Repeat-Zigzag-Hadamard Codes based on the EXIT Functions of Hadamard Codes

Kai Li* Xiaodong Wang* Alexei Ashikhmin†

Abstract

We investigate the extrinsic information transfer (EXIT) functions of Hadamard codes in the context of repeat-zigzag Hadamard (RZH) codes with parallel decoding. EXIT functions over both the binary erasure channel (BEC) and the binary-input additive white Gaussian noise (BIAWGN) channel are derived. The application of these EXIT functions to the design of low-rate capacity-approaching irregular RZH (IRZH) codes is also considered. Using the EXIT functions, the bit error rate (BER) for a given code profile can be easily estimated and the differential evolution (DE) technique can be employed to find the optimal degree profile for given design parameters. The EXIT functions derived in this paper can serve as an effective tool for designing low-rate IRZH codes with parallel decoding in both BEC and BIAWGN channels.

Keywords: EXIT functions, Hadamard codes, low-rate codes, irregular code design.

*Dept. of Electrical Engineering, Columbia Univ., New York, NY 10027. ({likai,wangx}@ee.columbia.edu)

†Bell Laboratories, Lucent Technologies, Murray Hill, NJ 07974. (aea@bell-labs.com)

1 Introduction

For communication systems operating in the low signal-to-noise ratio (SNR) regime (e.g., code-spread communication systems and power-limited sensor networks), low-rate coding schemes play a critical role. Traditional low-rate channel coding schemes include Hadamard codes [12, 24] and super-orthogonal convolutional codes [23]. These codes offer low coding gain and hence their performance is far away from the ultimate Shannon limit[1]. Turbo-code-based low-rate concatenated coding schemes have been proposed in [3, 8] which offer higher performance but they also incur higher complexities due to the complex trellis structures that specify the codes. The low-density parity-check (LDPC) codes [4] and the repeat-accumulate (RA) codes [7] offer capacity-approaching capability for various code rates when the ensemble profiles are optimized. However, in the low-rate region, they both suffer from significant performance loss and extremely slow convergence with iterative decoding. On the other hand, Hadamard codes have been shown to be a useful tool for improving the code performance in the low-rate region. Specifically, constructed from Hadamard code arrays, low-rate turbo-Hadamard codes [14] offer a bit-error-rate (BER) of 10^{-5} at $E_b/N_0 = -1.2$ dB, which is only around 0.4 dB away from the Shannon limit. More recently, built on zigzag codes [13], parallel-concatenated zigzag-Hadamard (PCZH) codes [9] and repeat-zigzag-Hadamard (RZH) codes [11] have been proposed, both of which exhibit much simpler encoder and decoder structures while still offering a similar performance as that of the turbo-Hadamard codes.

Similar to LDPC codes and RA codes, an RZH code can be represented by its Tanner graph, based on which iterative decoding algorithms can be derived. In [11] two belief-propagation (BP)-based decoding schemes, namely, the turbo-like serial decoding and the LDPC-like parallel decoding, are proposed. Although simulations have shown that for reg-

[1]The ultimate Shannon capacity of an additive white Gaussian noise (AWGN) channel in terms of SNR per information bit is about -1.6 dB for codes with rates approaching zero.

ular RZH codes, both decoding schemes work well and offer a similar BER performance, the LDPC-like decoding has certain advantage over the turbo-like decoding as follows: 1) its parallel nature is preferable in terms of hardware implementation; 2) its inner decoder is a Hadamard decoder, a fact that makes it possible to derive analytical expression of the extrinsic information transfer (EXIT) function [20, 21] for the component decoder, which may significantly ease the irregular code design. To elaborate the second point, we know that the parameters of an RZH code need to be chosen carefully for capacity-approaching performance, which leads to the irregular RZH (IRZH) code with optimized degree profile. As in [11], for turbo-like decoding, the EXIT chart technique can be employed to optimize the code profiles. However, since in this case the inner decoder is a trellis-based zigzag-Hadamard (ZH) decoder, an analytical expression of its EXIT function is not available, and we need to resort to Monte Carlo simulations to obtain the EXIT functions of the inner code. On the other hand, for LDPC-like parallel decoding, the connection bits of the ZH code are treated as degree-2 variable nodes, and the inner constituent decoder becomes a Hadamard decoder, having the channel observations of the parity bits and the *a priori* probabilities about the information/connection bits as the input, and the extrinsic information for the information/connection bits as the output. EXIT functions with multiple inputs and outputs are difficult to obtain using Monte Carlo simulations. In this paper, by extending the techniques introduced in [2, 17], we derive the EXIT functions of Hadamard codes with multiple inputs and outputs in the context of low-rate RZH codes over both BEC and BIAWGN channels, and apply these results to the design of IRZH codes. For BEC, exact analytical EXIT functions are given, whereas for BIAWGN channel the EXIT functions are approximated in the form of a series expansion of EXIT functions for single parity-check codes. With the EXIT functions, the BERs for a given code profile can be easily estimated and the differential evolution (DE) method [15] can be employed to optimize the degree profile in both channels.

The remainder of this paper is organized as follows. In Section 2, we introduce the RZH codec. In Section 3, we derive the EXIT functions of Hadamard code in BEC. In Section 4, we derive the EXIT functions over BIAWGN channel. In Section 5, the analytical EXIT functions are employed to optimize the IRZH code ensemble profiles and several design examples are given. Section 6 contains the conclusions.

2 Repeat-Zigzag Hadamard Codes

Hadamard Codes

A Hadamard codeword is obtained from a Hadamard matrix. Starting from $\underline{H}_1 = [+1]$, an $n \times n$ ($n = 2^r$, where r is called the order of the Hadamard code) Hadamard matrix $\underline{H}_n$ over $\{+1, -1\}$ can be constructed recursively as

$$\underline{H}_n = \begin{bmatrix} +\underline{H}_{\frac{n}{2}} & +\underline{H}_{\frac{n}{2}} \\ +\underline{H}_{\frac{n}{2}} & -\underline{H}_{\frac{n}{2}} \end{bmatrix}. \tag{1}$$

A length-2^r Hadamard codeword set is formed by the columns of the bi-orthogonal Hadamard matrix $\pm\underline{H}_n$, denoted by $\{\pm\underline{h}^j : j = 1, 2, \cdots, 2^r\}$, in a binary $\{0, 1\}$ form, i.e., $+1 \to 0$ and $-1 \to 1$. Each Hadamard codeword carries $(r + 1)$ bits of information. In systematic encoding, the bit indices $\{1\} \cup \{2^{k-1} + 1, k = 1, \cdots, r\}$ are used as systematic bit positions of the length-2^r Hadamard code. The other $m = (2^r - r - 1)$ bits are parity bits. Denote $\mathcal{H}$ as a Hadamard encoder. Then the Hadamard encoding can be represented by

$$\underline{c} = \mathcal{H}\left(d_0, d_1, \cdots, d_r\right), \tag{2}$$

where $\underline{c}$ is the resulting codeword. Clearly, $\underline{c}$ is a column of either $+\underline{H}_n$ or $-\underline{H}_n$. Hereafter, for simplicity, we will drop the subscript n in $\underline{H}_n$.

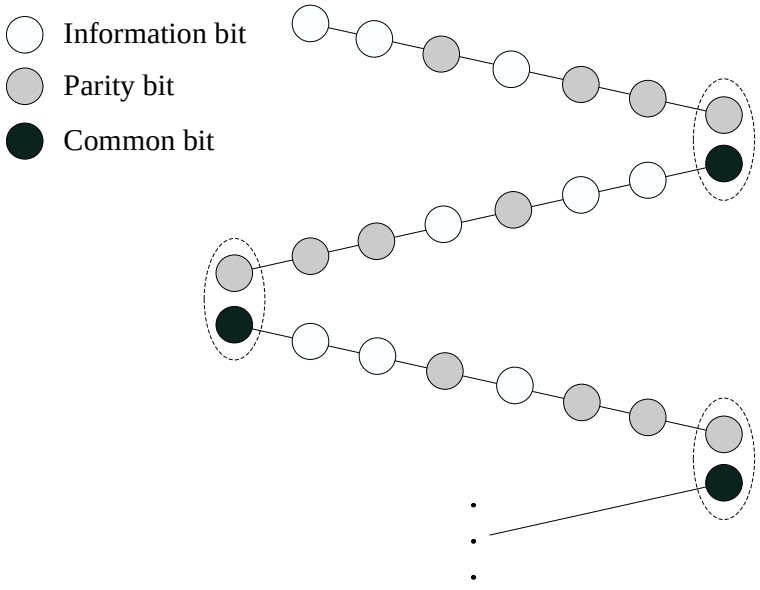


Figure 1: Graphical representation of an un-punctured systematic ZH code with $r = 3$. The first bit of the first segment is omitted.

Zigzag-Hadamard (ZH) Codes

As shown in Fig. 1, a zigzag-Hadamard (ZH) code [9] is described by a simple zigzag graph with each segment being a length-2^r Hadamard code. The data stream over $\{+1, -1\}$ is first segmented into blocks $\{\underline{d}_j\}_j$, where $\underline{d}_j = [d_{j,1}, d_{j,2}, \cdots, d_{j,r}]$, $j = 1, 2, \cdots, J$. As shown in Fig. 1, for a systematic ZH encoder, the last parity bit of the previous coded Hadamard segment is the first input bit to the Hadamard encoder for the current segment. The coded bits of the j-th segment are obtained by the Hadamard encoder with

$$\underline{c}_j = \mathcal{H}\left(c_{j-1,2^r}, d_{j,1}, d_{j,2}, \cdots, d_{j,r}\right), \tag{3}$$

where the codeword $\underline{c}_j = [c_{j,1}, c_{j,2}, \cdots, c_{j,2^r}]^T$, with $c_{j,1} = c_{j-1,2^r}$ being the *common bit* or the *connection bit* that connects the current segment to the previous segment and forms the zigzag structure, $c_{j,2^{i-1}+1} = d_{j,i}$, $i = 1, 2, \cdots, r$, being the systematic bits, and all other $(2^r - r - 1)$ bits being the parity bits. Denote $q_j \triangleq c_{j,1}$ and let $\underline{p}_j$ represent the parity bits, i.e., $\underline{p}_j \triangleq \{c_{j,i}, i \neq 1, i \neq 2^{\ell-1} + 1, \ell = 1, \cdots, r\}$. The j-th coded bits segment is then denoted

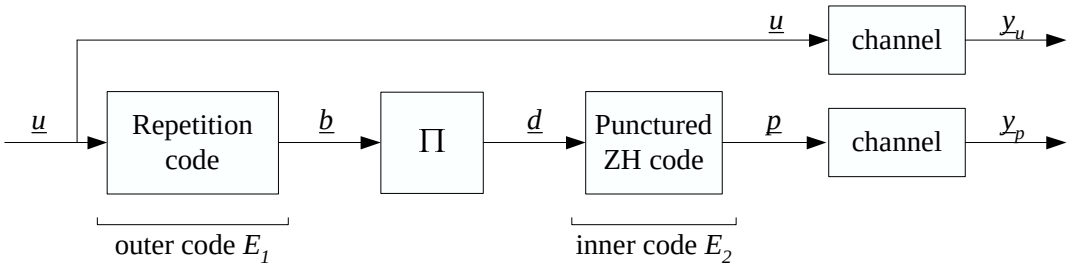


Figure 2: Systematical RZH encoder.

by a triplet $\underline{c}_j = \{\underline{d}_j, \underline{p}_j, q_j\}$, $j = 1, \cdots, J$, and the overall ZH code sequence is denoted by $\underline{c} = \{\underline{d}, \underline{p}, \underline{q}\}$. Obviously, the above ZH code is a *systematic* code. Note that the common bit of the first segment can be arbitrarily assigned and is assumed to be $+1$.

ZH codes are convolutional-like codes. A binary convolutional-like code is said to be *weight-recursive* if any information sequence with Hamming distance one will produce a codeword with Hamming weight $\to \infty$ when its length $\to \infty$. It is shown in [9] that a systematic ZH code with even Hadamard order r is weight-recursive, and a systematic ZH code with odd Hadamard order r is not weight-recursive. Consequently, in what follows, we only consider ZH codes with even Hadamard order.

Repeat-ZH Codes

In [11], we proposed an alternative code structure to parallel concatenated ZH codes and turbo-Hadamard codes, namely, the repeat-zigzag-Hadamard (RZH) codes. The structure of a systematic RZH code is shown in Fig. 2. An information bit block $\underline{u}$ of length k is encoded by an outer repetition encoder E_1 of rate $1/d_v$ into a codeword $\underline{b}$ of length $N = kd_v$, which is permuted by an interleaver Π of length N to form $\underline{d}$, and then encoded by an inner systematic recursive ZH encoder E_2. Since the outer code is a repetition code and the inner ZH code is a systematic code, we have d_v repetitions in the resulting ZH codeword for every information bit. Since a repetition code has no coding gain, it is desirable to transmit the information bits only once. Also notice that the common bit of a ZH segment is the same

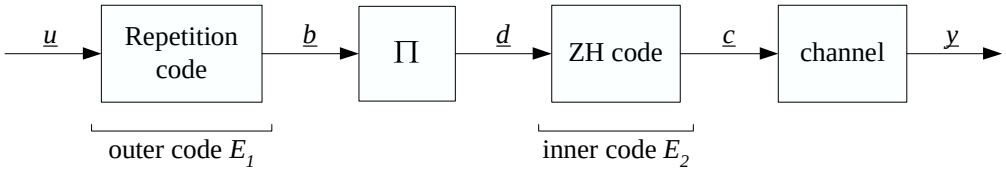


Figure 3: Un-punctured RZH encoder.

as the last parity bit of the previous Hadamard segment and hence introduces a degree-2 repetition code. As a consequence, we can puncture the last parity bit and only transmit the common bit. After puncturing, the inner code is a punctured ZH code with which we obtain the systematic RZH code as shown in Fig. 2, with the parity bits $\underline{p}$ being the output of the punctured ZH code. Let n be the length of the ultimate codeword $\underline{c} = \{\underline{u}, \underline{p}\}$, then the overall code is an (n, k) linear block code and the overall coding rate is given by

$$R_c = \frac{r}{d_v(2^r - r - 1) + r}, \tag{4}$$

where r is the order of the Hadamard code for the inner encoder.

