambda(x) = \sum_i \lambda_i x^{i-1}$ denotes the variable node degree distribution from the edge's perspective and $\rho(x) = \sum_i \rho_i x^{i-1}$ represents the check node distribution. $N_v(l)$ denotes the number of degree-l variable nodes in the Tanner graph. In general, the weight-1 column should be avoided during the degree distribution optimization. So, throughout the chapter, there is no degree-1 node in the degree distributions. One of the degree-2 nodes is converted to degree-1 when the parity check matrices of the extended IRA codes are constructed.

The rest of the chapter is organized as follows. Section 3.2 presents the code design criteria and graph-conditioning algorithm. Section 3.3 applies the code designed criteria to design long and medium blocklength high-rate LDPC codes with low error floors. In Section 3.4, we propose a rate-compatible scheme using information-nulling for high-rate LDPC codes and compare their performance to stand-alone codes. Section 3.5 concludes the chapter.

3.2 Code Design Criteria

3.2.1 Existing Criteria

Major criteria for designing the parity check matrix of the LDPC codes [33], [34] are summarized as follows:

(1) Optimize the degree distributions using density evolution.

(2) Forbid cycles involving only degree-2 variable nodes and avoid length-4 cycles.

(3) Low-degree variable nodes are made to correspond to non-systematic bits.

3.2.2 ACE Algorithm

Definition 1 *A variable node set is called a stopping set if all its neighbors are connected to it at least twice.*

The size of the smallest stopping set determines the error floor behavior. However, it is hard to find the smallest stopping set because the complexity is too high.

Definition 2 *The approximate cycle extrinsic message degree (ACE) of a length* $2d$ *cycle is* $\sum_i (d_i - 2)$*, where* d_i *is the degree of the* i^{th} *variable node in the cycle.*

An LDPC code has property (d_{ACE}, η) if all the cycles whose length is $2d_{\mathrm{ACE}}$ or less have ACE values of at least η. The ACE algorithm is an efficient Viterbi-like linear complexity algorithm proposed in [17] to detect and avoid harmful short cycles during code construction. Given the degree distribution, $\lambda(x)$, columns of the parity check matrix are generated one at a time starting from low-degree nodes. The edges of every new column are generated randomly and the ACE

algorithm checks whether the (d_{ACE}, η) requirement is met. If not, this column must be generated again. This step repeats until the whole parity check matrix is generated.

It is more difficult to apply the ACE algorithm to high-rate codes than to low-rate codes because with the same block length, high-rate codes have fewer check nodes, i.e. columns are shorter in H, and the number of cycles will increase which makes it harder to guarantee certain (d_{ACE}, η) values. For high-rate LDPC codes, the optimal degree distributions usually have more than $n-k$ degree-2 variable nodes and there always exist cycles between only degree-2 nodes [35]. Figure 3.2 gives two example of four length-4 degree-2 columns. The first matrix has a length-8 cycle which is the longest cycle possible while the second matrix has a length-4 cycle. Note that if $N_v(2) \geq n-k$, any combination of $n-k$ degree-2 columns forms cycle(s) and these cycles are all stopping sets. Figure 3.3 shows the curves of the optimal ratio of degree-2 nodes and the ratio of parity bits. The two curves cross each other at approximately $R = 1/2$. Therefore, if we don't constrain the number of degree-2 nodes and choose the optimal degree distributions, the ACE algorithm can only help to lower the error floor a little, but it is still high since too many loops with small ACE values.

To lower the error floor further, we must decrease the number of degree-2 variable nodes and adopt a degree distribution optimized with the constraint $N_v(2) < n-k$. The error floor is lowered at the cost of a small increase in the threshold SNR. This tradeoff between threshold and error floor is also observed in many LDPC and turbo-code papers [3], [7], [17].

An exciting discovery is that for this constrained degree distribution, the graph conditioning can be carried out successfully and lower the error floor dramatically. The semi-regular LDPC codes in [33] also constrain the number of

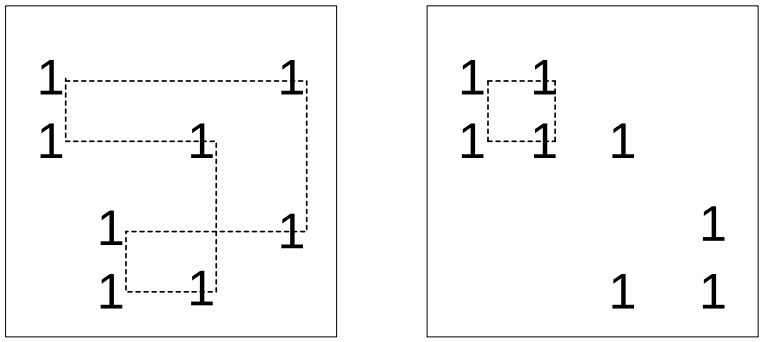


Figure 3.2: An example of the number of degree-2 nodes equals to $n-k$, where $n-k=4$. There must exist cycles between these nodes and every cycle consisting of all degree-2 nodes is a stopping set. Example (1) has a length-8 cycle and example (2) has a length-4 cycle.

degree-2 nodes but use a regular H_1 to guarantee low error floors. Figure 3.4 shows the theoretical gap to BPSK capacity for the three types of codes. The semi-regular design trades about 0.5 dB of threshold SNR for low error floors while our design only requires 0.1 dB increase of threshold SNR to have a low error floor because of the effective graph-conditioning algorithm.

3.3 Code Design Examples and Simulation Results

3.3.1 Long Block Length High-Rate LDPC Codes

Figure 3.5 includes design examples of rate-3/4 LDPC codes with blocklength $n = 10688$ bits simulated on the AWGN channel using BPSK modulation for a maximum of 200 iterations using message passing decoder. Three different degree distributions are simulated here (See Table 3.1). Scheme-A is the optimal degree

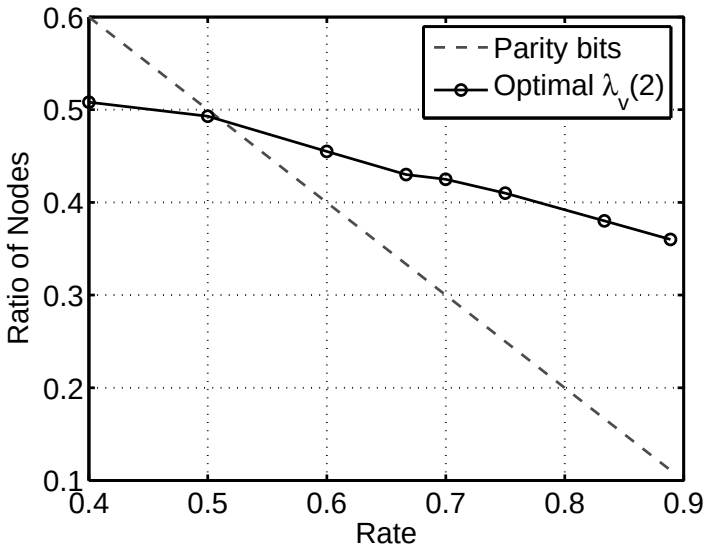


Figure 3.3: The ratio of degree-2 nodes, $\lambda_v(2)$, in the optimal degree distribution as a function of the code rate. Also plotted is the ratio of parity check bits which equals $1 - R$. Note that for approximately $R \geq 1/2$, $\lambda_v(2)$ is greater than $1 - R$ which results in loops between only degree-2 nodes.

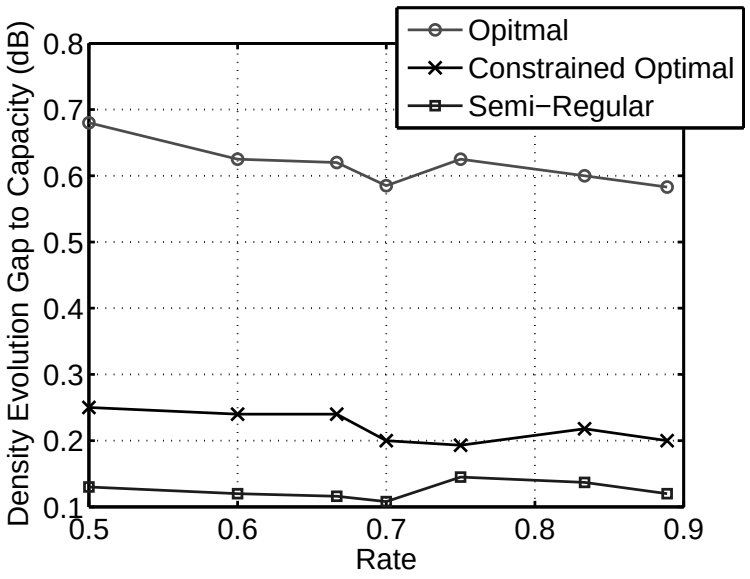


Figure 3.4: The gap to capacity for the optimal, constrained optimal, and semi-regular degree distributions found using density evolution technique as a function of the code rate.

distribution without any constraint. Scheme-B is the optimal degree distribution with the constraint $N_v(2) < n - k$ while scheme-C is the semi-regular code with a regular H_1 of column weight 5.

