

Joint Channel State Estimation and Decoding of Low-Density Parity-Check Codes on the Two-State Noiseless/Useless BSC Block Interference Channel

Wongkot Vijacksungsithi and Kim A. Winick

Abstract—We apply the density evolution technique to determine the thresholds of low-density parity-check codes when the sum-product algorithm is employed to perform joint channel state estimation and decoding. The channel considered is the two-state noiseless/useless BSC block interference channel, where a block of h consecutive symbols shares the same channel state which is either a noiseless BSC (crossover probability 0) or a useless BSC (crossover probability 1/2). The channel state is selected independently and at random from block to block according to a known prior distribution. The threshold of the joint channel state estimation/decoding scheme when utilized over such a channel is shown to be greatly superior to that of a decoder that makes no attempt to estimate the channel state. These results are also confirmed by simulation. The maximum likelihood performance of LDPC codes when used over this channel is investigated. Lower bounds on the error exponents of regular LDPC codes, when maximum likelihood decoded, are shown to be close to the random coding channel error exponent when the LDPC variable node degree is high.

Index Terms—LDPC codes, iterative decoding, channel state estimation, density evolution, error exponent.

I. INTRODUCTION

IN 1962, R. Gallager [1] proposed a class of codes, known as *Low-Density Parity-Check (LDPC) Codes*, that held the promise of achieving good performance while using low complexity iterative decoding algorithms. To investigate the performance of LDPC codes when optimally decoded, Gallager derived an upper bound on the error probability of maximum-likelihood decoding that is applicable to any specific code or to the average performance of a code ensemble used on binary-input symmetric memoryless channels provided that the weight distributions of the code or code ensemble are known. The maximum likelihood decoder, however, is prohibitively too complex to implement, and in practice, LDPC decoders usually employ suboptimal iterative algorithms that can provide good empirical performance while having a complexity that grows only linearly with the codeword length. In order to analyze the behavior of suboptimal iterative decoding algorithms, Gallager proposed a technique that keeps track of the probability of passing an incorrect message along an edge of the bipartite graph representation of the code at each iteration step. Gallager's analysis assumed that the bipartite

graph representation of the code does not contain any cycles of length less than the number of iterations, and a method to construct LDPC codes whose bipartite graph representations have arbitrary large diameters was also described. In 1998, Luby et al. [2] generalized Gallager's result to include irregular LDPC codes and showed that, for a randomly chosen code, the actual fraction of decoded bit errors at iteration step l converges with probability 1 to the expected number of decoded bit errors as the codeword length, n , of the code approaches ∞ . This result is known as the *Concentration Theorem* and the maximum value of the channel *noise* for which the expected fraction of incorrectly decoded codeword bits converges to zero as the number of message passing iterations increases is called the *threshold*. Subsequently, Richardson et al. [3] extended the result to message passing algorithms (such as belief propagation or sum-product) having infinite-sized alphabets. Richardson also proposed a technique, named *Density Evolution*, to numerically compute thresholds. This technique has been successfully employed to find codes that have performance close to channel capacity [4][5].

In this study, we apply the density evolution technique to investigate the performance of LDPC when the iterative message passing algorithm is used for *joint* channel state estimation/decoding over the two-state, noiseless/useless BSC, block interference channel. Related results have also been described in a recent conference presentation [6]. The message passing algorithm is derived based on a factor graph representation suggested by Wiberg [7] and first implemented by Worthen [8][9][10][11] for this application. The threshold of the joint channel state estimation/decoding scheme is numerically evaluated by density evolution and compared with capacity. These results are also confirmed by simulation, and substantial improvements in performance are demonstrated by using this joint channel state estimation/decoding approach. Finally, the maximum likelihood performance of LDPC codes used over a block interference channel is investigated. By applying the bounding technique originally proposed by Gallager [1] for binary input symmetric memoryless channels, a lower bound on the error exponent for LDPC codes used over the two-state noiseless/useless BSC block interference channel is derived from the average weight distribution spectra of these codes.

Joint channel state estimation/LDPC decoding algorithms based on factor graph representations of the combined channel and code have also been studied by others. García-Frías [12] and Rätzer [13] presented simulation results demonstrating

W. Vijacksungsithi was with the University of Michigan, Ann Arbor. He is now with CAT Telecom, Bangkok, Thailand (e-mail: vwongkot@catttelecom.com)

K. A. Winick is with the Department of Electrical Engineering and Computer science, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: winick@eecs.umich.edu)

that such an approach used for the Gilbert-Elliott channel could outperform receiver implementations that make no attempt to estimate the channel state. Eckford et al. [14][15][16] in a series of conference papers extended these results (for the Gilbert-Elliott channel) by computing thresholds of the joint estimation/decoding algorithms using density evolution.