In case that the inner code is not punctured, we have the un-punctured RZH code whose structure is shown in Fig. 3. For the un-punctured code, the overall code rate is given by $R_c = R_1 R_2$, where $R_1 = 1/d_v$ is the rate of the outer repetition code and R_2 is the rate of the inner ZH code. Since normally the first common bit of the ZH code is omitted, the overall rate of the un-punctured RZH code is given by

$$R_c = \frac{k}{\left(\frac{kd_v}{r}\right) 2^r - 1} = \frac{r}{d_v 2^r - r/k}. \tag{5}$$

With $k \to \infty$, we have $R_c \to \frac{r}{d_v 2^r}$. Note that as shown in [11], when the coding rate is very low, e.g., $R_c \leq 0.01$, the performance difference between the punctured one and the un-punctured code is relatively small; whereas in the range around $R_c = 0.05$, the punctured code offers a significant coding gain over the un-punctured one due to the elimination of the repetition bits embedded in the inner ZH codes.

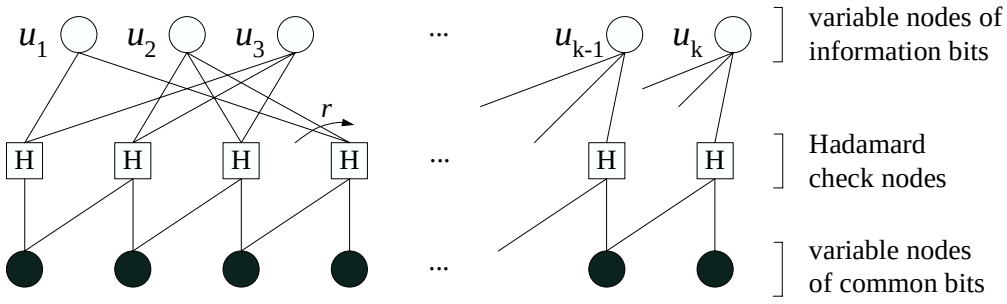


Figure 4: Tanner graph of an un-punctured (n, k) RZH code.

LDPC-Like Parallel Decoding of RZH Codes

Since the un-punctured RZH codes are standard serial concatenated codes, here we first introduce the decoding method for un-punctured codes, based on which the decoding method of RZH codes in Fig. 2 can be easily obtained. An un-punctured RZH code can be represented by its Tanner graph [18]. As shown in Fig. 4, there are three sets of nodes in the Tanner graph of an un-punctured RZH code: the *variable nodes of information bits*, corresponding to $\underline{u}$, the *variable nodes of common bits*, corresponding to $\underline{q}$, and the *Hadamard check nodes*, corresponding to the Hadamard code constraints. Note that the parity bits of Hadamard codewords are embedded in their Hadamard check nodes. The information bits that are repeated d_v times are represented by information bit variable nodes with degree d_v, since they participate in d_v Hadamard code constraints; and the common bit variable nodes can be viewed as degree-2 information bit variable nodes. Each Hadamard check node is connected to r information bit nodes and to two common bit nodes. The connections between Hadamard check nodes and information bit nodes are determined by the edge interleaver and are random, whereas the connections between Hadamard check nodes and common bit nodes are arranged in a highly structured zigzag pattern.

Based on the Tanner graph, an LDPC-like parallel decoding algorithm can be derived. For LDPC-like decoding, the common bit nodes are treated as degree-2 information bit nodes. In this case, all Hadamard check nodes and all information bit nodes are activated

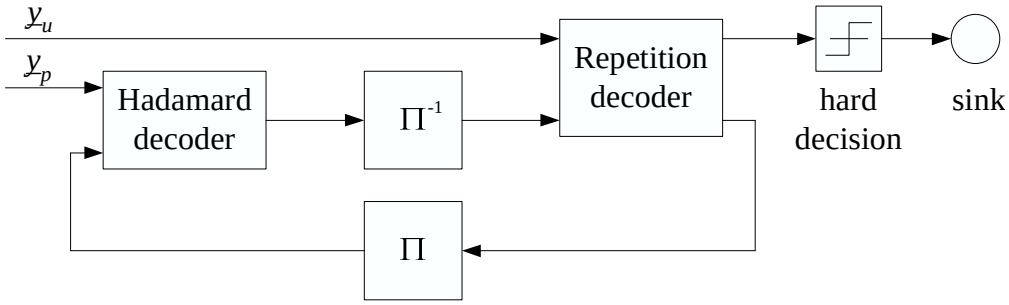


Figure 5: Parallel decoding of systematic RZH codes.

alternately and in parallel. Every time the Hadamard check nodes are activated, the J parallel Hadamard decoders firstly compute the *a posteriori* log-likelihood ratios (LLRs) of the information bits as in [14] based on the channel observation $\{\underline{y}_j\}_j$ and the *a priori* LLRs passed from the information bit nodes, with which the extrinsic LLRs of the information bits can be obtained by subtracting the *a priori* LLRs from the *a posteriori* LLRs. The resulting extrinsic LLRs are then fed to the information bit nodes as the *a priori* information, with which the information bit nodes are activated in parallel and the extrinsic LLRs are computed for the next iteration.

The decoding of systematic RZH codes is similar to that of the un-punctured case. The only difference is the treatment of the channel observation for the information bits $\underline{y}_u$. As shown in Fig. 2, the information bits are transmitted only once, then similar to the decoding of systematic RA codes, it is more convenient for $\underline{y}_u$ to participate in the repetition decoding. Fig. 5 depicts the structure of the LDPC-like decoder for systematic RZH codes. With such a structure, an information bit variable node of degree d_v has $d_v + 1$ incoming messages: d_v from the edge interleaver and one from the communication channel. The repetition decoder decodes by computing, for $i = 1, 2, \cdots, d_v$

$$L_{\text{rep}}^{\text{out}}(i) = L_{\text{ch}} + \sum_{j \neq i} L_{\text{rep}}^{\text{in}}(j), \tag{6}$$

where L_{ch} is the LLR value about the information bit from the channel. For BEC with

erasure probability ϵ_{ch}, we have $L_{\text{ch}} = \infty$ in case the bit is not erased, and $L_{\text{ch}} = 0$ in case it is erased. For AWGN channel with noise variance $\sigma_{\text{ch}}^2 = \frac{N_0}{2}$, we have $L_{\text{ch}} = \frac{2y_u}{\sigma_{\text{ch}}^2}$ where y_u is the channel observation of the BPSK modulated information bit u.

3 EXIT Functions on BEC

The EXIT functions are a convenient tool for analyzing and designing coding systems with iterative decoding. In this section, we derive the EXIT functions of Hadamard codes on BEC in the context of RZH codes with parallel decoding. The basic idea is to utilize the information function introduced in [6] to compute the average mutual information of the coded bits. In what follows, we first introduce the concept of split information function.

Information Function and Split Information Function

Given an (n, k) linear code $\mathcal{C}$, the information function [6] in h positions is defined as the average amount of information carried on the h positions of $\mathcal{C}$. More precisely, let $\underline{c} = [c_1, c_2, \ldots, c_n] \in \mathcal{C}$ be a codeword of $\mathcal{C}$ and $\mathcal{S}_h$ be the set of all subsets of $\{1, 2, \ldots, n\}$ of size h. Denote $\mathcal{S} = \{i_1, i_2, \ldots, i_h\} \in \mathcal{S}_h$. A subcode $\mathcal{C}_\mathcal{S}$ of $\mathcal{C}$ is defined by

$$\mathcal{C}_\mathcal{S} \triangleq \{\underline{c}_\mathcal{S} : \underline{c} \in \mathcal{C}\}, \quad \text{where} \quad \underline{c}_\mathcal{S} \triangleq [c_{i_1}, c_{i_2}, \ldots, c_{i_h}].$$

Let $\underline{G}$ be the generation matrix of $\mathcal{C}$ and $\underline{G}_\mathcal{S}$ be the generation matrix for $\mathcal{C}_\mathcal{S}$. Then the *unnormalized information function* [2] in h positions of $\mathcal{C}$ is given by

$$\tilde{e}_h \triangleq \sum_{\mathcal{S} \in \mathcal{S}_h} \kappa_\mathcal{S}, \tag{7}$$

where $\kappa_\mathcal{S}$ is the *dimension* of the subcode $\mathcal{C}_\mathcal{S}$ which equals to the rank of its generation matrix $\underline{G}_\mathcal{S}$. Given the generation matrix of a code, it is easy to compute the unnormalized information functions according to (7).

In case that different code bits has different prior information, we need to consider a simple generalization of $\tilde{e}_h$. Let $\mathcal{C}$ be an (n,k) linear code formed by L segments $\{\underline{c}_\ell\}_{\ell=1}^{L}$ where the length of $\underline{c}_\ell$ is n_ℓ with $\sum_{\ell=1}^{L} n_\ell = n$. Let $\mathcal{S}_{h_1,h_2,\ldots,h_L}$ be the set of all subsets of $\{1,2,\ldots,n\}$ of the form $\{i_{1,1}, i_{1,2}, \ldots, i_{1,h_1}; i_{2,1}, i_{2,2}, \ldots, i_{2,h_2}; \ldots; i_{L,1}, i_{L,2}, \ldots, i_{L,h_L}\}$ where $1 \le i_{1,1} < i_{1,2} < \cdots i_{1,h_1} \le n_1$, $n_1 + 1 \le i_{2,1} < i_{2,2} < \cdots i_{2,h_2} \le n_1 + n_2$, $\ldots$, $\sum_{\ell=1}^{L-1} n_\ell + 1 \le i_{L,1} < i_{L,2} < \cdots i_{L,h_L} \le \sum_{\ell=1}^{L} n_\ell = n$. In other words, $\mathcal{S}_{h_1,h_2,\ldots,h_L}$ is the set of subsets of choosing h_ℓ positions from the ℓ-th segment of $\mathcal{C}$, for $\ell = 1,2,\ldots,L$. Then the *unnormalized split information function* in $(h_1,h_2,\ldots,h_L)$ position of $\mathcal{C}$ is defined as

$$\tilde{e}_{h_1,h_2,\ldots,h_L} \triangleq \sum_{\mathcal{S}\in\mathcal{S}_{h_1,h_2,\ldots,h_L}} \kappa_{\mathcal{S}}. \tag{8}$$

Again, given the code generation matrix $\underline{G}$ and the division rule of the code segments, it is easy to obtain the generation matrix of its subcode $\underline{G}_{\mathcal{S}}$ and to compute the corresponding unnormalized split information function according to (8).

Next we will show that the EXIT functions of a Hadamard code in a ZH codeword over a BEC can be computed from its unnormalized split information functions.

EXIT Functions of Hadamard Codes

We first examine the division rule of the Hadamard code in a ZH code under parallel decoding. As shown in Fig. 1, a Hadamard segment in a ZH codeword consists of three smaller segments: the information bits $\underline{d}$ of length r, the parity bits $\underline{p}$ of length $m = n - r - 1$ where $n = 2^r$, and the connection bit q which connects the current segment to the previous one. Note that as seen in Fig. 1, q is the same as the last parity bit p_m of the previous Hadamard segment.

Now let us focus on the Hadamard decoder in the context of parallel decoding for RZH codes. Let $\underline{y}_p$ be the channel observation of the parity bits, and $\underline{y}_{p_{[m]}}$ be the observation exclusive of y_{p_m}. Also let $\underline{a}$ be the *a priori* LLRs of $\underline{d}$, v_0 be the *a priori* LLR of q. Note that since the last parity bit p_m is identical to the connection bit of the next segment, it is connected to the degree-2 connection bit variable node in the Tanner graph as depicted in

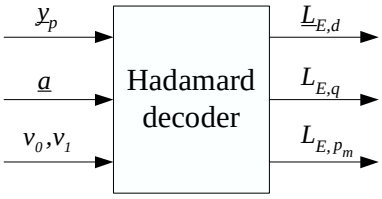


Figure 6: Inputs/Outputs of the Hadamard decoder for RZH codes with parallel decoding.