Our results show that for scheme-A, the BER error floor is at about 10^{-3} for a randomly generated LDPC code without graph conditioning. The ACE algorithm can only achieve $(d_{\text{ACE}}, \eta) = (3, 4)$ for this degree distribution to improve the error floor to a level between 10^{-4} and 10^{-5}, which is still high. Scheme-B without graph conditioning, (B, -, -), has an error floor around BER = 10^{-5}. Scheme-C uses the same design proposed in [33] which adopts a non-optimal degree distribution and trades some threshold for a lower error floor. Note that we were able to construct a (C, 5, 6) code but its performance is almost identical to the (C, -, -) code at the BER above 10^{-7}. This is because graph conditioning plays

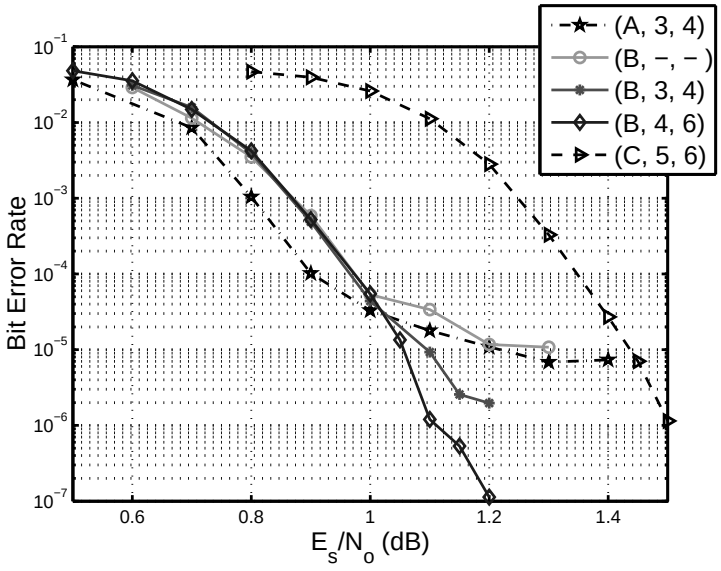


Figure 3.5: Simulation results for $(8016, 10688)$ codes for 200 iterations. The BPSK capacity at R=3/4 is E_s/N_o=0.38 dB. Codes are labeled by (Scheme, d_{ACE}, η).

the role of lowering the error floor but the error floor of scheme-C is lower than 10^{-7} even without any graph conditioning. The results of (B, 3, 4) and (B, 4, 6) are exciting because with proper graph conditioning, the error floor can be lowered from 10^{-5} to at least 10^{-7} with little compromise (less than 0.1 dB) of the threshold of convergence. As a result, at BER $= 10^{-5}$, our best code, (B, 4, 6), performs within 0.67 dB of the Shannon limit and is 0.38 dB better than the semi-regular code.

3.3.2 Medium Block Length High-Rate LDPC Codes

For applications requiring high throughput such as the wireless local area network (WLAN), high-rate LDPC codes are considered a good candidate channel coding scheme. The decoder complexity and delay constraint limit the code length to be

Table 3.1: The three degree distributions used in the long LDPC code design example. Scheme-A is the optimal degree distribution, Scheme-B is optimal with the constraint $N_v(2) \leq n-k$, and Scheme-C is the semi-regular code with $n-k$ degree-2 variable nodes and k degree-5 variable nodes.

	Scheme-A (optimal)	Scheme-B (constrained optimal)	Scheme-C (semi-regular)
λ_2	0.1970	0.1250	0.1176
λ_3	0.0801	0.4460	
λ_4	0.2410		
λ_5	0.0082		0.8824
λ_{11}		0.4078	
λ_{12}		0.0213	
λ_{14}	0.4736		
ρ_{16}		1.0000	
ρ_{17}			1.0000
ρ_{18}	0.700		
ρ_{19}	0.300		

less than a few thousand bits and the number of decoding iterations to be about 10-20. Unlike the previous example which adopts a long, length-10688, LDPC code, here we will limit the codeword length to 2000 bits and design LDPC codes using the afore mentioned criteria.

Figure 3.6 shows the simulation results of stand-alone rate-1/2, 2/3, 3/4, 4/5, 5/6 and 8/9 LDPC codes designed separately, all with $n = 2000$ bits. The variable node degrees are limited to degree 2, 3, and 10 for implementation simplicity. Note that the achievable (d_{ACE}, η) region is still an open problem for a given blocklength and the degree distribution. Because of the linear complexity of the ACE algorithm, it is not too computationally intensive to run the algorithm starting from small (d_{ACE}, η) values and increase the constraint until such codes cannot be constructed. Then several candidate codes with largest η for each d_{ACE} value are simulated to determine the best code.

Note for the same rate (rate-3/4) and the same $d_{\text{ACE}} = 4$, the largest achievable ACE value, η, in the previous example is 6 while it decreases to 3 as the code blocklength is reduced from 10688 to 2000. Although the achievable (d_{ACE}, η) is smaller especially for the highest-rate code (rate-8/9), the graph conditioning algorithm can still effectively lower the error floors. All the codes in Figure 3.6 have error floors lower than 10^{-6}. Figure 3.7 plots the SNR gap to the capacity for these stand-alone LDPC codes as a function of the code rate. Simulations are run for 10 and 100 iteration respectively. SNRs are measured at BER $= 10^{-5}$. Also plotted as a reference is the density evolution threshold which is the theoretical limit of the given degree distributions used at each rate when the blocklength and number of iteration go to infinity.

It is observed that the SNR gap to the capacity is larger at lower rates than at higher rates when only 10 iteration is allowed due to throughput or delay

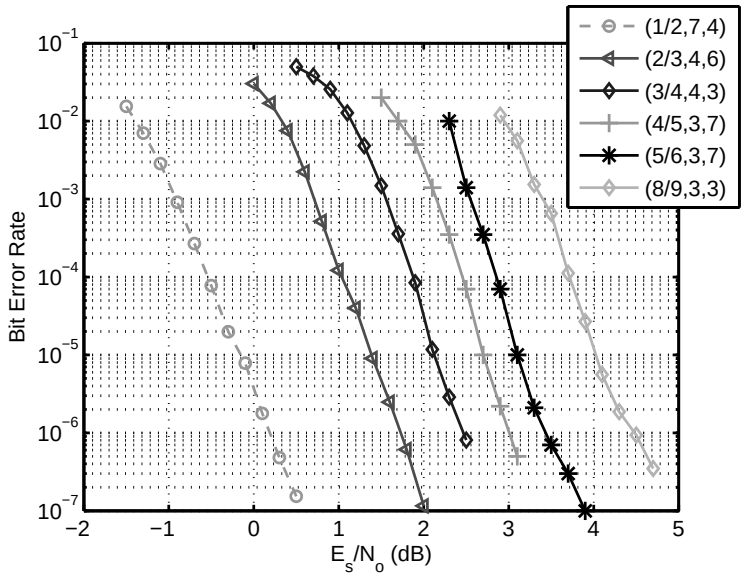


Figure 3.6: BER curves of medium blocklength ($n = 2000$ bits) LDPC codes constructed by the ACE algorithm with the constrained optimal degree distribution. The ACE algorithm applies well as in the long blocklength case, and the proposed codes all have BER error floors lower than 10^{-6}. The simulations use 10 decoder iterations and the channel is AWGN channel. BER curves are labeled as (R, d_{ACE}, η).

constraints. This is because for the same block length, low-rate codes have more check nodes and each check node has a lower degree such that it requires more iterations to converge. When the number of iterations is increased to 100, it is found that all the stand-alone codes are about 1.2 to 1.5 dB from the capacity at BER $= 10^{-5}$.

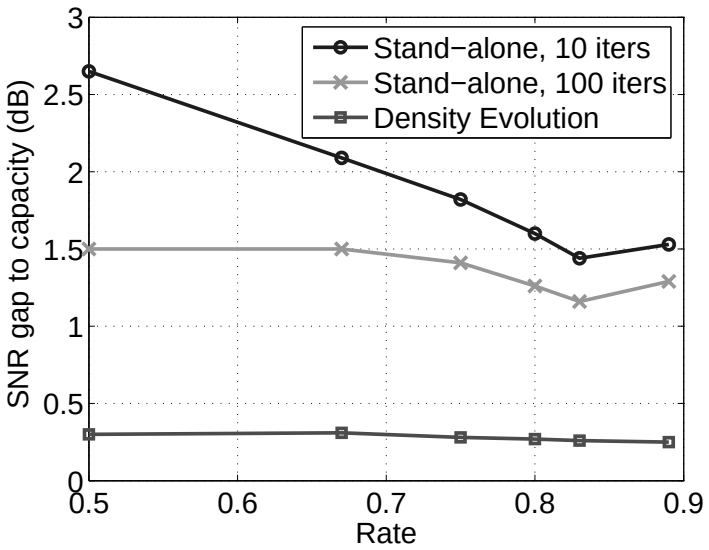


Figure 3.7: When the allowed variable node degrees are only 2, 3 and 10, the density evolution threshold of the constrained optimal code is consistently good. The other two curves are the SNR gap to the capacity for the stand-alone codes at BER $= 10^{-5}$ on the AWGN channel. The simulations use 10 and 100 decoder iterations respectively. All codes have blocklength of 2000 bits.

3.4 Rate-Compatible LDPC Codes Using Information-Nulling

Error correction codes are often designed for a certain code rate which matches the error-correcting requirement of the worst case channel or the average channel conditions. However, in some communication systems, the channel could be time-varying such that different levels of error correcting capability is required. Usually, the hardware implementation of channel codes is code-dependent and non-reconfigurable. So it is inefficient to implement stand-alone, i.e., separately designed, codes for each code rate.

The concept of rate-compatible codes aims to reduce the hardware complexity by using only one encoder/decoder pair based on a single code called the mother code, and the rate can be changed using techniques that do not increase the hardware complexity by much. A rate-compatible technique provides a method to vary the code rate from the mother code such that the resulted codes all have good performances compared to stand-alone codes.

One common technique to achieve rate-compatibility is puncturing which results in so called rate-compatible punctured codes (RCPC) [36][37][38]. In puncturing, the lowest-rate code is chosen as the mother code. A portion of the parity bits of the mother code is not transmitted, i.e., punctured, to achieve the desired code rate On the decoder side, it is assumed that the decoder knows exactly the positions of the punctured bit and treat them as erasures. In the message-passing LDPC decoder, the log-likelihoods (LLRs) of the punctured bit are set to be zero before the decoding iterations begins.