The organization of this paper is as follows. In section II, we present the factor graph representation of LDPC codes when used over block interference channel along with the corresponding sum-product algorithm for joint channel state estimation and decoding. In section III, we describe the density evolution process for the algorithms presented in section II. Threshold values computed using density evolution along with simulation results are also presented. In section IV, a lower bound on the error exponent of LDPC codes when used over the two-state BSC useless/noiseless block interference channel, based on maximum likelihood decoding, is derived and evaluated. Finally, our conclusions are given in section V.

II. JOINT CHANNEL STATE ESTIMATION AND DECODING OF LDPC CODES ON BLOCK INTERFERENCE CHANNELS

A binary low-density parity-check (LDPC) code is a binary linear error correcting code specified by a parity-check matrix \mathbf{H} whose column and row weights are a small fraction of the codeword length. If the parity-check matrix has constant row and column weight, the code is said to be regular, and otherwise irregular. A regular LDPC code will be denoted by the triplet (n, j, k) where n is the codeword length, j is the column weight (also called the variable node degree) and k is the row weight. The rate, R , of such a code satisfies the condition $R \geq 1 - j/k$.

A *factor graph* representation [17] is a bipartite graph that expresses the structure of the factorization of a global function as the product of local functions. A factor graph has two types of nodes, there is a *variable node* for each variable x_i and a *factor node* for each local function f_j . An edge connecting variable node of x_i to factor node of f_j if and only if x_i is an argument of f_j .

Given a function $g(x_1, \dots, x_n)$, we are often interested in computing the *marginal functions* $g_i(x_i)$. For each $a \in A_i$, the value of $g_i(a)$ is obtained by summing the value of $g(x_1, \dots, x_n)$ over all possible values of its variables other than x_i , with $x_i = a$. For example, if we have some real-valued function $g(x_1, x_2, x_3, x_4)$, then

$$g_2(a) = \sum_{x_1 \in A_1} \sum_{x_3 \in A_3} \sum_{x_4 \in A_4} g(x_1, x_2 = a, x_3, x_4) \quad (1)$$

The operation described by (1) is called the *marginalization* of the global function $g(x_1, \dots, x_n)$ with respect to variable x_2 . The *sum-product* algorithm is an efficient message passing algorithm for computing these marginal functions. The sum-product algorithm is described in details in [7][17][18].

In this paper we will consider the *block interference channel* as described by McEliece et al. [19]. For each transmitted symbol x_t , this channel is completely characterized by its channel transition probabilities $p(y_t|x_t, s)$ where y_t is the corresponding channel output and s denotes the channel state. The channel is assumed to remain in the same state for each

block of h consecutive transmitted channel symbols. This state is selected at random, independently from block to block and independently from the transmitted symbol sequence, according to some known prior distribution. This channel model is a good representation for a frequency hopped system operating in the presence of partial band jamming. We will further restrict our analysis to two-state, binary-input channels, and in particular, to the two-state, noiseless/usesless, BSC. For each transmitted bit, this channel behaves as a BSC whose crossover probability is either zero (noiseless state) or $1/2$ (useless state). The prior probability that the channel is in the useless state will be denoted as ϵ_b .

Let \mathbf{x} be a codeword of a binary LDPC code \mathbb{C} of length n and rate R , let $\mathbf{y} = \{y_t\}$, $t = 0, 1, \dots, n-1$ be the received sequence resulting from transmitting $\mathbf{x} = \{x_t\}$, $t = 0, 1, \dots, n-1$ over the block interference channel and let s_i be the channel state corresponding to i^{th} block of h symbols. Assuming that all codewords \mathbf{x} are equiprobable, the maximum likelihood (ML) bit decoding rule is given by

$$\hat{x}_i = \arg \max_{x_i} \sum_{\mathbf{x}' \in \mathbb{C}: x'_i = x_i} \Pr(\mathbf{y}|\mathbf{x}') \quad (2)$$

$$= \arg \max_{x_i} \sum_{\mathbf{x}' \in \{0,1\}^n: x'_i = x_i} \mathbb{I}[\mathbf{x}' \in \mathbb{C}] \Pr(\mathbf{y}|\mathbf{x}') \quad (3)$$

where n is the length of the code and $\mathbb{I}[r]$ is the indicator function which equals 1 if the statement r is true and is 0 otherwise. We have