Fig. 4 and will have prior information passed from the degree-2 repetition decoder which can be denoted by v_1. As shown in Fig. 6, based on the channel observation of the parity bits, the prior information about the information bits, the connection bit and the last parity bit p_m, the soft-input soft-output (SISO) Hadamard decoder needs to compute the corresponding extrinsic LLRs for $\underline{d}$, q and p_m. Consequently, in order to perform EXIT analysis, we need to compute the average mutual information about the information bits $\underline{d}$

$$I_{E,d} = \frac{1}{r}\sum_{i=1}^{r} I(X_{d_i}; E_{d_i}), \tag{9}$$

where X_{d_i} and E_{d_i} are respectively the random variables associated with d_i and its extrinsic information of d_i. Letting X_q and X_{p_m} be the random variables associated with q and p_m, respectively, we also need the mutual information about the connection bits q and p_m which are respectively denoted as

$$I_{E,q} = I(X_q; E_q) \quad \text{and} \quad I_{E,p_m} = I(X_{p_m}; E_{p_m}), \tag{10}$$

where E_q and E_{p_m} are respectively the random variables associated with the extrinsic information of q and p_m.

Suppose that the communication channel for $\underline{y}_p$ is a BEC with erasure probability ϵ_y. For EXIT analysis, we also assume that the *a priori* LLRs about the information bits, connection bit and the last parity bit p_m are the outputs of the corresponding virtual BECs which are called the *extrinsic channels*. In other words, suppose that the mutual information between the *a priori* LLR E and the transmitted bit $X \in \{+1, -1\}$ is I_A, then the virtual extrinsic

channel is assumed to be a BEC with an erasure probability of $\epsilon = 1 - I_A$. Then the EXIT function of information bits $\underline{d}$ is given by the following result.

Theorem 3.1 *If the extrinsic and communication channels are BECs with respective erasure probabilities $\epsilon_a = 1 - I_{A,d}$, ϵ_y, $\epsilon_{v_0} = 1 - I_{A,q}$ and $\epsilon_{v_1} = 1 - I_{A,p_m}$, where $I_{A,d}$, $I_{A,q}$ and I_{A,p_m} are respectively the a priori mutual information about the information bits $\underline{d}$, connection bit q and the last parity bit p_m, then for a Hadamard segment of length $n = 2^r$ in a ZH codeword under parallel decoding we have*

$$\begin{aligned} &I_{E,d}^{BEC}(\epsilon_a, \epsilon_y, \epsilon_{v_0}, \epsilon_{v_1}) = \\ &\quad 1 - \frac{1}{r}\sum_{h=0}^{m-1}(1-\epsilon_y)^h \epsilon_y^{m-1-h} \sum_{j=0}^{1}(1-\epsilon_{v_0})^j \epsilon_{v_0}^{1-j} \sum_{k=0}^{1}(1-\epsilon_y\epsilon_{v_1})^k \epsilon_y^{1-k}\epsilon_{v_1}^{1-k} \sum_{g=1}^{r}(1-\epsilon_a)^{g-1}\epsilon_a^{r-g} \\ &\quad \cdot \left[g \cdot \tilde{e}_{g,h,j,k} - (r-g+1)\cdot \tilde{e}_{g-1,h,j,k}\right], \end{aligned} \tag{11}$$

where $m = n - r - 1$ and $\tilde{e}_{g,h,j,k}$ is the unnormalized split information function of Hadamard code and can be easily computed by its definition (8) based on the generation matrix of the Hadamard code.

Proof: Since for p_m, we have both the information y_{p_m} from the communication channel and the information v_1 from the extrinsic channel, for convenience we combine these two sources of information and denote it by v_1^*. Then the equivalent BEC for v_1^* is the one with erasure probability $\epsilon_{v_1^*} = \epsilon_y \epsilon_{v_1}$.

Let $\underline{A}_{[i]}$ be a random vector associated with $\underline{a}$, excluding a_i. Also let $\underline{Y}_{p_{[m]}}$, V_0 and V_1^* be the corresponding random variables for $\underline{y}_{p_{[m]}}$, v_0 and v_1^*, respectively. Since E_{d_i} is a function of $\underline{A}_{[i]}$, $\underline{Y}_{p_{[m]}}$, V_0 and V_1^*, the summands in (9) can be rewritten as

$$I(X_{d_i}; E_{d_i}) = I(X_{d_i}; \underline{A}_{[i]}, \underline{Y}_{p_{[m]}}, V_0, V_1^*) = \underbrace{H(X_{d_i})}_{1} - H(X_{d_i} \mid \underline{A}_{[i]}, \underline{Y}_{p_{[m]}}, V_0, V_1^*). \tag{12}$$

Let $S_{\underline{a}_{[i]}}$, $S_{\underline{y}_{[m]}}$, S_{v_0} and $S_{v_1^*}$ be the sets of positions without erasures in the respective $\underline{a}_{[i]}$, $\underline{y}_{p_{[m]}}$, v_0 and v_1^*. For *linear* codes, we have

$$H(X_{d_i} \mid \underline{A}_{[i]} = \underline{a}_{[i]}, \underline{Y}_{p_{[m]}} = \underline{y}_{p_{[m]}}, V_0 = v_0, V_1^* = v_1^*) = \kappa_{i \cup S_{\underline{a}_{[i]}} \cup S_{\underline{y}_{[m]}} \cup S_{v_0} \cup S_{v_1^*}} - \kappa_{S_{\underline{a}_{[i]}} \cup S_{\underline{y}_{p_{[m]}}} \cup S_{v_0} \cup S_{v_1^*}}.$$

For BEC, averaging over all $S_{\underline{a}_{[i]}}$, $S_{\underline{y}_{p_{[m]}}}$, S_{v_0}, $S_{v_1^*}$ and r positions, we obtain

$$\begin{aligned}
&\frac{1}{r}\sum_{i=1}^{r} H(X_{d_i} \mid \underline{A}_{[i]}, \underline{Y}_{[m]}, V_0, V_1^*) \\
&= \frac{1}{r}\sum_{i=1}^{r}\sum_{h=0}^{m-1}(1-\epsilon_y)^h \epsilon_y^{m-1-h} \sum_{|S_{\underline{y}_{p_{[m]}}}|=h} \sum_{j=0}^{1}(1-\epsilon_{v_0})^j \epsilon_{v_0}^{1-j} \sum_{|S_{v_0}|=j} \sum_{k=0}^{1}(1-\epsilon_{v_1^*})^k \epsilon_{v_1^*}^{1-k} \\
&\quad \cdot \sum_{|S_{v_1^*}|=k} \sum_{g=1}^{r}(1-\epsilon_a)^{g-1}\epsilon_a^{r-g} \left[\sum_{|S_{\underline{a}_{[i]}}|=g-1} \kappa_{i \cup S_{\underline{a}_{[i]}} \cup S_{\underline{y}_{p_{[m]}}} \cup S_{v_0} \cup S_{v_1^*}} - \kappa_{S_{\underline{a}_{[i]}} \cup S_{\underline{y}_{p_{[m]}}} \cup S_{v_0} \cup S_{v_1^*}} \right]. \quad (13)
\end{aligned}$$

We move the summations over i, $|S_{\underline{y}_{p_{[m]}}}| = h$, $|S_{v_0}| = j$ and $|S_{v_1^*}| = k$ inside the square brackets and write

$$\begin{aligned}
&\sum_{|S_{\underline{y}_{p_{[m]}}}|=h} \sum_{|S_{v_0}|=j} \sum_{|S_{v_1^*}|=k} \left(\sum_{i=1}^{r} \left\{ \sum_{|S_{\underline{a}_{[i]}}|=g-1} \kappa_{i \cup S_{\underline{a}_{[i]}} \cup S_{\underline{y}_{p_{[m]}}} \cup S_{v_0} \cup S_{v_1^*}} - \kappa_{S_{\underline{a}_{[i]}} \cup S_{\underline{y}_{p_{[m]}}} \cup S_{v_0} \cup S_{v_1^*}} \right\} \right) \\
&= \sum_{|S_{\underline{y}_{p_{[m]}}}|=h} \sum_{|S_{v_0}|=j} \sum_{|S_{v_1^*}|=k} \left(\sum_{i=1}^{r} \left\{ \sum_{|S_{\underline{a}_{[i]}}|=g-1} \kappa_{i \cup S_{\underline{a}_{[i]}} \cup S_{\underline{y}_{p_{[m]}}} \cup S_{v_0} \cup S_{v_1^*}} - \kappa_{S_{\underline{a}_{[i]}} \cup S_{\underline{y}_{p_{[m]}}} \cup S_{v_0} \cup S_{v_1^*}} \right\} \right) \\
&= \sum_{|S_{\underline{y}_{p_{[m]}}}|=h} \sum_{|S_{v_0}|=j} \sum_{|S_{v_1^*}|=k} \left(\sum_{|S_{\underline{a}}|=g} g\kappa_{S_{\underline{y}_{p_{[m]}}} \cup S_{\underline{a}} \cup S_{v_0} \cup S_{v_1^*}} - \sum_{|S_{\underline{a}}|=g-1} (r-g+1)\kappa_{S_{\underline{y}_{p_{[m]}}} \cup S_{\underline{a}} \cup S_{v_0} \cup S_{v_1^*}} \right) \\
&= g \cdot \tilde{e}_{g,h,j,k} - (r-g+1) \cdot \tilde{e}_{g-1,h,j,k}, \quad (14)
\end{aligned}$$

where $\tilde{e}_{g,h,j,k}$ is the unnormalized split information function of the Hadamard code $\mathcal{C}$ with the four parts $(\underline{d}, \underline{p}_{[m]}, q, p_m)$. Finally, substituting (14) into (12) and (9), we obtain (11). ∎

Now we consider the mutual information $I_{E,q} = I(X_q; E_q)$ for the connection bit q.

Theorem 3.2 *If the extrinsic and communication channels are BECs with respective erasure probabilities $\epsilon_a = 1 - I_{A,d}$, ϵ_y and $\epsilon_{v_1} = 1 - I_{A,p_m}$, where $I_{A,d}$ and I_{A,p_m} are respectively the a priori mutual information about the information bits $\underline{d}$ and the last parity bit p_m then we*

have

$$I_{E,q}^{BEC}(\epsilon_a, \epsilon_y, \epsilon_{v_1}) = 1 - \sum_{h=0}^{m-1}(1-\epsilon_y)^h \epsilon_y^{m-1-h} \sum_{k=0}^{1}(1-\epsilon_y\epsilon_{v_1})^k \epsilon_y^{1-k}\epsilon_{v_1}^{1-k} \sum_{g=0}^{r}(1-\epsilon_a)^g \epsilon_a^{r-g}\left[\tilde{e}_{g,h,1,k} - \tilde{e}_{g,h,0,k}\right]. \quad (15)$$

Proof: Similar to the proof of Theorem 3.1, we have

$$I(X_q; E_q) = I(X_q; \underline{A}, \underline{Y}_{[m]}, V_1^*) = \underbrace{H(X_q)}_{1} - H(X_q \mid \underline{A}, \underline{Y}_{[m]}, V_1^*),$$

where V_1^* again denotes the random variable associated with v_1^* which combines the information from the communication channel and the extrinsic channel about p_m. Following the similar steps for deriving (11), we obtain the mutual information for the connection bit q given by (15). ∎

The EXIT function of the last parity bit p_m is different from that of q, since p_m has a channel observation y_m with an erasure probability of ϵ_y. Let $\mathcal{C}^*$ be a new code formed by all pairs $(\underline{d}, \underline{p}, q, p_m)$. Since L_{E,p_m} is a function of $\underline{a}$, $\underline{y}_p$ and v_0, we have

$$I_{E,p_m} = I(X_{p_m}; E_{p_m}) = I(X_{p_m}; \underline{A}, \underline{Y}_p, V_0) = 1 - H(X_{p_m} \mid \underline{A}, \underline{Y}_p, V_0).$$

It is clear that $\mathcal{C}^*$ is also a linear code and its generation matrix is given by $\underline{G}^* = [\underline{G} \mid \underline{g}]$ where $\underline{g}$ be the last column of $\underline{G}$. Hence we have

$$H(X_{p_m} \mid \underline{A} = \underline{a}, \underline{Y}_p = \underline{y}_p, V_0 = v_0) = \kappa_{i_m \cup S_{\underline{a}} \cup S_{\underline{y}_p} \cup S_{v_0}} - \kappa_{S_{\underline{a}} \cup S_{\underline{y}_p} \cup S_{v_0}},$$

where i_m denotes the bit index for p_m. Averaging over all possible $\underline{a}$, $\underline{y}_p$ and v_0, we have

$$H(X_{p_m} \mid \underline{A}, \underline{Y}_p, V_0) = \sum_{h=0}^{m}(1-\epsilon_y)^h \epsilon_y^{m-h} \sum_{j=0}^{1}(1-\epsilon_{v_0})^j \epsilon_{v_0}^{1-j} \sum_{g=0}^{r}(1-\epsilon_a)^g \epsilon_a^{r-g}\left[\tilde{e}^*_{g,h,j,1} - \tilde{e}^*_{g,h,j,0}\right],$$

where $\tilde{e}^*_{g,h,j,k}$ denotes the unnormalized split information function of $\mathcal{C}^*$. The EXIT function for p_m is then given by the following result.