Another technique is information-nulling, a.k.a. code shortening. In this technique, the highest-rate code is the mother code and the code rate can be lowered

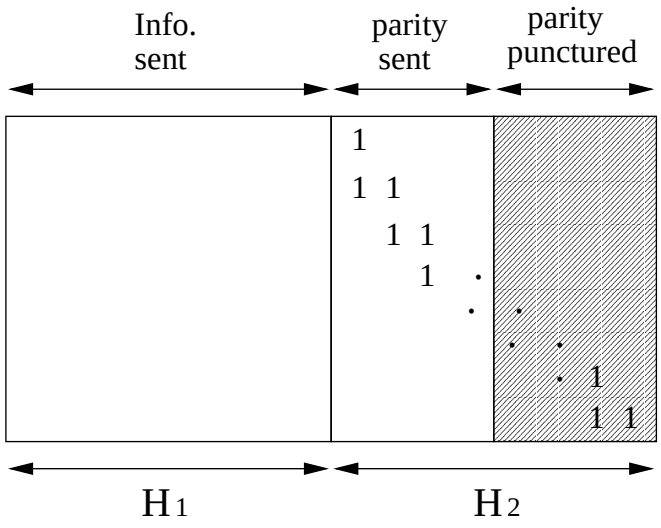
by setting part of the information bits to be zeros. The decoder knows exactly the positions of the nulled bits and treats them as having infinity reliability. In the message-passing LDPC decoder, the LLRs of the nulled bit are set to be negative infinity throughout the decoding process. Figure 3.8 illustrates the concepts of puncturing and information-nulling and their effect on the parity check matrix.

Ha and McLaughlin [39] proposed the optimal puncturing ratio to puncture LDPC codes using density evolution such that the punctured codes has good thresholds when decoded using the iterative message-passing decoder. However, this is not applicable to designing the eIRA LDPC codes because all the parity bits of the eIRA codes have degree of 2 in order to have the low complexity encoder. Besides, the graph-conditioning is difficult to be applied to the puncturing scheme and thus good graph conditioning is not guaranteed and thus the punctured LDPC codes may suffer from high error floors.

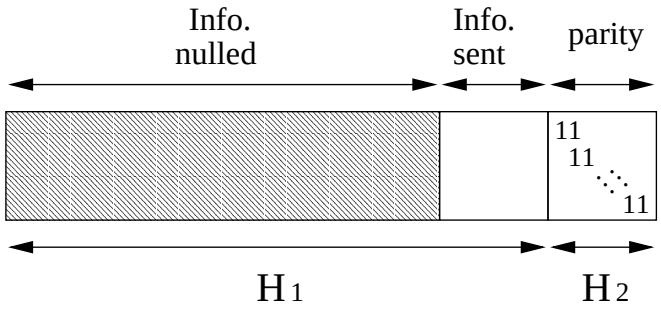
Information-nulling, on the other hand, turns out to be a better approach because the optimal degree distribution and good graph-conditioning can both be guaranteed. In [40], Tian proposed an algorithm that combines the graph-conditioning and information nulling to design rate-compatible LDPC codes. With information-nulling, the equivalent parity check matrix of the lower rate LDPC code results from deleting the columns of the mother code parity matrix corresponding to the nulled bits.

Tian showed that for the low-rate range, $0.1 \leq R \leq 0.5$, a rate-1/2 mother can be constructed such that all of the shortened codes are still ensured certain ACE constraint. Here we will take similar approach to design rate-compatible LDPC codes for high-rate, $0.5 \leq R \leq 0.9$, application.

A key requirement of the algorithm to work is the consistency of degree distributions over the rate range. When a rate-R_m, lenth-n_m mother code is shortened



(a) Puncturing



(b) Information-Nulling

Figure 3.8: Two different rate-compatible techniques: (a) Puncturing uses the lowest-rate code as the mother code and puncture some of the parity bits. (b) Information-nulling starts from the highest-rate code and null part of the information bits.

to rate-R' using information-nulling, the codeword length becomes

$$n' = n_m(\frac{1-R_m}{1-R'}) \tag{3.4}$$

Let the constrained optimal variable node degree distribution from the node's perspective at rate-R for degree-l be $\lambda_v(R,l)$, the number of variable nodes of degree-l at rate-R' with code length-n' is

$$N_v(R',l) = n'\lambda_v(R',l) = n_m(\frac{1-R_m}{1-R'})\lambda_v(R',l) \tag{3.5}$$

Figure 3.9 plots the the constrained-optimal number of variable nodes of degree-2, 3, and 10 at each rate with $n_m = 2000$. Only three different variable node degrees are allowed for design simplicity. Note that since $N_v(R,l)$ is a non-decreasing function of the rate for each variable node degree, we can use the following algorithm to construct a rate-compatible LDPC code which code rate may be lowered using information nulling.

(1) Specify the corresponding code length at each rate of operation.

(2) Starting from the lowest-rate code, the process of generating the columns from the current code to the next-rate code is called a stage. Calculate the number of columns of each degree to be generated at each stage.

(3) Fix the d_{ACE} value and apply appropriate η values at each stage. Note that low-rate codes in general has higher achievable η which means the ACE constraint is also consistent.

The proposed rate-compatible codes have not only constrained-optimal degree distributions but also good ACE constraint. Figure 3.10 shows the performance of the rate-compatible code. Note that none of the codes suffers from high error floor which means the graph-conditioning works effectively. Hence the shortened

code is as good as a specifically designed constrained-optimal LDPC code with the same length.

Compared to the same rate-compatible scheme but using a semi-regular codes with degree-2 and degree-4 nodes as the mother code, the semi-regular codes with no graph-conditioning have error floors even higher than the constrained-optimal graph-conditioned codes and are 0.2 to 0.4 dB worse at BER $= 10^{-5}$. We can definitely apply the ACE algorithm to the semi-regular codes to lower their error floors but the threshold SNR will remain the same, which means the semi-regular codes are still inferior to the constrained-optimal codes. We are also interested in how much performance loss is resulted from the reduction of effective blocklength in the information-nulling scheme. Figure 3.11 compares the SNR gap to the capacity for the rate-compatible LDPC codes and stand-alone LDPC codes when 10 and 100 iterations are used. A 1.0 dB loss is observed at rate-1/2 when compared to the stand-alone LDPC codes with a fixed code length due to this shortened blocklength. In [41] and Chapter 4, rate-compatible LDPC codes with constant blocklength is proposed to avoid this performance loss resulted from the loss of blocklength in both information-nulling and puncturing schemes.

3.5 Conclusion

In this chapter, we applied the graph conditioning algorithm (ACE) to high-rate irregular extended IRA LDPC codes and showed that it can effectively lower the error floor even though the graph-conditioning becomes more difficult as the rate increases. For code rate greater than 1/2, the optimal degree distribution has more than $n-k$ degree-2 nodes and it results in high error floors. The proposed LDPC codes with constrained optimal degree distributions can trade only 0.1 dB of threshold SNR for error floors which are several orders lower. This indicates

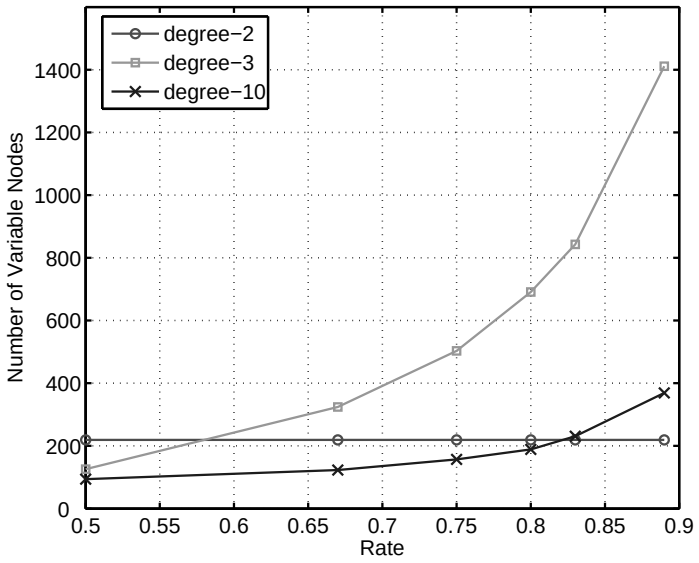


Figure 3.9: The constrained optimal number of variable nodes of degree-2, 3, and 10 considering the code-shortening effect of information nulling. The non-decreasing curves of each degree enables the construction of a rate-compatible code with optimal thresholds at each rate.

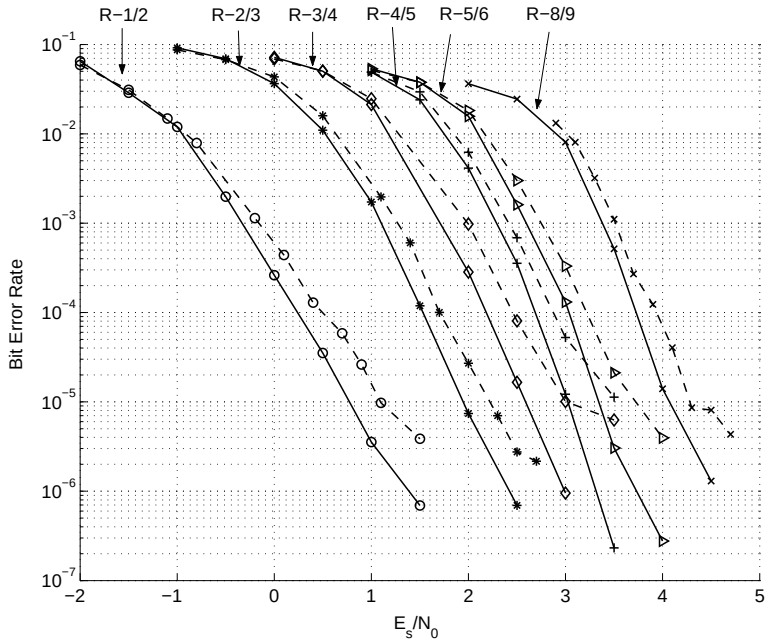


Figure 3.10: BER simulation results of rate-compatible LDPC codes using a length-2000, rate-8/9 mother code. The code rates are lowered using information-nulling. All codes are simulated for 10 iterations on the AWGN channel. Solid lines represents the LDPC codes designed using the algorithm in Section 3.2 and dash lines show the simulation results using a semi-regular mother codes without graph-conditioning.