$$\begin{aligned} & \sum_{\mathbf{x}' \in \{0,1\}^n: x'_i = x_i} \mathbb{I}[\mathbf{x}' \in \mathbb{C}] \Pr(\mathbf{y}|\mathbf{x}') \\ &= \sum_{\mathbf{x}' \in \{0,1\}^n: x'_i = x_i} \mathbb{I}[\mathbf{x}' \in \mathbb{C}] \sum_{\mathbf{s}} p(\mathbf{s}, \mathbf{y}|\mathbf{x}') \end{aligned} \quad (4)$$

$$= \sum_{\mathbf{x}' \in \{0,1\}^n: x'_i = x_i} \mathbb{I}[\mathbf{x}' \in \mathbb{C}] \sum_{\mathbf{s}} p(\mathbf{y}|\mathbf{s}, \mathbf{x}') p(\mathbf{s}|\mathbf{x}') \quad (5)$$

$$\begin{aligned} &= \sum_{\mathbf{x}' \in \{0,1\}^n: x'_i = x_i} \sum_{\mathbf{s}} \mathbb{I}[\mathbf{x}' \in \mathbb{C}] \cdot \prod_{t=0}^{n-1} p(y_t|s_{\lfloor \frac{t}{h} \rfloor}, x'_t) \\ & \quad \cdot \prod_{j=0}^{\lceil \frac{n}{h} \rceil - 1} p(s_j) \end{aligned} \quad (6)$$

$$\begin{aligned} &= \sum_{\mathbf{x}' \in \{0,1\}^n: x'_i = x_i} \sum_{\mathbf{s}} \prod_{k=0}^{m-1} \mathbb{I}[\mathbf{x}' \text{ satisfies the } k^{\text{th}} \text{ parity check}] \\ & \quad \cdot \prod_{t=0}^{n-1} p(y_t|s_{\lfloor \frac{t}{h} \rfloor}, x'_t) \cdot \prod_{j=0}^{\lceil \frac{n}{h} \rceil - 1} p(s_j) \end{aligned} \quad (7)$$

where m is the number of rows in the parity-check matrix of \mathbb{C} , i.e. $m = n(1 - R)$. Eq. (6) follows from (5) by invoking the independence of the state from the transmitted symbols and from block to block. Note that (7) represents the marginalization of a global function that can be factored. The corresponding factor graph follows immediately and is shown in Fig. 1 for the case of a (2,3)-regular LDPC code with codeword length 9 and block length $h = 3$. Variable nodes are shown as filled circles and factor nodes are shown as open circles. Here, S_i is the variable node representing

the channel state, X_i is the variable node representing each bit in the codeword, C_i is the factor node representing each parity check, i.e. C_i corresponds to $[x'$ satisfies the i^{th} parity check equation], U_i is the factor node corresponding to the prior probability distributions on the channel states and W_i is the factor node that represents $p(y_i|s_{\lfloor \frac{i}{h} \rfloor}, x'_i)$. Note that we suppress the variable nodes for the received symbol Y_i and incorporate this information directly into node W_i .

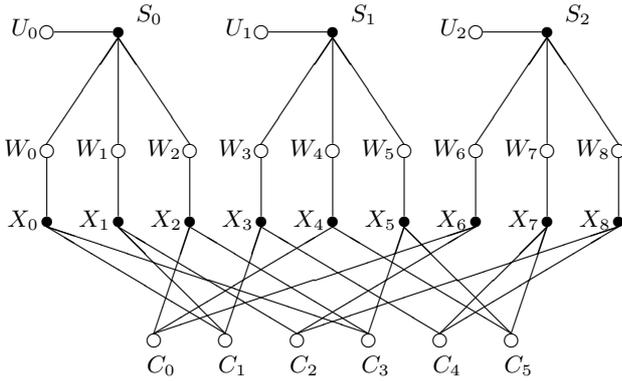


Fig. 1. Factor graph representing (2,3)-regular LDPC code over the block interference channel with $h = 3$ and $m = 6$

We can apply the sum-product algorithm to approximate the likelihood function of each codeword bit as described by (4). In the sum-product algorithm, the message sent along any edge connected to X_i at message-passing cycle l equals or approximates (depending on whether the graph has cycles) the a posteriori probability of bit x_i conditioned on all the information available in the subgraph that contributes to that message computation. The message consists of two a posteriori probabilities P_0 for $x_i = 0$ and P_1 for $x_i = 1$. It will be more convenient to represent these two numbers, and hence the message, by the single quantity $\log(P_0/P_1)$. This will be referred to as the log-likelihood ratios (LLRs) representation. If the channel possesses more than 2 states, it is not possible to also use an LLR message representation along edges $(S_{\lfloor \frac{i}{h} \rfloor}, W_i)$ and (U_i, S_i) . Here, however, we will only consider the case when the channel has 2 states, labeled 0 and 1. This assumption enables us to also pass the state message as a single number in an LLR representation along each edge connected to a state node S_i . The sum-product message update rules for a two-state channel when all messages are in an LLR format can be explicitly written as follows