Theorem 3.3 *If the extrinsic and communication channels are BECs with respective erasure probabilities* $\epsilon_a = 1 - I_{A,d}$, ϵ_y *and* $\epsilon_{v_0} = 1 - I_{A,q}$, *where* $I_{A,d}$ *and* $I_{A,q}$ *are respectively the a priori mutual information about the information bits* $\underline{d}$ *and connection bit* q, *then we have*

$$I_{E,p_m}(\epsilon_a, \epsilon_y, \epsilon_{v_0}) = 1 - \sum_{h=0}^{m} (1-\epsilon_y)^h \epsilon_y^{m-h} \sum_{j=0}^{1} (1-\epsilon_{v_0})^j \epsilon_{v_0}^{1-j} \sum_{g=0}^{r} (1-\epsilon_a)^g \epsilon_a^{r-g} \left[\tilde{e}^*_{g,h,j,1} - \tilde{e}^*_{g,h,j,0}\right]. \quad (16)$$

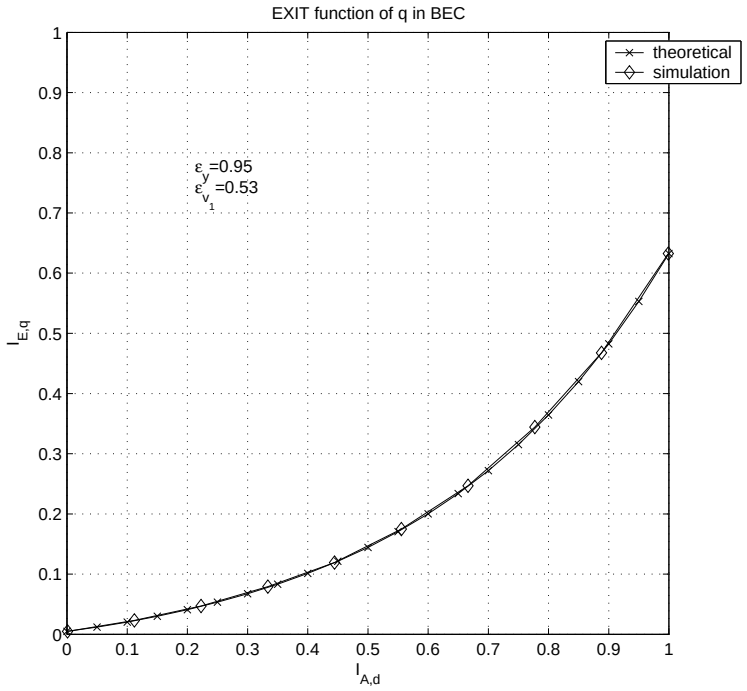


Figure 7: EXIT function for connection bit q in BEC.

Numerical Example

We show by the following example the procedure of computing the EXIT functions for the Hadamard component in a RZH code.

Example 1: (EXIT functions of Hadamard codes in BEC.) The EXIT functions are related to the unnormalized split information functions which can be computed by a code's generation

matrix. Note that for $r = 2$, the Hadamard code is essentially a $(4, 3)$ single parity-check code with the generation matrix given by

$$\underline{G}_{H,r=2} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}. \tag{17}$$

Given the generation matrix of the Hadamard codes $\underline{G}_{H,r}$, with the Hadamard matrix given in (1), the generation matrix of the Hadamard codes with order $r + 1$ is given by

$$\underline{G}_{H,r+1} = \begin{bmatrix} \underline{G}_{H,r} & \widetilde{\underline{G}}_{H,r} \\ \mathbf{0}^T_{2^r \times 1} & \mathbf{1}^T_{2^r \times 1} \end{bmatrix}, \tag{18}$$

where $\mathbf{0}_{2^r \times 1}$ denotes a length-2^r all-zero vector, $\mathbf{1}_{2^r \times 1}$ denotes a length-2^r all-one vector, and

$$\widetilde{\underline{G}}_{H,r} \triangleq (\boldsymbol{x}\mathbf{1}^T_{2^r \times 1}) \oplus \underline{G}_{H,r}, \tag{19}$$

where $\boldsymbol{x} = [1, 0, \cdots, 0]^T_{(r+1)\times 1}$ and $\oplus$ denotes modula-2 summation. It is clear that $\widetilde{\underline{G}}_{H,r}$ only differs from $\underline{G}_{H,r}$ in the first row with $0 \to 1$ and $1 \to 0$. By using (17) and (18), the generation matrix for the Hadamard code with $r = 4$ is given by

$$\underline{G}_{H,r=4} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix},$$

with which we can obtain the corresponding unnormalized split information functions $\tilde{e}_{g,h,j,k}$. For example, to compute $\tilde{e}_{1,0,1,0}$, we first obtain all the four subsets of $\mathcal{S}_{1,0,1,0}$ which are $\{1, 2\}$, $\{1, 3\}$, $\{1, 5\}$ and $\{1, 9\}$. Then we obtain the rank of the 5×2 generation matrix of each corresponding subcode, e.g., $\{1, 2\}$ leads to the subcode constructed by the 1st and 2nd columns of $\underline{G}_{H,r=4}$. It is easily seen that all four matrices have a rank of 2. Then with

(8), we have $\tilde{e}_{1,0,1,0} = 2 + 2 + 2 + 2 = 8$. Other terms can be computed in a similar way. Moreover, appending $\underline{G}_{H,r=4}$ by its last column, we obtain the generation matrix of $\mathcal{C}^*$, with which $\tilde{e}^*_{g,h,j,k}$ can be obtained similarly. Using these unnormalized split information functions, EXIT functions for different segments of the Hadamard code can then be calculated according to Theorems 3.1-3.3. For instance, to consider the EXIT function for the connection bit q, we first fix $\epsilon_y = 0.95$ and $\epsilon_{v_1} = 0.53$. By varying $\epsilon_a = 1 - I_{A,d}$, we obtain $I_{E,q}$ as a function of $I_{A,d}$ using Theorem 3.2 which is plotted in Fig. 7. Also plotted in the figure is the simulated EXIT curve of $I_{E,q}$ as a function of $I_{A,d}$ with fixed ϵ_y and ϵ_{v_1}. It is seen that the simulated and theoretical EXIT curves match very well. Fig. 8 shows the theoretical and simulated EXIT curves for the last parity bit p_m as a function of $I_{A,d}$ where $\epsilon_y = 0.95$ and $\epsilon_{v_0} = 0.3$. Again the theoretical result matches well with the the simulation result.

The application of EXIT functions to the irregular code design in BEC will be considered in Section 5.

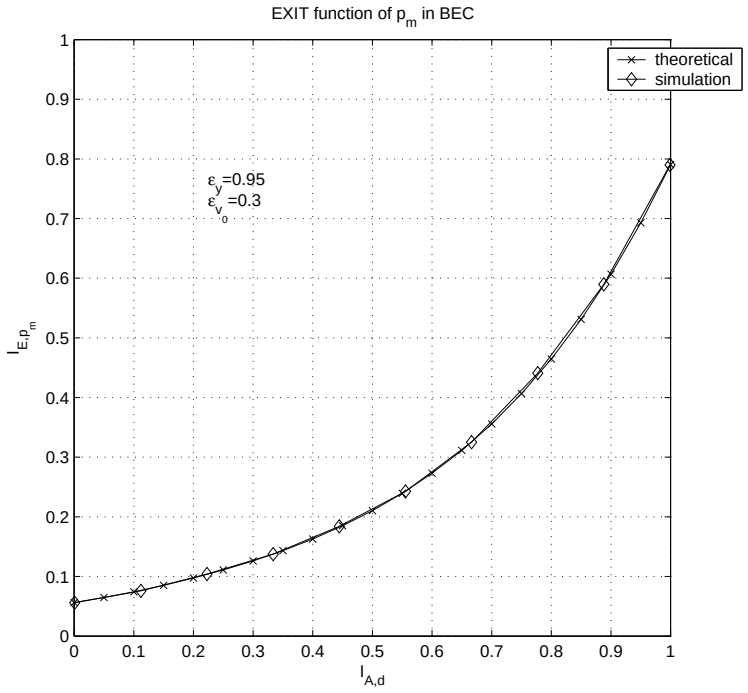


Figure 8: EXIT function for the last parity bit p_m in BEC.

4 EXIT Functions on AWGN Channels

In this section, we consider the EXIT functions for Hadamard codes with multiple inputs on BIAWGN channels. Since it is difficult to directly compute the EXIT functions of the output from the MAP decoder, we make use of the pMAP decoder defined in [17] to derive an approximation for the extrinsic mutual information over BIAWGN channels. In what follows, we first introduce some basic properties of BIAWGN channel and the pMAP decoder, with which we will derive approximate EXIT functions over BIAWGN channels for general linear block codes and Hadamard codes.

Properties of BIAWGN Channels

We first characterize the BIAWGN channel, its extrinsic LLR and "soft bit" outputs. A BIAWGN channel is characterized by its variance σ^2. Let Y denote the BIAWGN channel observation for a transmitted BPSK modulated bit X, and let $L \triangleq \log \frac{\Pr(Y|X=+1)}{\Pr(Y|X=-1)}$, $T \triangleq \Pr(Y|X=+1) - \Pr(Y|X=-1)$ denote the extrinsic LLR and "soft bit" estimation of X based on the channel observation Y respectively. For a BIAWGN channel we have $L = \frac{2Y}{\sigma^2}$ and $T = \tanh(\frac{L}{2})$. Suppose that $+1$ is transmitted, then $(L|X=+1) \sim \mathcal{N}(m, 2m)$, where $m = \frac{2}{\sigma^2}$. Hence $(T|X=+1)$ has the following pdf

$$f_{T|X=+1}(t) = \frac{2}{1-t^2} f_L\left(\ln \frac{1+t}{1-t}\right), \tag{20}$$

where f_L denotes the pdf of $(L|X=+1)$. With (20), the $(2i)$-th moment of $(T|X=+1)$ as a function of m is given by

$$\Phi_i(m) \triangleq \mathbb{E}\left\{T^{2i} \mid X=+1\right\} = \int_{-1}^{+1} \frac{2t^{2i}}{(1-t^2)\sqrt{4\pi m}} \exp\left(-\frac{\left(\ln \frac{1+t}{1-t} - m\right)^2}{4m}\right) dt. \tag{21}$$

The mutual information between X and L as a function of m is given by [19]

$$J(m) \triangleq I(X;L) = \int_{\mathcal{R}} \frac{1}{\sqrt{4\pi m}} \exp\left(-\frac{(\ell-m)^2}{4m}\right) \left(1 - \log_2(1+e^{-\ell})\right) d\ell, \tag{22}$$

which is invertible and hence $m = J^{-1}(I)$.

MAP and pMAP Decoders

Now we consider the extrinsic output of the MAP decoder and the pMAP decoder [17]. Given an (n,k) linear block code $\mathcal{C}$ over $GF(2)$, the dual code $\mathcal{C}^{\perp}$ is an $(n,n-k)$ linear block code. Let $\underline{c}_j^{\perp} = [c_{j,1}^{\perp}, c_{j,2}^{\perp}, \cdots, c_{j,n}^{\perp}]^T$ denote the j-th codeword of the $(n,n-k)$ dual code. Denote by $I_{i,1}^{\perp} = \{j | c_{j,i}^{\perp} = 1\}$ the set of indices of all codewords in the dual code with the i-th bit being 1 (or the set of indices of all parity-checks that the i-th bit participates in), and $I_{i,0}^{\perp} = \{j | c_{j,i}^{\perp} = 0\}$ the set of indices of all codewords in the dual code with the i-th bit being 0.

Suppose that the n transmitted BPSK modulated coded bits are $\underline{X} = [X_1, X_2, \cdots, X_n]$ and the prior soft estimations for these bits are $\underline{T} = [T_1, T_2, \cdots, T_n]$ which will be used as the input of the MAP decoder. Based on $\underline{T}$, the extrinsic output of a MAP decoder for the i-th bit in terms of the LLR is defined as

$$L_{EMAP,i} \triangleq \log \frac{\Pr(X_i = +1 \mid \underline{T}_{[i]})}{\Pr(X_i = -1 \mid \underline{T}_{[i]})}, \tag{23}$$

where $\underline{T}_{[i]}$ represents all the elements of $\underline{T}$ except T_i. The extrinsic output "soft bit" of the decoder is then defined as

$$T_{EMAP,i} \triangleq \Pr(X_i = +1 \mid \underline{T}_{[i]}) - \Pr(X_i = -1 \mid \underline{T}_{[i]}). \tag{24}$$

As in [5], the extrinsic MAP decoding can be implemented by using the dual code as follows

$$T_{EMAP,i} = \Pr(X_i = +1 \mid \underline{T}_{[i]}) - \Pr(X_i = -1 \mid \underline{T}_{[i]}) = \frac{\sum_{j \in I_{i,1}^{\perp}} \prod_{\ell=1, \ell \neq i}^{n} T_{\ell}^{c_{j,\ell}^{\perp}}}{\sum_{j \in I_{j,0}^{\perp}} \prod_{\ell=1, \ell \neq j}^{n} T_{\ell}^{c_{j,\ell}^{\perp}}}. \tag{25}$$

Let $D_{EMAP,i}$ be the extrinsic output of MAP decoder (25) after the following modification

$$D_{EMAP,i} \triangleq 1 + T_{EMAP,i} = 2\Pr(X_i = +1 \mid \underline{T}_{[i]}) = 1 + \frac{\sum_{j \in I_{i,1}^{\perp}} \prod_{\ell=1, \ell \neq j}^{n} T_{\ell}^{c_{j,\ell}^{\perp}}}{\sum_{j \in I_{j,0}^{\perp}} \prod_{\ell=1, \ell \neq i}^{n} T_{\ell}^{c_{j,\ell}^{\perp}}}. \tag{26}$$

The extrinsic output of a suboptimal decoder which we call pMAP decoder [17] is defined as

$$D_{EpMAP,i} \triangleq \prod_{S \in P(I_{i,1}^{\perp}) \setminus \phi} \left(1 + \prod_{\ell=1, \ell \neq i}^{n} T_{\ell}^{\vee_{j \in S}(c_{j,l}^{\perp})} \right)^{(-1)^{|S|+1}}, \tag{27}$$

where $P(I_{i,1}^{\perp})$ denotes the set of all subsets of $I_{i,1}^{\perp}$ and "$\vee$" denotes the logical "or" operation.