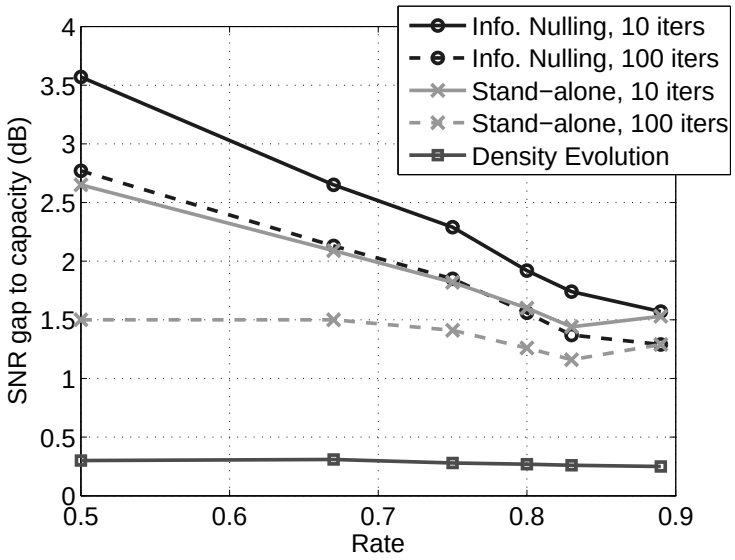


Figure 3.11: Comparison of the gap of the capacity for the rate-compatible LDPC codes using informatio-nulling and stand-alone LDPC codes. The SNR gap is measured at BER $= 10^{-5}$ on the AWGN channel. The simulations use 10 and 100 decoder iterations respectively. The rate-compatible code uses a rate-8/9, length-2000 mother code, and the stand-alone codes have blocklength of 2000 bits.

that degree distributions near the optimal degree distribution are still quite good in terms of threshold SNR and may be better than the optimal degree distribution for practical block lengths. Code design examples from short block length (a few hundred bits) to long block length (around 16000) all perform well with both low SNR thresholds and low error floors. In addition, we proposed a rate-compatible LDPC code design by combining information-nulling and graph-conditioning. By designing a highest rate mother code with all the desired lower rates as the constraint, the proposed rate-compatible LDPC codes maintain constrained optimal degree distribution and reasonably good ACE constraint throughout the high-rate range. The rate-compatible schemes suffers a loss of about 1 dB at the lowest rate (rate-1/2) due to the reduced effective blocklength.

CHAPTER 4

Multiple-Rate Low-Density Parity-Check Codes with Constant Blocklength

4.1 Introduction

The performance of an LDPC code using an iterative message-passing decoder is determined by its convergence threshold and its error floor. Richardson *et* al. [34] created the density evolution technique to optimize the degree distributions in cycle-free bipartite graphs as the blocklength and the number of iterations go to infinity. As for the error floor, it is generally believed that the height of the error floor depends on the graph property of the Tanner graph associated with the parity-check matrix. For finite-blocklength LDPC codes, it is inevitable that there are cycles in the Tanner graph. Various techniques such as girth-conditioning [42][43], or graph-conditioning based on stopping sets [17][44] focus on avoiding harmful short cycles in the Tanner graph to lower the error floor.

In Section 3.4, the importance of having a rate-compatible channel codehas been discussed. The goal of a good rate-compatible LDPC code is to have comparable performance as stand-alone LDPC codes designed specifically for each rate while still maintains approximately the same hardware complexity of a single LDPC code. Two different rate-compatible LDPC schemes, puncturing and information-nulling, are discussed in Section 3.4. Puncturing can maintain opti-

mal degree distribution throughout the rate range but good graph connectivity is hard to be guaranteed. Information-nulling can not only maintain optimal degree distribution but also guarantee good graph connectivity. However, both puncturing and information-nulling have the problem of the reduction of effective blocklength when more bits are punctured or nulled. The performance loss due to the reduction of blocklength can be significant when the mother code length is medium or short.

Therefore, we proposed a rate-compatible scheme called the row-combining technique to design LDPC codes that keeps the code length constant while varying the code rate. Row-combining uses the lowest-rate code as the mother code and combines the rows, i.e., the parity-check equations to become a higher-rate code. In such a scheme, the variable-node degree distribution is constant for all the rates and thus is not always optimal.

In practical communication systems, hardware complexity of the encoder and decoder is an important issue for channel codes. The encoder and decoder implementation for LDPC codes are quite different from that of the turbo code. Turbo decoder uses the trellis structure of its constituent codes which provides simplified organization for the implementation. On the other hand, the belief-propagation LDPC decoder uses the Tanner graph of the underlying parity-check matrix. Certain constraints or structures can be applied to the parity-check matrix to facilitate the hardware implementation of LDPC encoder and decoder [45][46][47]. These LDPC parity-check matrices are designed with the hardware implementation in mind and thus are called hardware-aware LDPC codes.

The rest of the chapter is organized as follows. Section 4.2 explains the proposed rate-compatible scheme using row-combining. Section 4.3 provides a brief introduction of the irregular partitioned permutataion (IPP) codes which we

think is an efficient and flexible architecture. IPP codes allow parallel processing and can reduce the memory requirement. In Section 4.4, the hardware structure of the encoder and decoder of RC-IPP codes are proposed by making small modifications to the IPP decoder. Section 4.5 presents an LDPC design example for WLAN applications subject to various delay and hardware complexity constraints. Section 4.6 concludes the chapter.

4.2 Row Combining

A simple example in Section II of [41] describes the technique of row-combining. Assume that we rake a rate-1/2 mother LDPC matrix, $H_{\frac{1}{2}}$, as shown in Eq. (4.1).

$$H_{\frac{1}{2}} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}. \tag{4.1}$$

If we combine the first row with the fourth row, the second row with the fifth row, and the third row with the the sixth row, it results in a rate-3/4 LDPC parity-check matrix

$$H_{\frac{3}{4}} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}. \tag{4.2}$$

Similarly, more rows can be combined to form a even higher-rate LDPC code. For example, combining three rows into one row results in a rate-5/6 code.

The idea of row-combining is motivated by empirical evidences that the op-

timal average check node degree, d_c^*, is approximately inversely proportional to $1-R$ where $R = k/n$ is the rate of the code, k is the number of information bits and n is the blocklength. Therefore, for a fixed code length n, the number of one's, N_1, in the parity-check matrix is a constant.

$$\begin{aligned} N_1 &= (n-k) \times d_c^* && (4.3)\\ &\propto (n-k) \times \frac{1}{1-R} && (4.4)\\ &= (n-k) \times \frac{n}{n-k} && (4.5)\\ &= n. && (4.6) \end{aligned}$$

If the mother code parity-check matrix is carefully designed such that two ones representing the same variable node are never combined under any row-combining rules, the total number of one's remains constant. If each row of the new code results from combining an equal number of rows of the mother code matrix, the new check-node degree is concentrated, i.e., all the check nodes have the same degree. It is found that the performance of LDPC codes with a concentrated check-node degree perform better than those with a non-concentrated check-node degree. Therefore, the check-node degree is kept as concentrated as possible when designing the row-combining codes.

As for the variable-node degree distribution, row-combining results in a constant variable node degree distribution which obviously is not optimal for all the rates. In order to have low error floor properties, the number of degree-2 variable nodes cannot exceed the number of check nodes as shown in Section 3.2. Since the highest-rate code has the least number of check nodes, its optimal variable-node degree distribution is chosen as the mother code variable-node degree distribution. This disadvantage limits the performance of the lower-rate codes because their optimal variable-node degree distributions generally require more degree-2 nodes.

The graph-conditioning using ACE algorithm to avoid harmful short cycles can be applied to the row-combining rate-compatible scheme. In the ACE algorithm, the parity-check matrix is constructed one column (variable node) at a time. Each column generated randomly according to its degree is checked if it satisfies the the ACE constraint given the existing columns. The process repeats until all the columns are successfully generated. For the row-combining LDPC code design, the newly generated column not only has to pass the ACE constraints of the mother parity-check matrix, it must also satisfy the ACE constraints of all the other higher-rate matrices generated according to predefined row-combining rules.

4.3 Irregular Partitioned Permutation LDPC Codes

Among all the hardware architectures in the literature, we found that Hocevar's irregular partitioned permutation (IPP) structure [47] has several advantages over other structures including low memory requirements, simplified addressing, simplified routing, and high-degree of parallelism. It is also found that the proposed row-combining technique can be applied to the IPP structure with small modifications to the encoder and decoder implementation to design the row-combined IPP (RC-IPP) codes.

We will briefly introduce the IPP structure with focus on the design constraints it puts on the parity-check matrix. The constraints are summarized as follows.