In what follows, we will employ the pMAP decoder to obtain an approximate expression for the EXIT functions of linear block codes over a BIAWGN channel. Since the basic idea is to decompose the EXIT function in the form of a series EXIT functions of multiple single parity-check codes, we will start from the EXIT function of a single parity-check code.

EXIT Functions for General Linear Block Codes

Note that our goal is to find the extrinsic functions for different bits given that the *a priori* mutual information from the extrinsic and the communication channels for different bits is different. Keeping this in mind, we first compute the extrinsic information at the output of a single parity-check code.

Theorem 4.4 *The extrinsic information of the i-th bit at the output of MAP decoder of an* $(n, n-1)$ *single parity-check code is given by:*

$$I_{E,i} = I(X_i; T_{EMAP,i}) = \frac{1}{\ln 2} \sum_{j=1}^{\infty} \frac{1}{(2j)(2j-1)} \left(\prod_{\ell=1, \ell \neq i}^{n} \mathbb{E}\{T_{\ell}^{2j}\} \right). \tag{28}$$

Proof: The dual code of the $(n, n-1)$ single parity-check code is a repetition code of length n. Hence from (25) we have $T_{EMAP,i} = \prod_{\ell=1, \ell \neq i}^{n} T_{\ell}$, then using Proposition 2.6 in [17], it is easy to prove (28). ∎

As shown in [17], for the single parity-check codes, pMAP and MAP decoders are identical in any memoryless channel. Moreover, Conjecture 4.1 in [17] states that for other *high rate* codes, $D_{EMAP} \cong D_{EpMAP}$ when the all-zero codeword is transmitted over the BIAWGN channel. We have the following result.

Theorem 4.5 *Let* $\mathcal{C}$ *be an (n,k) linear block code such that in BIAWGN channel* $D_{EMAP} = D_{EpMAP}$. *Then the mutual information between* X_i *and* $T_{EMAP,i}$ *is given by*

$$I_{E,i}^{BIAWGN} = \frac{1}{\ln 2} \sum_{S \in P(I_{i,1}^{\perp}) \setminus \phi} (-1)^{|S|+1} \sum_{j=1}^{\infty} \frac{1}{(2j)(2j-1)} \left(\prod_{\ell \in V(S), \ell \neq i} \Phi_j(m_{\ell}) \right), \tag{29}$$

where $V(S) \triangleq \left\{\ell \in \{1,\ldots,n\} | \vee_{j\in S} c_{j,\ell}^{\perp} = 1\right\}$, *and* $m_\ell = J^{-1}(I_{A,\ell})$ *with* $I_{A,\ell}$ *being the a priori mutual information about* X_ℓ *from the channel given* $X_\ell = +1$ *is transmitted.*

Proof: According to Theorem 2.10 in [17] and (26), we have

$$\begin{aligned} I_{E_i}^{BIAWGN} &= I(X_i; T_{EMAP,i}) = \mathbb{E}_{T_{EMAP,i}|\underline{c}_0}\left\{\log_2(1+T_{EMAP,i})\right\} \\ &= \mathbb{E}_{T_{EMAP,i}|\underline{c}_0}\left\{\log_2(D_{EMAP,i})\right\}. \end{aligned}$$

Since $D_{EMAP} = D_{EpMAP}$, we further have

$$\begin{aligned} I_{E_i}^{BIAWGN} &= \mathbb{E}_{T_{EMAP,i}|\underline{c}_0}\left\{\log_2(D_{EpMAP,i})\right\} \\ &= \mathbb{E}_{\underline{T}|\underline{c}_0}\left\{\log_2\left(\prod_{S\in P(I_{i,1}^{\perp})\setminus\phi}\left(1+\prod_{\ell=1,\ell\neq i}^{n} T_\ell^{\vee_{j\in S}(c_{j,\ell}^{\perp})}\right)^{(-1)^{|S|+1}}\right)\right\} \\ &= \mathbb{E}_{\underline{T}|\underline{c}_0}\left\{\sum_{S\in P(I_{i,1}^{\perp})\setminus\phi}(-1)^{|S|+1}\log_2\left(1+\prod_{\ell=1,\ell\neq i}^{n} T_\ell^{\vee_{j\in S}(c_{j,\ell}^{\perp})}\right)\right\} \\ &= \sum_{S\in P(I_{i,1}^{\perp})\setminus\phi}(-1)^{|S|+1}\mathbb{E}_{\underline{T}|\underline{c}_0}\left\{\log_2\left(1+\prod_{\ell=1,\ell\neq i}^{n} T_\ell^{\vee_{j\in S}(c_{j,\ell}^{\perp})}\right)\right\}. \end{aligned} \tag{30}$$

Define $V(S) \triangleq \left\{\ell \in \{1,\ldots,n\} | \vee_{j\in S} c_{j,\ell}^{\perp} = 1\right\}$, then the $\mathbb{E}\{\cdot\}$ term in (30) is the extrinsic information of a $(|V(S)|, |V(S)|-1)$ single parity-check code. With Theorem 4.4, it is easy to prove (29). Note that here we cannot expand the EXIT functions in the form of multiple EXIT functions of the same code in BEC channel, which is different from [17]. ∎

For high rate codes, we have $D_{EMAP} \cong D_{EpMAP}$ when the all-zero codeword is transmitted over the BIAWGN channel, and Theorem 4.5 can be used to approximately compute the extrinsic information at the output of the MAP decoder for a linear block code. Note that Hadamard codes are low rate codes. As in [17], for a low rate code, the extrinsic information can be approximated by

$$I_{E,i}^{BIAWGN}(I_{A,1},\ldots,I_{A,n}) \cong 1 - I_{E,i}^{\perp,BIAWGN}(1-I_{A,1},\ldots,1-I_{A,n}), \tag{31}$$

where $I_{E,i}^{\perp,BIAWGN}(1-I_{A,1},\ldots,1-I_{A,n})$ is the extrinsic information of its high-rate dual code and can be computed by Theorem 4.5 with $m_\ell = J^{-1}(1-I_{A,\ell})$.

EXIT Functions for Hadamard Codes over BIAWGN Channel

Next we consider the EXIT functions of Hadamard codes in the context of RZH codes with parallel decoding. We first consider the average mutual information about the information bits $\underline{d}$ which is given by $I_{E,d}^{BIAWGN} = \frac{1}{r}\sum_{i=1}^{r} I_{E,d_i}^{BIAWGN}$, where I_{E,d_i}^{BIAWGN} can be computed by (31).

Let $\underline{c}_j = [c_{j,1}, c_{j,2}, \ldots, c_{j,n}]^T$ denote the j-th codeword of the Hadamard code over $GF(2)$. For $\underline{c}_j$, we have $c_{j,2^{i-1}+1} = d_{j,i}$ for $i = 1, \ldots, r$. Denote by $I_{i,1}^{H,d} = \{j | c_{j,2^{i-1}+1} = 1\}$, for $i = 1, \ldots, r$, the set of indices of all Hadamard codewords with the i-th systematic bit equal 1, and $P(I_{i,1}^{H,d})$ the set of all subsets of $I_{i,1}^{H,d}$. Since a Hadamard codeword can be divided into systematic bits, parity bits and connection bits, we have $\underline{c}_j = \{\underline{d}_j, \underline{p}_{j,[m]}, q_j, p_{j,m}\}$ where $\underline{p}_{j,[m]}$ denotes the parity bits exclusive of $p_{j,m}$. For $S \in P(I_{i,1}^{H,d})$, the following definitions will be used:

$$V(S, \underline{d}) \triangleq \{\ell \in \{1, \ldots, r\} \mid \vee_{j\in S} d_{j,\ell} = 1\}, \tag{32}$$

$$V(S, \underline{p}_{[m]}) \triangleq \{\ell \in \{1, \ldots, m-1\} \mid \vee_{j\in S} p_{j,\ell} = 1\}, \tag{33}$$

$$V(S, q) \triangleq \{\ell \in \{1\} \mid \vee_{j\in S} q_{j,\ell} = 1\}, \tag{34}$$

$$V(S, p_m) \triangleq \{\ell \in \{m\} \mid \vee_{j\in S} p_{j,\ell} = 1\}, \tag{35}$$

where "$\vee$" is the logical "or" operation. It is clear that $V(S, q)$ is either $\{1\}$ or ϕ; similar for $V(S, p_m)$. Further define

$$A_{g,h,j,k}^{d_i,odd} \triangleq \left|\left\{S \in P(I_{i,1}^{H,d}) \;\middle|\; |V(S,q)| = g, |V(S,\underline{d})| = h, |V(S,\underline{p}_{[m]})| = j, |V(S,p_m)| = k, |S| \text{ is odd}\right\}\right|$$

and

$$A_{g,h,j,k}^{d_i,even} \triangleq \left|\left\{S \in P(I_{i,1}^{H,d}) \;\middle|\; |V(S,q)| = g, |V(S,\underline{d})| = h, |V(S,\underline{p}_{[m]})| = j, |V(S,p_m)| = k, |S| \text{ is even}\right\}\right|.$$

Then we have the following results for Hadamard codes that will be subsequently used.

Lemma 4.1 *For Hadamard codes $\mathcal{C}$ with order r, we have* $A_{g,h,j,k}^{d_1,odd} = A_{g,h,j,k}^{d_2,odd} = \cdots = A_{g,h,j,k}^{d_r,odd} \triangleq A_{g,h,j,k}^{d,odd}$, *and* $A_{g,h,j,k}^{d_1,even} = A_{g,h,j,k}^{d_2,even} = \cdots = A_{g,h,j,k}^{d_r,even} \triangleq A_{g,h,j,k}^{d,even}$.

Proof: Since $A_{g,h,j,k}^{d_\ell,odd}$ and $A_{g,h,j,k}^{d_\ell,even}$ are based on the subcode $\mathcal{C}_i = \{\underline{c}_j | c_{j,2^{i-1}+1} = 1, \underline{c}_j \in \mathcal{C}\}$, $i = 1, \ldots, r$, we only need to show that for $\ell \neq i$, there is a $\mathcal{C}_\ell^*$ which is constructed from $\mathcal{C}_\ell$ by only switching coded bits within the systematic bit positions and/or the parity bit positions, such that $\mathcal{C}_i = \mathcal{C}_l^*$. Let the generation matrix of $\mathcal{C}$ in systematic form be $\underline{G}_{H,r}$ with $\underline{g}_0$ and $\underline{g}_i$ $(i = 1, \ldots, r)$ being the rows of $\underline{G}_{H,r}$ corresponding to the connection bit and the i-th information bit, respectively. Suppose that the input of the Hadamard encoder is $\underline{d} = [d_0 = q, d_1, \ldots, d_r]^T$, then the codewords belong to $\mathcal{C}_i$ are given by $\{\underline{g}_i^T + \sum_{\ell \neq i} d_\ell \underline{g}_\ell^T\}$ for all possible $\{d_\ell\}_{\ell \neq i}$. Since $\underline{G}_{H,r}$ is in systematic form, $\underline{g}_i$ can be written as $[\underline{u}_i, \underline{p}_i]$ where $\underline{u}_i = [u_0, u_1, \ldots, u_r]$ with $u_\ell = 0$ for $\ell \neq i$ and $u_i = 1$. By switching the rows $\underline{g}_i$ and $\underline{g}_\ell$, we obtain $\underline{G}_{H,r}^{(1)}$ with $\underline{g}_i^{(1)} = \underline{g}_\ell$ and $\underline{g}_\ell^{(1)} = \underline{g}_i$. It is clear that the subcode $\mathcal{C}_i^{(1)}$ generated by $\underline{G}_{H,r}^{(1)}$ is identical to $\mathcal{C}_i$. We further construct $\underline{G}_{H,r}^{(2)}$ from $\underline{G}_{H,r}^{(1)}$ by letting $\underline{g}_i^{(2)} = \underline{g}_i^{(1)} + (\underline{g}_\ell^{(1)} + \underline{g}_i^{(1)}) = \underline{g}_\ell^{(1)} = \underline{g}_i$ and $\underline{g}_\ell^{(2)} = \underline{g}_\ell^{(1)} + (\underline{g}_\ell^{(1)} + \underline{g}_i^{(1)}) = \underline{g}_i^{(1)} = \underline{g}_\ell$, by which we have $\underline{G}_{H,r}^{(2)} = \underline{G}_{H,r}$ and hence $\mathcal{C}_i^{(2)} = \mathcal{C}_i$. The effect of transforming $\underline{G}_{H,r}^{(1)}$ to $\underline{G}_{H,r}^{(2)}$ is only to change the order of the coded bits. Since both codes are systematic codes, there is one order change between systematic bit position i and ℓ (note that there are only two 1's in $\underline{u}_i + \underline{u}_\ell$), and all the other changes are between parity bit positions. It is clear that $\mathcal{C}_i^{(2)} = \mathcal{C}_i$ is the $\mathcal{C}_\ell^*$ we are looking for and this concludes the proof. ∎

In case that the size of the code book is not huge, it is easy to compute $A_{g,h,j,k}^{d,even}$ and $A_{g,h,j,k}^{d,odd}$ directly from their definitions given the codewords are known. To illustrate the basic procedure, let us consider a simple example with $r = 2$, for which the Hadamard codewords are given by [0 0 0 0; 0 1 0 1; 0 0 1 1; 1 1 1 1; 1 0 1 0; 1 1 0 0; 1 0 0 1], where the 1st column corresponds to the connection bit position, the 2nd and the 3rd columns correspond to the information bit positions, and the last column corresponds to the last parity bit position. Consider the first information bit position for which we have $I_{1,1}^{H,d} = \{2, 4, 5, 7\}$.