(a) The parity-check matrix of IPP LDPC codes consists of $m \times m$ sub-matrices as the building blocks. Each sub-matrix is either an all-zero matrix or an identity matrix whose rows have been shifted by a set amount. Assume

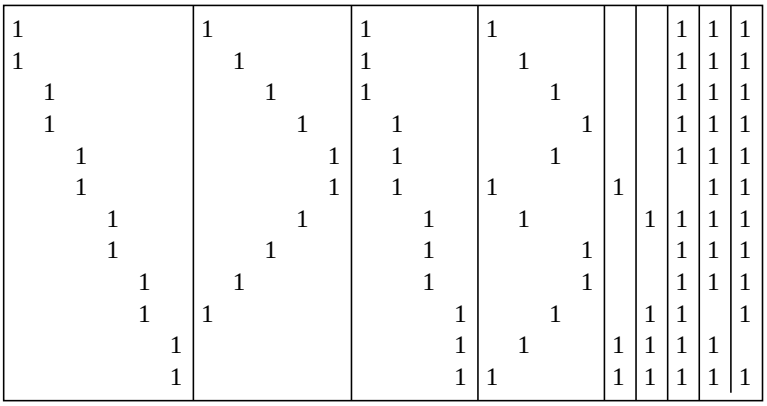


Figure 4.1: An example of column grouping where in the macro matrix, the columns form groups where in each group, there is at most one nonzero column per row. The number of column groups is also the number of column sum blocks (CSBs) in the IPP LDPC decoder (See Figure 4.2).

that H has j_M rows and n_M columns of blocks, the parity-check matrix has dimension of $(j_M \cdot m) \times (n_M \cdot m)$. The macro matrix, H_M, is a $j_M \times n_M$ matrix with entry being a "1" if the corresponding block is a shifted identity, and being a "0" if the corresponding block is an all-zero matrix. The maximum number of parallel row-processing is limited by the block size, m. Parallel processing can increase the throughput but it also requires more hardware.

(b) The degree distribution of the macro matrix is the same as that of the parity-check matrix because each non-zero entry represents a shifted identity matrix which has exactly one 1 per row and one 1 per column. The IPP structure requires the variable node degree and the check node degree to be less or equal to j_M and n_M respectively. Since the good irregular degree distributions usually require some high-degree nodes, this imposes an upper bound on the level of parallelization for IPP LDPC codes.

(c) A further constraint of the IPP structure is that the columns in the macro

matrix form column groups such that in each group there is at most one non-zero column per row. Figure 4.1 shows an example of a macro matrix whose columns form eight column groups. The purpose of forming columns groups is to reduce the number of bit update blocks, which more importantly reduces the number of individual memories.

The IPP decoder uses iterative belief-propagation algorithm described as follows. The log-likelihood of the variable-to-check message from variable node j to check node m is denoted as

$$L_{q_{mj}} = \sum_{i \in M(j), i \neq m} R_{ij} - 2\frac{r_j}{\sigma^2}, \tag{4.7}$$

where $M(j)$ is the set of check nodes connected to variable node j, r_j is the received channel information for variable node j, and σ^2 is the noise variance. R_{ij} denotes the check-to-variable node message from check node i to variable node j.

$$R_{ij} = -\left\{ \prod_{n \in N(i),\ n \neq j} \text{Sign}\left(L_{q_{in}}\right) \right\} \Psi\left(\sum_{n \in N(i),\ n \neq j} \Psi\left(L_{q_{in}}\right) \right) \tag{4.8}$$

where $N(i)$ is the set of variable nodes connected to the check node i and the function $\Psi(x) = \log(\tanh(\|x/2\|))$ is implemented using look-up tables (LUT).

The parallel decoder structure for IPP codes proposed in [5] consists of modules of R storage memory (RSM), parallel adder block (PAB), parity-check update block (PCUB), router, column sum block (CSB) and reserve router as shown in Figure 4.2. Note that p rows are processed in parallel and the columns form g column groups. Details of the blocks are provided in [47][5].

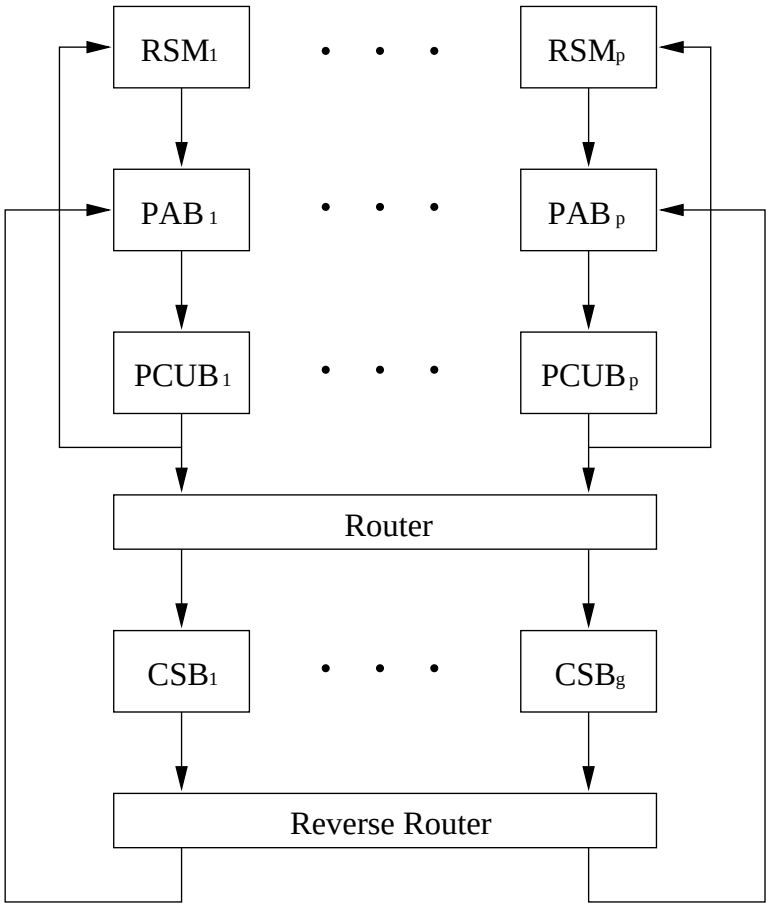


Figure 4.2: IPP LDPC decoder top level architecture with p parallel processing units and g column sum units.

4.4 Rate-Compatible IPP codes

In this section, we propose RC-IPP LDPC codes which has the IPP structure and achieves rate-compatibility using the technique of row-combining. Row-combining brings some additional constraints to the parity-check matrix. The decoder structure and blocks in Figure 4.2 must also be modified accordingly. The design procedures of RC-IPP codes are summarized as follows.

(i) Choose the code length, n, and block size, m.

(ii) Choose the target code rates and row-combining rules.

(iii) Design the parity-check matrix for the lowest-rate code using the optimal degree distribution of the highest-rate code.

(iv) Generate lower-triangular macro matrix and design interleavers such that column groups are formed for the IPP decoder. The macro matrix should also satisfy the condition that no bi-diagonal sub-matrices are formed for any of the row-combining rule.

(v) Randomly generate the shift values of the sub-matrices column by column and the resulting parity-check matrix must satisfy certain ACE constraints at each rate according to the row-combining rules.

4.4.1 Encoder Structure

Usually, the hardware implementation complexity of of LDPC codes focuses on the decoder because the encoder complexity is much lower than the decoder complexity. Still, simple encoder with the complexity grows linearly with the code length is desired. IRA codes [33] has a low-complexity encoder by forcing

the parity section of H, i.e., H_2 in Eq. (3.1), to be a staircase matrix as shown in Eq. (3.2). The encoder then consists of a sparse matrix multiplication followed by an accumulator. Richardson [34] developed an algorithm to build efficient encoders based on swapping rows and columns to form an approximate lower triangular parity-check matrix.

For the parity-check matrix with block structure, we use a combination of the two methods where the macro matrix is lower-triangular and the bottom-right sub-matrix is a staircase matrix. Using back-substitution, the parity bits can be solved sequentially from top to bottom. For the last block row, the remaining parity bits are calculated using exactly the same way as the IRA encoder. This structure will have one degree-1 variable node. If row-combining is performed at the macro matrix level properly, the low-triangular structure is preserved for all the higher-rate codes such that all the resulting codes can be encoded using back-substitution and sparse matrix multiplication followed by an accumulator.

4.4.2 Decoder Structure

The decoder structure of the RC-IPP LDPC codes follows that of the single-rate IPP LDPC code proposed in [5] with minor modifications. The basic idea is to use the decoder corresponding to the lowest-rate IPP code and reuse as much hardware as possible when performing row-combining. When doing the check-node update of the row-combined codes, the combined rows must be processed at the same time. So, instead of processing p consecutive rows in the IPP code decoder, the p parallel parity-check update block (PCUB) process p/N_{rc} rows from each of the N_{rc} sub-matrices that are combined in the current code. Since the RC-IPP LDPC code is designed such that no 1's are combined in the macro matrix level (i.e. no bi-diagonal sub-matrix is allowed), no memory conflict will

occur when the parity-check update is performed in such an order.

One problem for this scheme is the router and reverse router. In [5], the function of the router and reverse router is to assign the parity-to-variable and variable-to-check messages to the correct column groups. Each block row may require different routing depending on how the column group are formed. This can cause trouble for the RC-IPP decoder when updating rows coming from different block rows. A solution to this problem is to form complete column groups. By complete, we mean that in the macro matrix, each group has exactly one non-zero row per column (Note that the original requirement is at most one non-zero row per column). If all the column groups are complete, the i^{th} message always goes to the i^{th} column group and thus the router and the reverse router can be eliminated. If the total number of 1's in the macro matrix is not multiples of the number of rows, the degree distribution will be tweaked such that complete column groups can be formed for each column group.

The modified PCUB for RC-IPP decoder is shown in Figure 4.3. Each PCUB has two additional inputs of the sign and magnitude of the sum of $L_{q_{mn}}$ from other rows and also outputs the sign and magnitude of its own sum of $L_{q_{mn}}$ to other rows. Additional circuit are used to calculate and select the message S_{in} and M_{in} from other PCUBs. These circuits and connections between PCUBs are determined by the row-combining rules. For example, Figure 4.4 shows the connection of PCUB_1 assuming that it takes messages from PCUB_2 when combining every two rows and takes messages from both PCUB_2 and PCUB_3 when combining every 3 rows. A good row-combining rule should reuse the same combination as often as possible which not only reduces the PCUB connection complexity but also makes graph-conditioning easier.

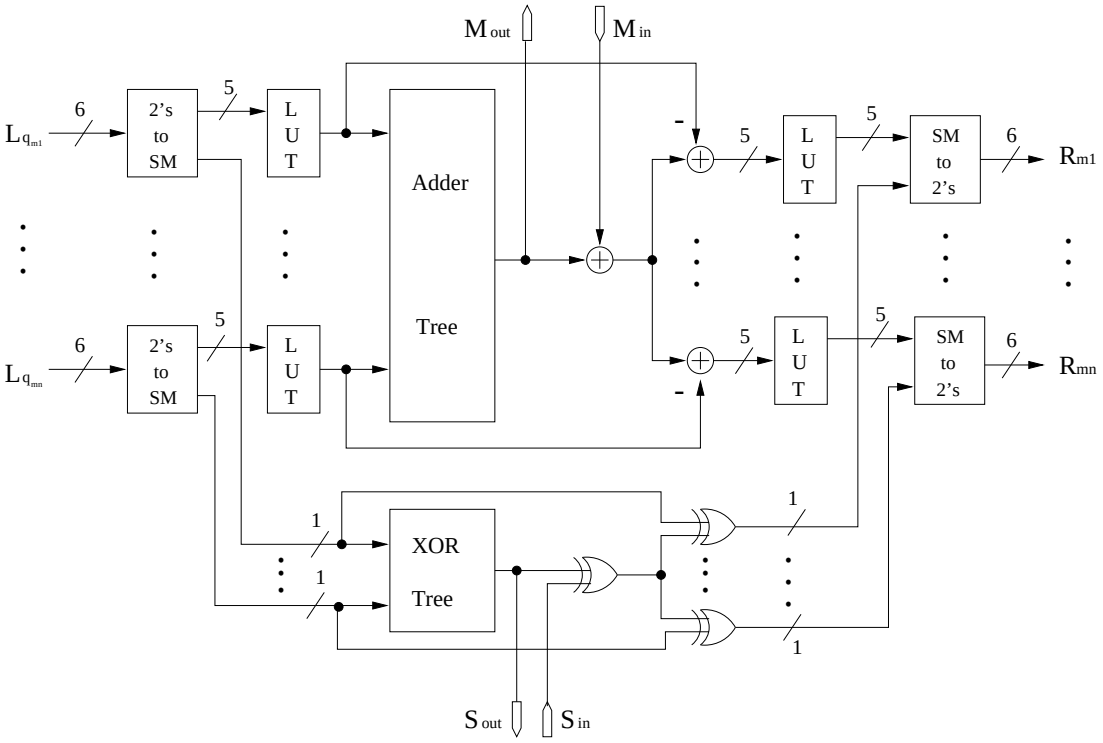


Figure 4.3: The architecture of parity-check update block (PCUB) for an RC-IPP LDPC code. This is modified from the modules proposed in [5] such that it can output LLRs to other PCUBs and also take inputs from them.

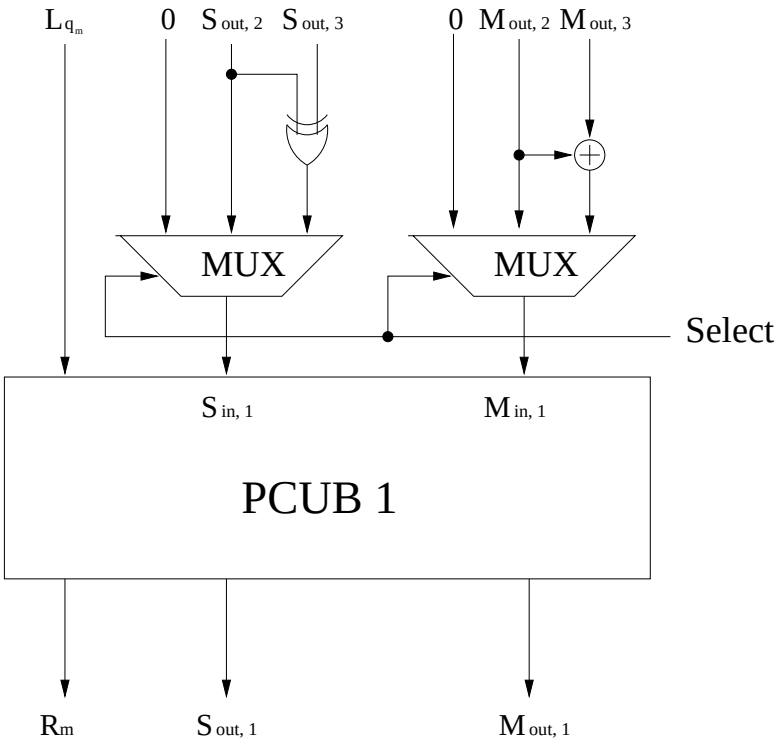


Figure 4.4: The connection of $PCUB_1$ to $PCUB_2$ and $PCUB_3$ assuming that it takes messages from those two PCUBs when performing row-combining.

4.5 Example RC-IPP LDPC Code Design

Following the procedures in Section 4.4, we designed RC-IPP LDPC codes for wireless LAN applications subject to various delay, throughput, and hardware complexity constraints. The code length is a medium-length 1944 bits with the sub-matrix size of 27×27. The target code rate is in the high-rate range between 1/2 and 7/8. Therefore, the macro matrix for the lowest-rate code ($R = 1/2$) is a 36×72 matrix. The row-combining rules are described as follows

(1) Rate-2/3: Combine block rows 1-12 with rows 19-30.

(2) Rate-3/4: Combine block rows 1-18 with rows 19-36.

(3) Rate-5/6: combines block rows 1-12, 13-24, and 25-36 together.

The highest-rate for this application is 5/6 which has 12 block rows. So, the maximum variable node degree cannot exceed 12. The row-combining rule imposes another constraint on the maximum variable node degree. Note that for all the positions subject to row-combining in any of the rule, at most one can be nonzero. For the proposed row-combining rules, every four positions form a set, e.g. $\{1, 19, 13, 25\}$, where at most one of them can be nonzero. Thus the maximum variable node degree is $36/4 = 9$. Using the constrained optimal degree distribution for the rate-5/6 code and limiting the possible variable node degree to be 2, 3, and 9, the macro matrix, H_M, of the rate-1/2 code consists of 14 columns with degree 9, 46 columns with degree 3, and 12 columns with degree 2.

By ordering the column in descending column degrees and assign the low-degree columns to the parity section of H_M, we can form a lower-triangular H_M that is suitable for the low-complexity encoder presented in Section 4.4.1.

Table 4.1: The column groups of the proposed RC-IPP codes. Every group except the last group has exactly 36 nonzero blocks and there is exactly one block for each row of H_M^* in a group. This complete group constraint eliminates the requirement of router and reverse router in the IPP LDPC decoder which enables the parallel processing of RC-IPP decoder.

Column Group	(1)	(2)	(3)	(4)	(5)	(6)	(7)		(8)				(9)
Column degrees	9	9	9	3	3	3	2	3	2	3	4	9	2
No. of columns	4	4	4	12	12	12	6	8	4	2	1	2	1

As for the decoder, the columns are interleaved such that they form complete column groups where every group has exactly one 1 per column and one 1 per row. First we slightly tweak the variable node degree by converted a degree-2 column into a degree-4 column, and then the complete column group can be formed for this tweaked degree distribution as shown in Table 4.1. Note that the sum of the column degrees in each group is exactly 36 which is equal to the number of rows. The only exception is the 9^{th} column group which consists of only the staircase block. Figure 4.5 illustrates that the encoder uses the lower-triangular H_M and by interleaving the columns, the column-grouped H_M^* is suitable for the decoder implementation.

Now that the macro matrix is fixed, shift values are generated randomly for one block column at a time. If the resulting columns satisfy certain ACE parameters for all the possible rates subject to predefined row-combining rules, we can proceed to the next block column. Otherwise, another set of shift values will be generated. This step repeats until all the shift values are successfully generated. Slightly smaller ACE parameters can be met compared to unstructured stand-alone codes or structured single-rate codes which means inferior graph connectivity for the RC-IPP codes because of the increased constraints. However,