To compute $A_{g,h,j,k}^{d,odd}$ and $A_{g,h,j,k}^{d,even}$ from their definitions we only need to know the values of $|V(S,\underline{d})|$, $|V(S,\underline{p}_{[m]})|$, $|V(S,q)|$ and $|V(S,p_m)|$ for all possible S. For instance, in case that $|S|=2$, we have four possible S, namely, $\{2,4\}$, $\{2,5\}$, $\{2,7\}$, $\{4,5\}$, $\{4,7\}$ and $\{5,7\}$. For $S=\{2,4\}$, since the logical "or" result for the 2nd codeword [0 1 0 1] and the 4-th codeword [0 1 1 0] is [0 1 1 1], we have $V(S,\underline{d})=\{1,2\}$, $V(S,\underline{p}_{[m]})=\phi$, $V(S,q)=\phi$ and $V(S,p_m)=\{m\}$; hence we have $|V(S,\underline{d})|=2$, $|V(S,\underline{p}_{[m]})|=0$, $|V(S,q)|=0$ and $|V(S,p_m)|=1$. We can obtain similar values for other possible S with $|S|=odd$ or $|S|=even$, from which we are able to get $A_{g,h,j,k}^{d,odd}$ and $A_{g,h,j,k}^{d,even}$ from their definitions.

With the above results, the average mutual information about the information bits $\underline{d}$ can be computed by the following result.

Theorem 4.6 *If the extrinsic and communication channels are AWGN channels with respective mutual information $I_{A,d}$, $I_{A,p}$, $I_{A,q}$ and I_{A,p_m}, we have*

$$I_{E,d}^{BIAWGN} \cong 1-\frac{1}{\ln 2}\sum_{g=0}^{1}\sum_{h=1}^{r}\sum_{j=0}^{m-1}\sum_{k=0}^{1}\left(A_{g,h,j,k}^{d,odd}-A_{g,h,j,k}^{d,even}\right)\sum_{i=1}^{\infty}\frac{1}{(2i)(2i-1)}\cdot \Phi_i^g(m_q)\Phi_i^{h-1}(m_d)\Phi_i^j(m_p)\Phi_i^k(m_{p_m}), \tag{36}$$

where $m_q=J^{-1}(1-I_{A,q})$, $m_d=J^{-1}(1-I_{A,d})$, $m_p=J^{-1}(1-I_{A,p})$ and $m_{p_m}=J^{-1}(1-I_{A,p_m})$. Note that here for simplicity, I_{A,p_m} denotes the overall mutual information about p_m from both the communication channel and the extrinsic channel.

Proof: Using Theorem 4.5, (31) and Theorem 4.1, we obtain the above result. ∎

Next we consider the connection bit q. Similarly, we denote by $I_1^{H,q}=\{j|c_{j,1}=1\}$ the set of indices of all Hadamard codewords with the connection bit equal 1, and $P(I_1^{H,q})$ the set of all subsets of $I_1^{H,q}$. Let $S\in P(I_1^{H,q})$, then we can obtain $V(S,\underline{d})$, $V(S,\underline{p}_{[m]})$, and $V(S,p_m)$ as in (32), (33) and (35). Define

$$A_{h,j,k}^{q,odd}\triangleq\left|\left\{S\in P(I_1^{H,q})\;\middle|\;|V(S,\underline{d})|=h,|V(S,\underline{p}_{[m]})|=j,|V(S,p_m)|=k,|S|\text{ is odd}\right\}\right|$$

and

$$A_{h,j,k}^{q,even} \triangleq \left| \left\{ S \in P(I_{i,1}^{H,q}) \;\middle|\; |V(S,\underline{d})| = h, |V(S,\underline{p}_{[m]})| = j, |V(S,p_m)| = k, |S| \text{ is even} \right\} \right|.$$

Then similar to Theorem 4.6, the mutual information of q can be computed by the following result.

Theorem 4.7 *If the extrinsic and communication channels are AWGN channels with respective mutual information $I_{A,d}$, $I_{A,p}$ and I_{A,p_m}, then we have*

$$\begin{aligned} I_{E,q}^{BIAWGN} \;\cong\; 1 - \frac{1}{\ln 2} \sum_{h=0}^{r} \sum_{j=0}^{m-1} \sum_{k=0}^{1} \left(A_{h,j,k}^{d,odd} - A_{h,j,k}^{d,even} \right) \sum_{i=1}^{\infty} \frac{1}{(2i)(2i-1)} \cdot \\ \Phi_i^h(m_d) \Phi_i^j(m_p) \Phi_i^k(m_{p_m}), \end{aligned} \tag{37}$$

where $m_d = J^{-1}(1 - I_{A,d})$, $m_p = J^{-1}(1 - I_{A,p})$ and $m_{p_m} = J^{-1}(1 - I_{A,p_m})$. Again, here I_{A,p_m} represents the overall mutual information about p_m from both the communication channel and the extrinsic channel.

Finally we consider the last parity bit p_m. Since p_m has a channel observation y_m, the EXIT function of the last parity bit p_m is different from that of q. Again, let $\mathcal{C}^*$ be a new code formed by all pairs $(\underline{d}, \underline{p}, q, p_m)$, then we can define $I_1^{H,p_m} = \{j | p_{j,m} = 1\}$ the set of indices of all codewords in $\mathcal{C}^*$ with $p_{j,m} = 1$, and $P(I_1^{H,p_m})$ the set of all subsets of I_1^{H,p_m}. Then for $S \in P(I_1^{H,p_m})$, similar to the case for q, we can obtain $V(S,q)$, $V(S,\underline{d})$, and $V(S,\underline{p})$, with which we can define

$$A_{g,h,j}^{p_m,odd} \triangleq \left| \left\{ S \in P(I_1^{H,p_m}) \;\middle|\; |V(S,q)| = g, |V(S,\underline{d})| = h, |V(S,\underline{p})| = j, |S| \text{ is odd} \right\} \right|$$

and

$$A_{g,h,j}^{p_m,odd} \triangleq \left| \left\{ S \in P(I_1^{H,p_m}) \;\middle|\; |V(S,q)| = g, |V(S,\underline{d})| = h, |V(S,\underline{p})| = j, |S| \text{ is even} \right\} \right|,$$

by which we can compute the EXIT function for p_m as follows.

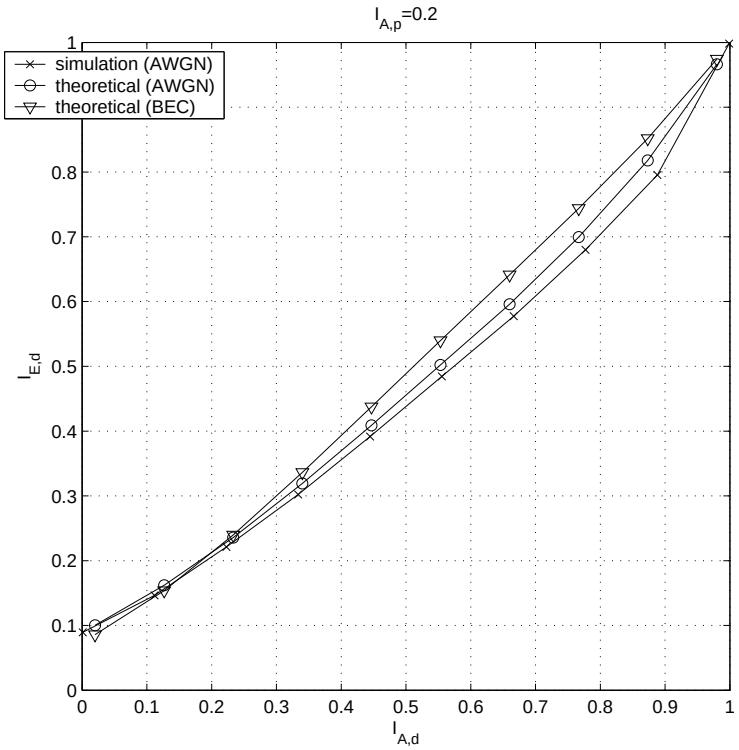


Figure 9: EXIT function of information bits in BIAWGN channel.

Theorem 4.8 *If the extrinsic and communication channels are AWGN channels with respective mutual information $I_{A,q}$, $I_{A,d}$ and $I_{A,p}$, then we have*

$$I_{E,p_m}^{BIAWGN} \cong 1 - \frac{1}{\ln 2} \sum_{g=0}^{1} \sum_{h=0}^{r} \sum_{j=0}^{m} \left(A_{g,h,j}^{p_m,odd} - A_{g,h,j}^{p_m,even} \right) \sum_{i=1}^{\infty} \frac{1}{(2i)(2i-1)} \cdot \Phi_i^g(m_q)\Phi_i^h(m_d)\Phi_i^j(m_p), \quad (38)$$

where $m_q = J^{-1}(1 - I_{A,q})$, $m_d = J^{-1}(1 - I_{A,d})$ and $m_p = J^{-1}(1 - I_{A,p})$.

Numerical Example

We use the following example to demonstrate the procedure for computing of EXIT functions for Hadamard codes in a BIAWGN channel.

Example 2: (EXIT functions of Hadamard codes in BIAWGN channel.) Consider a

Hadamard code with $r = 4$. The code book of $\mathcal{C}$ can be obtained by (1), from which we are able to obtain $A_{g,h,j,k}^{d,even}$ and $A_{g,h,j,k}^{d,odd}$. Let $I_{A,p} = 0.2$, and $I_{A,q} = I_{A,p_m} = I_{A,d}$, then the EXIT function of the information bits has only one variable $I_{A,d}$, and we are able to obtain $I_{E,d}$ as a function of $I_{A,d}$ by using Theorem 4.6 which is plotted in Fig. 9. Also depicted in the figure are the simulated EXIT curve under AWGN channel and the EXIT curve under BEC channel with the same setting. It is seen that the EXIT function over BIAWGN channel obtained in this section provides a much better approximation of the actual one than that of BEC.

5 Application to IRZH Code Design

In this section we consider the application of Hadamard code EXIT functions in the design of IRZH codes where the outer code is a mixture of repetition codes with different rates. The design approach closely follows that for irregular LDPC codes using the deferential evolution (DE) technique [15]. During the optimization, we need to estimate the BER for a given code profile, and this can be done through mutual information evolution which will be introduced next.

BER Estimation of RZH Codes

For an (n, k) IRZH codes, the outer code consists of k repetition codes with variable rates. As shown in Fig. 4, suppose that the rate of the i-th repetition code is $1/d_v^{(i)}$, then the degree of the i-th variable node of information bit is $d_v^{(i)}$. For the outer repetition codes, the EXIT function is obtained in a similar manner as that for RA codes and LDPC codes. For systematic RZH codes, an information bit variable node of degree d_v has $d_v + 1$ incoming messages: d_v from the edge interleaver and one from the channel. The decoder outputs are

given by (6). For BEC, the EXIT function of a degree-d_v variable node is given by

$$I_{E,\mathrm{VND}}(d_v) = 1 - (1 - I_{\mathrm{ch}})(1 - I_{A,\mathrm{VND}})^{d_v - 1}, \tag{39}$$

where $I_{\mathrm{ch}} = 1 - \epsilon_{\mathrm{ch}}$ is the mutual information from the communication channel with erasure probability ϵ_{ch}. For BIAWGN channel, the EXIT function of a degree-d_v variable node is given by

$$I_{E,\mathrm{VND}}(d_v) = J\left(\frac{(d_v - 1)[2J^{-1}(I_{A,\mathrm{VND}})] + \sigma_{\mathrm{ch}}^2}{2}\right), \tag{40}$$

where $J(\cdot)$ is defined in (22), and $\sigma_{\mathrm{ch}}^2 = \frac{4}{\sigma^2} = \frac{8E_s}{N_0}$ with $\frac{E_s}{N_0} = R_c \cdot \frac{E_b}{N_0}$ being the symbol SNR.

Given the Hadamard code order r, the IRZH design involves specifying the variable node degrees $d_v^{(i)}$, $i = 1, 2, \cdots, k$. Let D be the number of different variable node degrees, and denote these degrees by $d_{v,i}$, $i = 1, 2, \cdots, D$. Then the degree profile of a code can be specified by a polynomial $f(x) = \sum_{i=1}^{D} a_i x^{d_{v,i}}$ where $a_i \geq 0$ is the fraction of *nodes* having degree $d_{v,i}$. Note that $\{a_i\}$ must satisfy $\sum_i a_i = 1$. The average variable node degree is then given by $\overline{d}_v = \sum_{i=1}^{D} a_i \cdot d_{v,i}$. Since for $k \to \infty$, $R_c = \frac{r}{\overline{d}_v 2^r}$, we have $\overline{d}_v = \frac{r}{R_c 2^r}$. Let b_i be the fraction of *edges* incident to variable nodes of degree $d_{v,i}$. There are $(ka_i)d_{v,i}$ edges involved with such nodes, so we have

$$b_i = \frac{ka_i d_{v,i}}{k\overline{d}_v} = \frac{d_{v,i}}{\overline{d}_v} \cdot a_i. \tag{41}$$

Note that $\{b_i\}$ must satisfy $\sum_i b_i = 1$. It is shown in [1, 22] that the EXIT function of a mixture of codes is an average of the component EXIT functions. For systematic RZH codes, we have $R_c = \frac{r}{\overline{d}_v(2^r - r - 1) + r}$, hence

$$\overline{d}_v = \frac{r}{2^r - r - 1}\left(\frac{1}{R_c} - 1\right). \tag{42}$$

Then with (40), (41) and (42), we have the effective EXIT function of the outer mixture codes as follows

$$I_{E,\mathrm{VND}} = \sum_{i=1}^{D} b_i \cdot I_{E,\mathrm{VND}}(d_{v,i}) = \frac{2^r - r - 1}{r} \cdot \frac{R_c}{1 - R_c} \sum_{i=1}^{D} d_{v,i} \cdot a_i \cdot I_{E,\mathrm{VND}}(d_{v,i}), \tag{43}$$

where $I_{E,\mathrm{VND}}(d_{v,i})$ is given by (39) for BEC and (40) for BIAWGN.

Since the connection bit nodes are viewed as variable nodes with degree 2, the corresponding EXIT functions for q and p_m are simply the EXIT functions for rate 1/2 repetition code without message from the channel. Now we have both the EXIT functions of the outer repetition decoder and the inner Hadamard decoder, we can then trace the mutual information exchanged between the two constituent components for a given degree profile and calculate the corresponding BER after a given number of iterations as in [10], with which we are able to perform degree profile optimization by using differential evolution (DE) [15].

Specifically, let I_{ch} be the mutual information about the LLRs of the coded bits transmitted from the communication channel. For a BEC with an erasure probability of ϵ_{ch}, we have $I_{\mathrm{ch}} = 1 - \epsilon_{\mathrm{ch}}$. For a BIAWGN channel with $\frac{E_s}{N_0} = R_c \cdot \frac{E_b}{N_0}$, we have $I_{\mathrm{ch}} = J(2\sigma_{\mathrm{ch}}^2) = J(\frac{16E_s}{N_0})$. Also let $I_{A,a}^{\mathrm{H}}$, $I_{A,q}^{\mathrm{H}}$, I_{A,p_m}^{H} be the mutual information about the *a priori* LLRs passing from the repetition decoder to the Hadamard decoder (the extrinsic channels) for the information bits, connection bit and the last parity bit, respectively, which are initialized to zero at the first iteration. With I_{ch}, $I_{A,a}^{\mathrm{H}}$, $I_{A,q}^{\mathrm{H}}$ and I_{A,p_m}^{H} as the inputs, we are able to compute the output mutual information $I_{E,a}^{\mathrm{H}}$, $I_{E,q}^{\mathrm{H}}$ and I_{E,p_m}^{H} of the Hadamard decoder with the previously derived EXIT functions in either BEC or BIAWGN channel. Note that for the case of BEC, given the mutual information I_A, the erasure probability ϵ of the corresponding extrinsic channel is given by $\epsilon = 1 - I_A$. For BIAWGN channels, the mean of the "soft output" of the extrinsic channel is computed by $m = J^{-1}(I_A)$. The output mutual information of the Hadamard decoder will then serve as the input $I_{A,VND}$ for the EXIT functions of the corresponding repetition decoder, which will produce the corresponding $I_{E,VND}$ and feed back to the EXIT functions of the Hadamard decoder as the new inputs for the next iteration. After a given number of iterations, the mutual information passed to the outer repetition decoder is $I_{A,VND}$, then for BEC the average BER can be computed by

$$P_b = \sum_{i=1}^{D} a_i (1 - I_{\mathrm{ch}})(1 - I_{A,VND})^{d_{v,i}}. \tag{44}$$

For BIAWGN, the BER is given by

$$P_b = \sum_{i=1}^{D} a_i Q\left(\sqrt{\frac{1}{2} J\left(\frac{1}{2}\left(2d_v J^{-1}(I_{A,\mathrm{VND}}) + \sigma_{\mathrm{ch}}^2\right)\right)}\right), \tag{45}$$

where $Q(x) \triangleq \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} \exp(-t^2/2)\, dt$.

Code Ensemble Optimization

We first consider BIAWGN channel. For given r and R_c, the optimal IRZH ensemble parameters $\{a_i\}$ are the one that minimize E_b/N_0, subject to a vanishing BER P_b. In practice, the vanishing BER condition is often replaced by a BER threshold ε for a given iteration number N_{it}, i.e., $P_b(N_{it}) < \varepsilon$, which leads to the following optimization problem

$$\min\ E_b/N_0 \quad \text{s.t.} \quad \sum_i a_i = 1, a_i \geq 0, \quad \forall i \quad \text{and} \quad P_b(N_{it}) < \varepsilon. \tag{46}$$

To solve (46) for irregular RZH codes with parallel decoding, we resort to the differential evolution (DE) algorithm [15], which is a powerful population-based genetic algorithm and was introduced in [16] for finding the optimal degree profile of the irregular LDPC codes. The essential idea behind DE is a self-organizing scheme to generate a trial parameter vector (which is $\{a_i\}$ in our case) by adding the weighted difference between two population vectors of the current generation to a third vector which is the target vector. If the trial vector has a smaller cost value (which is P_b in our case and can be estimated by using the EXIT functions of the component codes as discussed in Section 5) than the target vector, it survives into the next generation of the evolution. After a given number of evolutions, the vector with the smallest cost value among the population of current generation is the optimized parameter vector. Note that although there is no mathematical convergence proof for DE, it is believed to be a simple and reliable global optimization method. To find the solution to (46) for given R_c and r, we first choose an E_b/N_0 and find a degree profile $\{a_i\}$ using DE that satisfies the BER constraint ϵ. If at least one $\{a_i\}$ exists, the E_b/N_0 is reduced and the procedure is

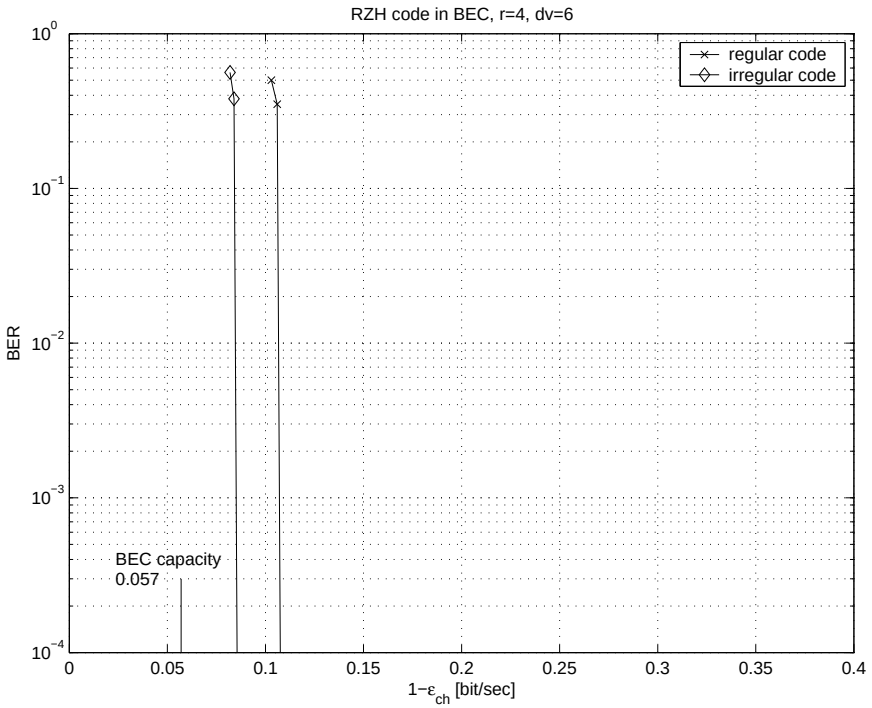


Figure 10: BER performance of RZH codes over BEC.

repeated until no such a vector $\{a_i\}$ exists. The smallest value of E_b/N_0 for which a power profile $\{a_i\}$ satisfies the BER constraint is the desired minimum SNR and the corresponding $\{a_i\}$ is the optimized degree profile.

For BEC, we simply replace E_b/N_0 by the BEC channel capacity $(1 - \epsilon_{\text{ch}})$ and similar procedure can be employed to obtain the optimal degree profile.

Design Examples

In what follows, we provide several examples for RZH code analysis and degree profile optimization by using the EXIT function results obtained in this paper.

Example 3: (IRZH code ensemble optimization with $r = 4$ and $\bar{d}_v = 6$ over BEC.) We first consider the performance of a regular systematic RZH code in BEC where $r = 4$ and $\bar{d}_v = 6$ corresponding to a rate of $R_c = 0.057$. Fig. 10 depicts the simulated BER performance as

a function of the channel capacity (which is given by $1 - \epsilon_{\text{ch}}$) of this code. The simulated capacity threshold (measured at $P_b = 10^{-4}$) is around 0.108 bit/sec which is 0.051 bit/sec away from the capacity. Now we consider the irregular code. With the BEC EXIT functions, we are able to obtain an optimized irregular code of the same rate at $1 - \epsilon_{\text{ch}} = 0.062$ bit/sec with profile $f(x) = 0.4272\,x^3 + 0.1857\,x^4 + 0.0426\,x^5 + 0.0530\,x^7 + 0.0245\,x^{10} + 0.0572\,x^{11} + 0.2098\,x^{12}$, which is only 0.005 bit/sec away from the capacity. The simulated capacity threshold for this code is 0.086 bit/sec, which is only 0.029 dB away from the capacity and has a gain of 0.022 bit/sec over the regular code.

Example 4: (IRZH code ensemble optimization with $r = 4$ and $\bar{d}_v = 7$ over BIAWGN channel.) Note that for the analysis and design of LDPC codes, it is a common practice to use the EXIT function of the inner check node over BEC to approximate that over BIAWGN channel. And results show that such an approximation works very well for irregular LDPC code design. Here we will show that for RZH codes, however, such a BEC approximation provides less satisfactory results. On the other hand, the approximate EXIT functions for BIAWGN channel derived in Section 4, when employed to design IRZH codes, lead to codes with performance much closer to the capacity.

We first consider the performance of the regular code. Consider a regular systematic RZH code with $r = 4$ and $\bar{d}_v = 7$ corresponding to a rate of $R_c = 0.0494$. Fig. 11 depicts the simulated and the estimated BER performance of this code. The simulated SNR threshold (measured at $P_b = 10^{-4}$) is around -0.66 dB which is 0.78 dB away from the Shannon capacity -1.44 dB. The estimated SNR threshold using EXIT functions over BEC is around -0.05 dB, which is 0.61 dB away from the simulated one, whereas by using EXIT functions derived for the BIAWGN channel, the estimated SNR threshold is around -0.75 dB which is only 0.09 dB away from the simulated result and far more better than that use BEC approximation. This is consistent with the observation in Fig. 9.