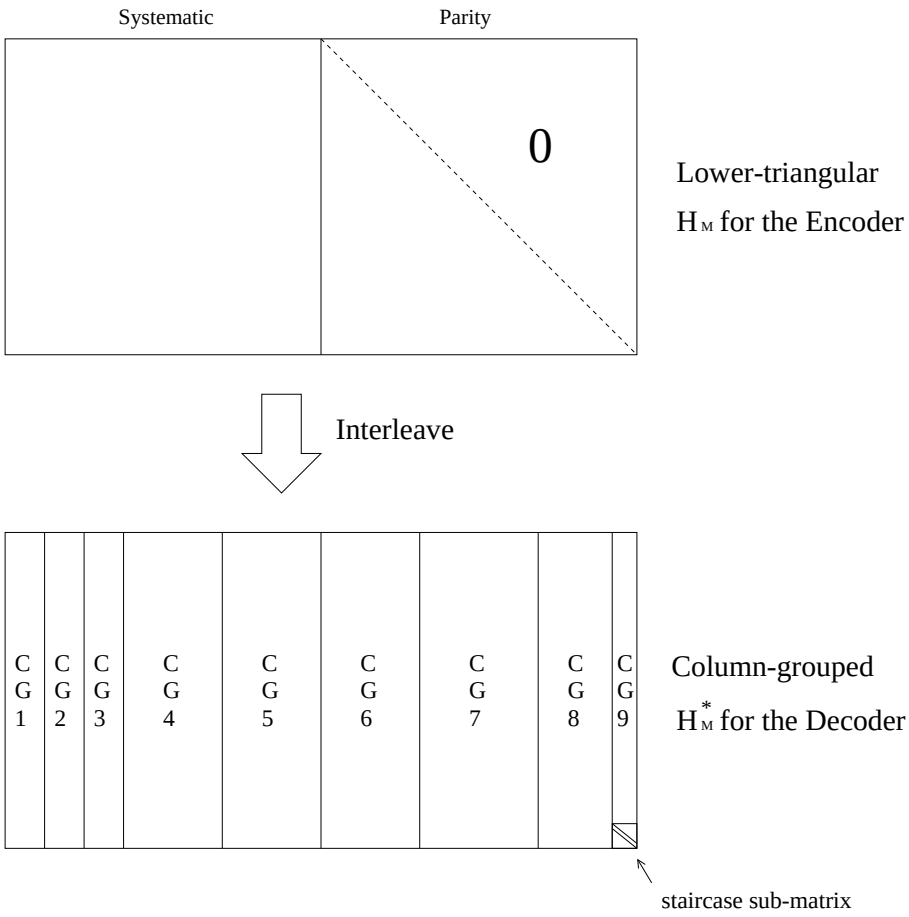


Figure 4.5: The encoder uses the lower-triangular H_M which facilitates the encoder implementation. Note that the low-degree columns are assigned to the parity parts of the matrix. By interleaving the columns of H_M (in the macro matrix level), the columns of H_M^* form complete groups and is thus suitable for the flexible decoder implementation. The variable node degree distributions for columns in each column group are listed in Table 4.1.

no error floor is observed for the proposed codes until FER=10^{-3} which means that the ACE algorithm is still very effective in lowering the error floors of the RC-IPP codes.

Simulation results in Figure 4.6 compares the FER of the proposed length-1944 RC-IPP LDPC code and the unstructured stand-alone codes for 10 iterations. If is found that these codes perform well under these conditions of block-length and number of iterations. Note that the non-optimality of the degree distribution is not noticed at the lower rates because the number of iteration is small. As pointed out in [41], when the number of iteration is increased to 50, the lower-rate row-combining codes suffer performance loss. The gap between the stand-alone code and the RC-IPP code is reported to be the largest (about 0.3dB) at rate-1/2.

4.6 Conclusions

In this chapter, we proposed the technique of row-combining to achieve rate-compatibility for LDPC codes. Row-combining has an advantage over puncturing or code-shortening because the effective blocklength code of the LDPC code remains constant. On the other hand, row-combing also has an disadvantage of constant variable node degree which is non-optimal. However, simulation results show that the performance loss is only marginal and the overall performance is very close to that of the stand-alone LDPC codes. Moreover, the graph-conditioning technique, ACE algorithm, works well with row-combining and thus low error floors are guaranteed for each of the target rates.

Rate-compatible scheme is only meaningful if it changes the code rate while maintaining an affordable increase of the hardware complexity. Practical LDPC

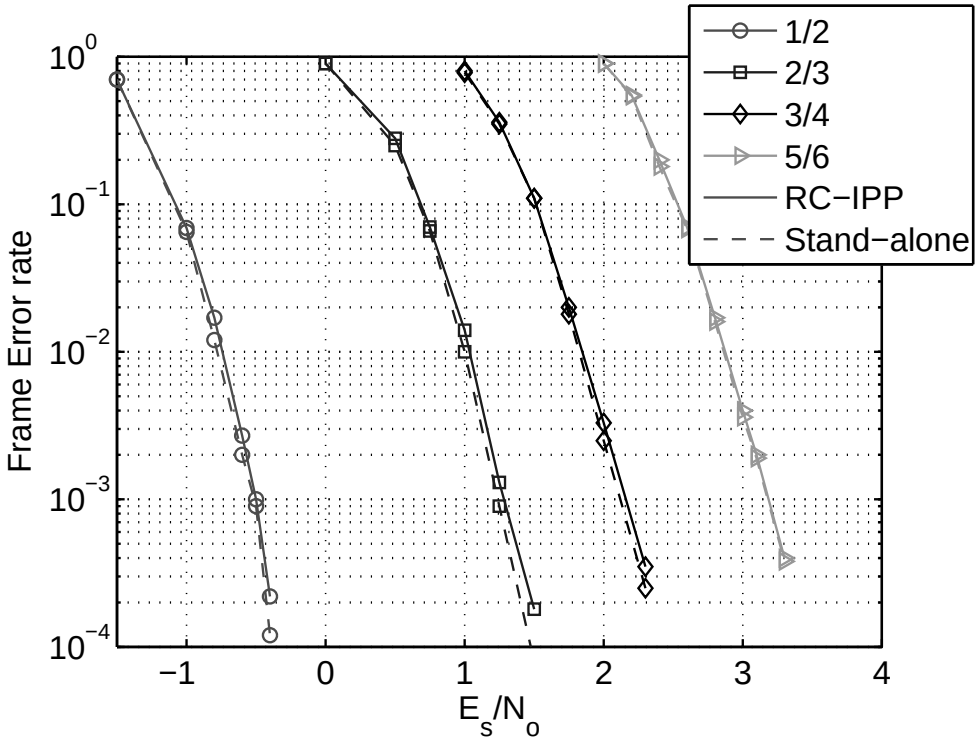


Figure 4.6: Frame-error-rate of the proposed RC-IPP LDPC codes at rate-1/2, rate-2/3, rate-3/4, and rate-5/6 for 10 decoder iterations. The FER of unstructured stand-alone codes at each rate is also plotted for reference.

code design are subject to structural constraints in order to facilitate the hardware implementation. It is found that the IPP LDPC codes has flexible and efficient hardware implementation, and its decoder structure is suitable for decoding the row-combining LDPC codes.

We proposed design procedures to design RC-IPP LDPC codes. The encoder of the RC-IPP code requires the macro matrix to be low-triangular and the encoding process uses back-substitution, sparse matrix multiplication followed by an accumulator. On the decoder side, an interleaver is used to rearrange the block columns into complete column groups such that it satisfies the efficient IPP decoder constraints. Some modifications are made to blocks of the decoder of the IPP LDPC code. With these modifications, the decoder can process all the combined rows in parallel and thus achieve rate-compatibility with small increase of hardware complexity of a single-rate IPP code decoder. Despite all these constraints introduced by the IPP code and row-combing, graph-conditioning can still be applied effectively to guarantee low error floors.

APPENDIX

The derivation of the approximate average EMI with TVLT in Eq. (2.18) is as follows. Assume that α, β, and γ are uniformly distributed over $[0, 2\pi)$ and use the EMI model in Eq. (2.17),

$$\begin{aligned}
& \mathrm{EMI}^*(\phi,\theta) \\
\approx\ & \frac{1}{N_\alpha N_\beta N_\gamma}\sum_\alpha\sum_\beta\sum_\gamma \mathrm{EMI}\left(\tilde{\phi},\tilde{\theta}\right) && (.9)\\
\approx\ & \mathrm{EMI}_{\min}+\frac{K_1}{N_\alpha N_\beta N_\gamma}\sum_\alpha\sum_\beta\left[1-\cos\left(4\tilde{\phi}\right)\right]\sum_\gamma\left[K_2+\cos\left(2\tilde{\theta}\right)\right] && (.10)\\
=\ & \mathrm{EMI}_{\min} \\
& +\frac{K_1}{N_\alpha N_\beta N_\gamma}\sum_\alpha\sum_\beta\left[1-\cos\left(4\tilde{\phi}\right)\right]\sum_\gamma\left[K_2+\cos\left(2\tilde{\theta}_R+2\gamma\right)\right] && (.11)\\
=\ & \mathrm{EMI}_{\min}+\frac{K_1K_2}{N_\alpha N_\beta}\sum_\alpha\sum_\beta\left[1-\cos\left(4\tilde{\phi}\right)\right]. && (.12)
\end{aligned}$$

By Eq. (2.13), $\cos\left(2\tilde{\phi}\right)=\cos(2\phi)\cos(2\alpha)-\sin(2\phi)\sin(2\alpha)\cos(\theta+\beta)$. Then,

$$\begin{aligned}
& 1-\cos\left(4\tilde{\phi}\right) \\
=\ & 2-2\cos^2\left(2\tilde{\phi}\right) && (.13)\\
=\ & 2-2\cos^2(2\phi)\cos^2(2\alpha)-2\sin^2(2\phi)\sin^2(2\alpha)\cos^2(\theta+\beta) \\
& +4\cos(2\phi)\sin(2\phi)\cos(2\alpha)\sin(2\alpha)\cos(\theta+\beta) && (.14)\\
=\ & \frac{5}{4}-\frac{1}{4}\cos(4\phi)-\frac{1}{4}\cos(4\alpha)-\frac{1}{4}\cos(2\theta+2\beta) \\
& -\frac{3}{4}\cos(4\phi)\cos(4\alpha)+\frac{1}{4}\cos(4\alpha)\cos(2\theta+2\beta) \\
& +\frac{1}{4}\cos(4\phi)\cos(2\theta+2\beta)-\frac{1}{4}\cos(4\phi)\cos(4\alpha)\cos(2\theta+2\beta) \\
& +\sin(4\phi)\sin(4\alpha)\cos(\theta+\beta)\,. && (.15)
\end{aligned}$$

The first two terms are constant over α and β while the averages of all the other terms are zero for uniform α and uniform β. Therefore,

$$\mathrm{EMI}^*(\phi,\theta)\approx\mathrm{EMI}_{\min}+\frac{K_1K_2}{4}\cdot[5-\cos(4\phi)]\,. \qquad (.16)$$

Note that a relaxed sufficient condition for Eq. (.16) and (.12) to hold is

$$\sum_{\alpha} e^{j4\alpha} = 0, \tag{.17}$$

$$\sum_{\beta} \cos(2\theta + 2\beta) = 0, \tag{.18}$$

$$\sum_{\gamma} \cos(2\gamma) = 0, \tag{.19}$$

which motivates the S-TVLT.