Next we consider irregular code design. It is seen from Fig. 12 that the performance of the

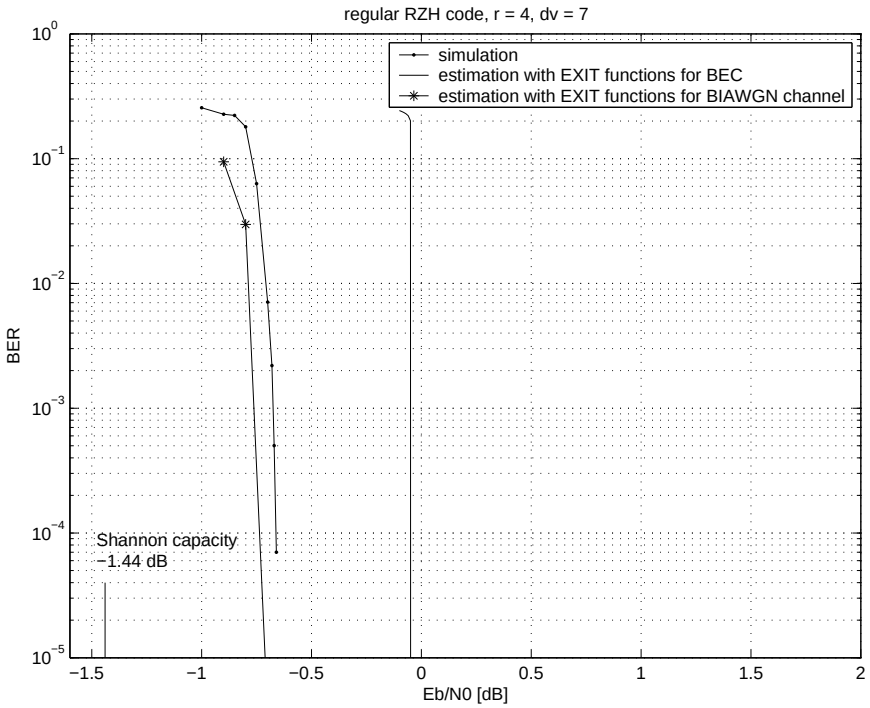


Figure 11: BER performance of regular RZH codes over BIAWGN channel.

regular RZH code with $r = 4$ and $\bar{d}_v = 7$ corresponding to a rate of $R_c = 0.0494$, is far away from the Shannon limit. To obtain better performance, we resort to irregular codes. For the code design, we set the BER threshold $\varepsilon = 10^{-4}$. Using the BEC EXIT functions, we are able to obtain a code at -0.96 dB with profile $f(x) = 0.547\,x^3 + 0.00430\,x^5 + 0.00760\,x^7 + 0.188\,x^9 + 0.0919\,x^{11} + 0.0984\,x^{13} + 0.0162\,x^{15} + 0.0148\,x^{17} + 0.0166\,x^{19} + 0.00380\,x^{21} + 0.00760\,x^{25} + 0.00380\,x^{29} + 0.00100\,x^{121}$. The simulated SNR threshold is -0.79 dB, which has a gain of 0.14 dB over the regular code. On the other hand, using the BIAWGN EXIT functions, we are able to optimize a code at -1.25 dB with profile $f(x) = 0.4353\,x^3 + 0.0072\,x^4 + 0.0016\,x^5 + 0.0099\,x^8 + 0.1759\,x^9 + 0.1829\,x^{10} + 0.0800\,x^{11} + 0.1071\,x^{12}$, which is only 0.19 dB away from the capacity and significantly better than the one obtained by BEC EXIT functions. The simulated SNR threshold for this code is -1.15 dB, which is only 0.29 dB away from the capacity and has a gain of 0.36 dB over the irregular code optimized by BEC

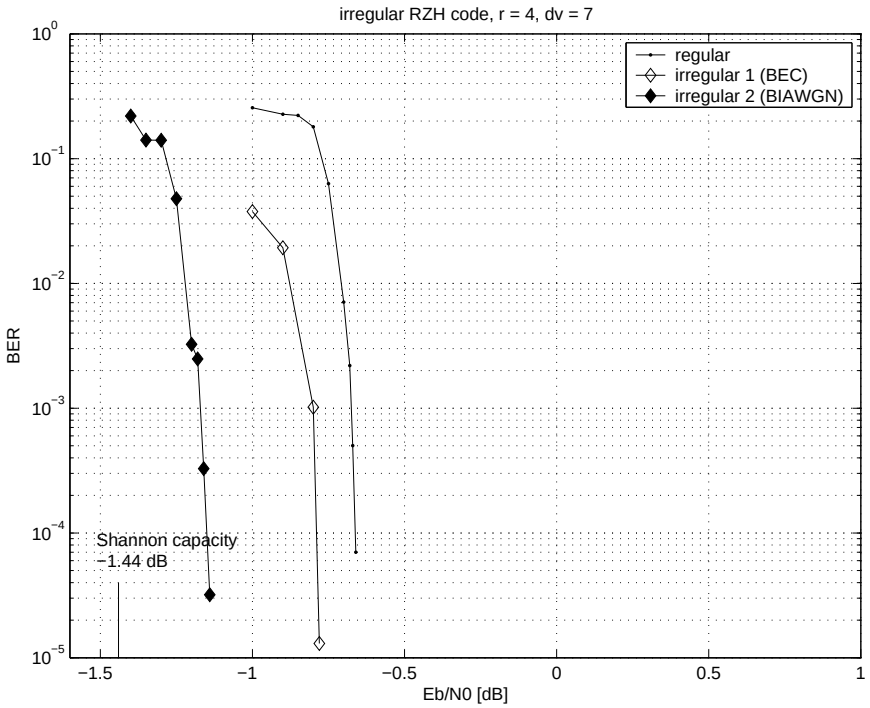


Figure 12: BER performance of irregular RZH codes over BIAWGN channel.

method and a gain of 0.49 dB over the regular one.

Example 5: (IRZH code ensemble optimization of with $r = 4$ and $\bar{d}_v = 8$ over BIAWGN channel) By setting $r = 4$ and $\bar{d}_v = 8$, the systematic RZH code has a rate of 0.0435. The Shannon limit for this code is -1.46 dB. Again, with EXIT functions over BIAWGN channel, we are able to optimize a code at -1.34 dB with profile $f(x) = 0.4044\,x^3 + 0.0026\,x^4 + 0.0048\,x^5 + 0.0245\,x^6 + 0.0453\,x^8 + 0.0669\,x^9 + 0.0296\,x^{10} + 0.0203\,x^{11} + 0.0985\,x^{12} + 0.3030\,x^{13}$, which is only 0.12 dB away from the capacity. BER performance of this code is depicted in Fig. 13. It is seen that the simulated SNR threshold for this code is -1.2 dB, which is only 0.26 dB away from the capacity and 0.4 dB away from the ultimate Shannon limit. Hence the EXIT functions for BIAWGN channels derived in this paper provide a good approximation in the low-rate region and can serve as a effective tool for the low-rate IRZH code design.

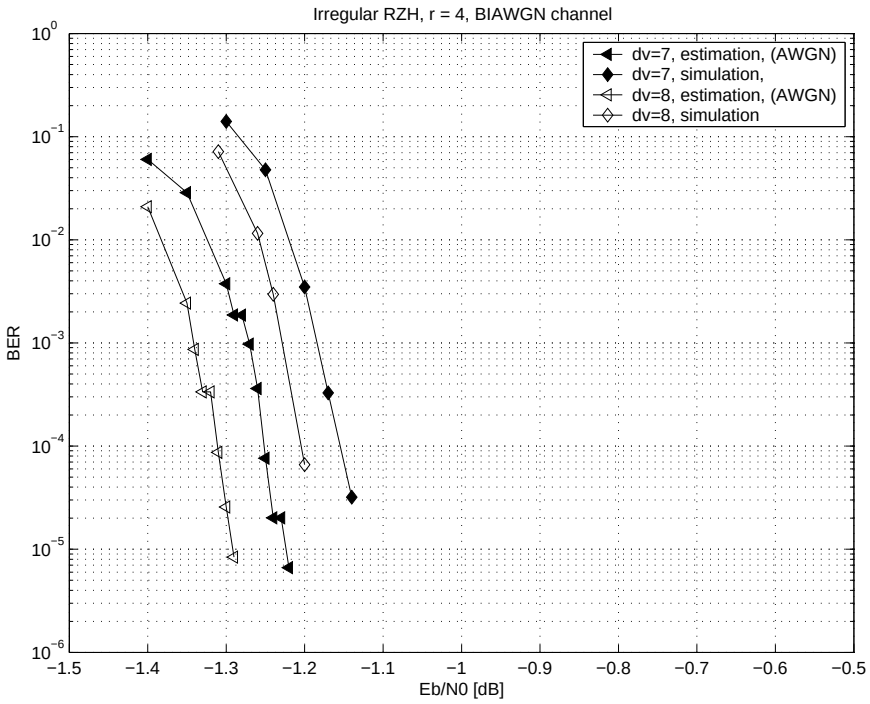


Figure 13: BER performance of irregular RZH codes over BIAWGN channel.

6 Conclusions

We have investigated the EXIT functions of Hadamard codes in the context of RZH codes with parallel decoding, and applied the obtained EXIT functions to the design of low-rate capacity-approaching irregular RZH codes. Two types of channels have been treated. For the BEC, exact analytical expressions for EXIT functions have been derived. For the BIAWGN channel, the approximate EXIT functions have been obtained in the form of a series expansion of multiple EXIT functions of single parity-check codes. With these EXIT functions, BERs for a given code profile can be easily estimated and differential evolution (DE) technique can be employed to optimize the degree profile for the given design parameters, e.g., the coding rate and the Hadamard order. It is seen that the obtained EXIT functions can serve as a practical tool for designing IRZH codes with parallel decoding. The techniques introduced in this paper can also be easily extended to the design of general low-density

parity-check codes where the check nodes are replaced by other linear block codes, e.g., Hadamard codes or Hamming codes.

References

[1] A. Ashikhmin, G. Kramer, and S. ten Brink. Extrinsic information transfer functions: A model and two properties. In *Proc. Conf. Inform. Sci. Syst. (CISS'02)*, pages 742–747, Princeton, NJ, Mar. 2002.

[2] A. Ashikhmin, G. Kramer, and S. ten Brink. Extrinsic information transfer functions: model and erasure channel properties. *IEEE Trans. Inform. Theory*, 50(11):2657–2673, Nov. 2004.

[3] M. Bingeman and A. Khandani. Symbol based turbo codes. *IEEE Commun. Lett.*, 3(10):285–287, Oct. 1999.

[4] R.G. Gallager. *Low-Density Parity-Check Codes*. Cambridge, MA: M.I.T. Press, 1963.

[5] C.R. Hartmann and L.D. Rudolph. An optimum symbol-by-symbol decoding rule for linear codes. *IEEE Trans. Inform. Theory*, 22(5):514–517, 1976.

[6] T. Helleseth, T. Kløve, and V. I. Levenshtein. On the information function of an error-correcting code. *IEEE Trans. Inform. Theory*, 43(2):549–557, Mar. 1997.

[7] H. Jin and R. McEliece. RA codes achieve AWGN channel capacity. In *Proc. Int. Symp. Applied Algebra & Error-Correcting Codes*, pages 14–19, Honolulu, HI, Nov. 1999.

[8] P. Komulainen and K. Pehkonen. Performance evaluation of superorthogonal turbo codes in AWGN and flat Rayleigh fading channels. *IEEE J. Select. Areas Commun.*, 16(2):196–205, Feb. 1998.

[9] W.K. Leung, G. Yue, L. Ping, and X. Wang. Concatenated zigzag Hadamard codes. *IEEE Trans. Inform. Theory*, 52(4):1711–1723, Apr. 2006.

[10] K. Li, L. Ping, and X. Wang. Analysis and optimization of interleave-division multiple-access communication systems. In *Proc. IEEE Int'l Conf. Acoust. Speech & Sig. Proc.*, Philadelphia, PA, Mar. 2005.

[11] K. Li, G. Yue, X. Wang, and L. Ping. Low-rate repeat-zigzag-Hadamard codes. In *Proc. IEEE Int'l Symp. Inform. Theory (ISIT'06)*, Seattle, WA., July 2006.

[12] F. MacWilliams and N. Sloane. *The Theory of Error-correcting Codes*. Amsterdam, The Netherlands: North-Holland, 1997.

[13] L. Ping, X. Huang, and N. Phamdo. Zigzag codes and concatenated zigzag codes. *IEEE Trans. Inform. Theory*, 47(2):800–807, Feb. 2001.

[14] L. Ping, W.K. Leung, and K.Y. Wu. Low-rate turbo-Hadamard codes. *IEEE Trans. Inform. Theory*, 49(12):3213–3224, Dec. 2003.

[15] K. Price and R. Storn. Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces. *J. Global Optimization*, 11:341–359, 1997.

[16] T. Richardson, A. Shokrollahi, and R. Urbanke. Design of capacity-approaching irregular low-density parity-check codes. *IEEE Trans. Inform. Theory*, 47(2):619–637, Feb. 2001.

[17] E. Sharon, A. Ashikhmin, and S. Litsyn. EXIT functions for continuous channels: Part I. constituent codes. *IEEE Trans. Commun.*, submitted for publication.

[18] R.M. Tanner. A recursive approach to low complexity codes. *IEEE Trans. Inform. Theory*, IT-27(5):533–547, Sept. 1981.

[19] S. ten Brink. Convergence of iterative decoding. *Electronics Letters*, 35(13):1117–1118, June 1999.

[20] S. ten Brink and G.K. Kramer. Design of repeat-accumulate codes for iterative detection and decoding. *IEEE Trans. Sig. Proc.*, 51(11):2764–2772, Nov. 2003.

[21] S. ten Brink, G.K. Kramer, and A. Ashikhmin. Design of low-density parity-check codes for modulation and detection. *IEEE Trans. Commun.*, 52(4):670–678, April 2004.

[22] M. Tuechler and J. Hagenauer. EXIT charts and irregular codes. In *Proc. Conf. Inform. Sci. Syst. (CISS'02)*, pages 748–753, Princeton, NJ, Mar. 2002.

[23] A. Viterbi. Orthogonal tree codes for communication in the presence of white Gaussian noise. *IEEE Trans. Commun. Technol.*, COM-15:238–242, Apr. 1967.

[24] R. Yarlagadda and J. Hershey. *Hadamard Matrix Analysis and Synthesis: With Applications to Communications and Signal/Image Processing.* Boston, MA: Kluwer, 1997.