References

[1] C. Jones, T. Tian, A. Matache, R. D. Wesel, and J. Villasenor. Robustness of LDPC Codes on Periodic Fading Channels. In *IEEE Globecom*, Nov. 2002.

[2] A. Stefanov and T. M. Duman. Turbo-coded modulation for systems with transmit and receive antenna diversity over block fading channels: system model, decoding approaches, and practical considerations. *IEEE Journal on Sel. Areas in Comm.*, 19(5):958–968, May 2001.

[3] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara. Serial Concatenation of Interleaved Codes: Performace Analysis, Design and Iterative Decoding. *IEEE Trans. on Inform. Theory*, 44(3):909–926, May 1998.

[4] W. Shi, C. Komninakis, B. Daneshrad, and R. D. Wesel. Robustness of Space-Time Turbo Codes. In *ICC*, pages 1700–1704, Helsinki, Finland, June 2001.

[5] Y. Chen and D. E. Hocevar. A FPGA and ASIC Implementation of Rate 1/2, 8088-b Irregular Low Density Parity Check Decoder. In *IEEE Globecom*, volume 1, pages 113–117, Dec. 2003.

[6] C. Berrou and A. Glavieux. Near Optimum Error Correcting Coding and Decoding: Turbo-codes. *IEEE Trans. on Comm.*, 44(10):1261–1271, Oct. 1996.

[7] D. J. C. MacKay, S. T. Wilson, and M. C. Davey. Comparison of Constructions of Irregular Gallager Codes. *IEEE Trans. on Comm.*, 47(10):1449–1454, Oct. 1999.

[8] W. L. Root and P. P. Varaiya. Capacity of Classes of Gaussian Channels. *SIAM J. of Applied Math*, 16(6):1350–1393, 1968.

[9] I. Sutskover and S. Shamai. Iterative Decoding of Low-Density Parity-Check Codes Over Compound Channelsl. *IEEE Trans. on Comm.*, 54(2):308–318, Feb. 2006.

[10] C. Köse and R. D. Wesel. Universal Space-time Trellis Codes. *IEEE Trans. on Inform. Theory*, 49(10):2717–2727, Oct. 2003.

[11] C. Köse and R. D. Wesel. Universal Space-Time Codes from Demultiplexed Trellis Codes. *IEEE Trans. on Comm.*, 54(7):1243–1250, July 2006.

[12] R. D. Wesel, X. Liu, and W. Shi. Trellis Codes for Periodic Erasures. *IEEE Trans. on Comm.*, 48(6):938–947, June 2000.

[13] R. D. Wesel and J. M. Cioffi. Trellis Codes for Periodic Interleavers. *IEEE Comm. Letters*, 3(4):103–105, Apr. 1999.

[14] A. Matache and R. D. Wesel. Universal Trellis Codes for Diagonally Layered Space-Time Systems. *IEEE Trans. on Signal Processing*, 51(11):2773 –2783, Nov. 2003.

[15] S. ten Brink. Design of Serially Concatenated Codes Based on Iterative Decoding Convergence. In *Second International Symposium on Turbo Codes and Related Topics, Brest, France*, 2000.

[16] D. Divsalar, S. Dolinar, and F. Pollara. Iterative Turbo Decoder Based on Density Evolution. *IEEE Journal on Sel. Areas in Comm.*, 19(5):891–907, May 2001.

[17] T. Tian, C. Jones, J. Villasenor, and R. D. Wesel. Constructions of Irregular LDPC Codes with Low Error Floors. In *IEEE Int. Conf. on Comm.*, May 2003.

[18] H. El Gamal and A. R. Hammons. Analyzing the Turbo Decoder Using the Gaussian Approximation. *IEEE Journal on Sel. Areas in Comm.*, 47:672–686, Feb. 2001.

[19] C. Fragouli and R. D. Wesel. Semi-Random Interleaver Design Criteria. In *Communication Theory Symposium at Globecom 99, Rio de Juanero, Brazil*, Dec. 1999.

[20] D. Divsalar, S. Dolinar, and F. Pollara. Serial Concatenated Trellis Coded Modulation with Rate-1 Inner Code. In *Global Telecommunications Conference, GLOBECOM*, pages 777–782, 2000.

[21] H. M. Tullberg and P. H. Siegel. Serial Concatenated Trellis Coded Modulation with Inner Rate-1 Accumulate Code. In *Global Telecommunications Conference, GLOBECOM*, pages 936–940, 2001.

[22] X. Lin and R. S. Blum. Guidelines for Serially Concatenated Space-Time Code Design in Flat Rayleigh Fading Channels. In *IEEE Third Workshop on Signal Processing Advances in Wireless Communications*, Mar. 2001.

[23] D. Tujkovic. High bandwidth efficiency space-time turbo coded modulation. In *ICC*, June 2001.

[24] Y. Hong, J. Yuan, Z. Chen, and B. Vucetic. Space-time turbo trellis codes for two, three, and four transmit antennas. *IEEE Trans. on Veh. Tech.*, 53(2):318–328, Mar. 2004.

[25] Y. Liu, M. P. Fitz, and O. Y. Takeshita. Full rate space-time turbo codes. *IEEE Journal on Sel. Areas in Comm.*, 19(5):969–980, May 2001.

[26] L. Zheng and D. Tse. Diversity and Multiplexing: A Fundamental Tradeoff in Multiple Antenna Channels. *IEEE Trans. on Inform. Theory*, 49(5):1073–1096, May 2003.

[27] L. Zheng and D. Tse. Diversity-Multiplexing Tradeoff in Multiple-Access Channels. *IEEE Trans. on Inform. Theory*, 50(9):1859–1874, Sept. 2004.

[28] S. Tavildar and P. Viswanath. Approximately Universal Codes over Slow Fading Channels. *To appear in IEEE Trans. on Inform. Theory*, 2006.

[29] D. Tse and P. Viswanath. *Fundamenals of Wireless Communication*. Cambridge University Press, 2005.

[30] C. Köse, W. Weng, and R. D. Wesel. Serially Concatenated Trellis Coded Modulation for the Compound Periodic Erasures Channel. In *ICC*, Anchorage, AK, May 2003.

[31] B. Clerckx, D. Vanhoenacker-Janvier, C. Oestges, and L. Vandendorpe. Robust signal constellations for spatial multiplexing in the presence of real fading propagation channels. In *ICC*, Anchorage, AK, May 2003.

[32] W. Firmanto, Z. Chen, B. Vucetic, and J. Yuan. Design of space-time turbo trellis coded modulation for fading channels. In *IEEE Globecom*, Nov. 2001.

[33] M. Yang and W. E. Ryan. Lowering the Error-Rate Floors of Moderate-Length High-Rate Irregular LDPC Codes. In *IEEE International Symposium on Information Theory, Yokohama, Japan*, June 2003.

[34] T. Richardson, A. Shokrollahi, and R. Urbanke. Design of capacity-approaching irregular low-density parity check codes. *IEEE Trans. on Inform. Theory*, 47:619–637, Feb 2001.

[35] M. Yang, W. E. Ryan, and Y. Li. Design of efficiently encodable moderate-length high-rate irregular LDPC codes. *IEEE Trans. on Comm.*, 52(4):564–571, Apr. 2004.

[36] D. M. Mandelbaum. An adaptive feed-back coding using incremental redundancy. *IEEE Trans. on Inform. Theory*, 20:388–389, May 1974.

[37] J. B. Cain Jr, G. C. Clark, and J. M. Geist. Punctured convolutional codes of rate $(n-1)/n$and simplified maximum likelihood decoding. *IEEE Trans. on Inform. Theory*, 25:97–100, Jan. 1979.

[38] J. Hagenauer. Rate-compatible punctured convolutional codes (RCPC codes) and their applications. *IEEE Trans. on Comm.*, 36:389–400, Apr. 1988.

[39] J. Ha, J. Kim, and S. W. McLaughlin. Rate-compatible puncturing of low-density parity-check codes. *IEEE Trans. on Inform. Theory*, 50(11):2824–2836, Nov. 2004.

[40] T. Tian, C. Jones, and J. Villasenor. Rate-compatible low-density parity-check codes. In *ISIT 2004, Chicago*, July 2004.

[41] A. I. Vila Casado, W. Weng, and R. D. Wesel. Multiple-Rate Low-Density Parity-Check Codes with Constant Blocklength. *Submitted to IEEE Trans. on Comm.*, 2006.

[42] Y. Mao and A. H. Banihashemi. A heuristic search for good low-density parity-check codes at shortblock lengths. In *IEEE Int. Conf. on Comm.*, Jun. 2001.

[43] D. M. Arnold, E. Eleftheriou, and X. -Y. Hu. Progressive edge-growth Tanner graphs. In *IEEE Globecom*, Nov. 2001.

[44] A. Ramamoorthy and R. D. Wesel. Analysis of an Algorithm for Irregular LDPC Code Construction. In *IEEE International Symposium on Information Theory*, Chicago, USA, June 2004.

[45] T. Zhang and K. K. Parhi. VLSI implementation-oriented (3,k)-regular low-density parity-check codes. In *IEEE Workshop on Signal Processing Systems*, pages 25–36, Sep. 2001.

[46] E. Boutillon, J. Castura, and F. R. Kschischang. Decoder-first code design. In *Int'l Symp. on Turbo Codes and Related Topics,*, pages 459–462, Sep. 2000.

[47] D. E. Hocevar. LDPC code construction with flexible hardware implementation. In *IEEE Int. Conf. on Comm.*, volume 4, pages 2708–2712, May 2003.