- **Variable-to-factor node message update**

$$\mu_{S_{\lfloor \frac{i}{h} \rfloor} \rightarrow W_i} = \mu_{U_{\lfloor \frac{i}{h} \rfloor} \rightarrow S_{\lfloor \frac{i}{h} \rfloor}} + \sum_{W_j \in \text{Ne}(S_{\lfloor \frac{i}{h} \rfloor}) \setminus W_i} \mu_{W_j \rightarrow S_{\lfloor \frac{i}{h} \rfloor}} \quad (8)$$

$$\mu_{X_i \rightarrow W_i} = \sum_{C_j \in \text{Ne}(X_i)} \mu_{C_j \rightarrow X_i} \quad (9)$$

$$\mu_{X_i \rightarrow C_j} = \mu_{W_i \rightarrow X_i} + \sum_{C_k \in \text{Ne}(X_i) \setminus C_j} \mu_{C_k \rightarrow X_i} \quad (10)$$

- **Factor-to-variable node message update**

$$\mu_{U_i \rightarrow S_i} = \log \frac{p(s_i = 0)}{p(s_i = 1)} \quad (11)$$

$$\mu_{W_i \rightarrow S_{\lfloor \frac{i}{h} \rfloor}} = \log \frac{\sum_{x_i \in \{0,1\}} p(y_i|x_i, s_{\lfloor \frac{i}{h} \rfloor} = 0) \cdot \frac{e^{(1-x_i) \cdot \mu_{X_i \rightarrow W_i}}}{1+e^{\mu_{X_i \rightarrow W_i}}}}{\sum_{x_i \in \{0,1\}} p(y_i|x_i, s_{\lfloor \frac{i}{h} \rfloor} = 1) \cdot \frac{e^{(1-x_i) \cdot \mu_{X_i \rightarrow W_i}}}{1+e^{\mu_{X_i \rightarrow W_i}}}} \quad (12)$$

$$\mu_{W_i \rightarrow X_i} = \log \frac{\sum_{s_{\lfloor \frac{i}{h} \rfloor} \in \{0,1\}} p(y_i|x_i = 0, s_{\lfloor \frac{i}{h} \rfloor}) \cdot \frac{e^{(1-s_{\lfloor \frac{i}{h} \rfloor}) \cdot \mu_{S_{\lfloor \frac{i}{h} \rfloor} \rightarrow W_i}}}{1+e^{\mu_{S_{\lfloor \frac{i}{h} \rfloor} \rightarrow W_i}}}}{\sum_{s_{\lfloor \frac{i}{h} \rfloor} \in \{0,1\}} p(y_i|x_i = 1, s_{\lfloor \frac{i}{h} \rfloor}) \cdot \frac{e^{(1-s_{\lfloor \frac{i}{h} \rfloor}) \cdot \mu_{S_{\lfloor \frac{i}{h} \rfloor} \rightarrow W_i}}}{1+e^{\mu_{S_{\lfloor \frac{i}{h} \rfloor} \rightarrow W_i}}}} \quad (13)$$

$$\mu_{C_i \rightarrow X_j} = 2 \tanh^{-1} \left(\prod_{X_k \in \text{Ne}(C_i) \setminus X_j} \tanh \left(\frac{\mu_{X_k \rightarrow C_i}}{2} \right) \right) \quad (14)$$

In the case of the joint channel state estimation and decoding, $p(s_i)$ is the prior probability of the channel state. If this prior information is not available, all channel states are assumed equiprobable. In the case when channel state is known to the receiver (i.e. perfect channel state side information (CSI)), $p(s_i)$ becomes an indicator function with $p(s_i) = 1$ if s_i is the channel state of block i and 0 otherwise. For both cases, the message update in (11) needs to be done only at the initialization and no further update is required. Note that there is no message update from S_i to U_i , because each node U_i has only one edge, and according to the sum-product algorithm, the message it receives on that single edge cannot be used to calculate an updated message for that edge.

The messages leaving each node are updated using (8)-(14) according to a schedule. When the graph has cycles, the result obtained from any schedule is just an approximation of the desired marginals and there is no general rule for optimal scheduling. A number of scheduling methods have been used including flooding and sequential. With *flooding scheduling*, a new message is passed along each edge in the graph in both directions in parallel during each message passing iteration, and message passing is stopped after some fixed number of iterations are performed or when some other specified conditions are satisfied. In sequential scheduling the nodes are partitioned into disjoint sets. The message passing is done within a set on a set by set basis according to some specified order. When a set of nodes is scheduled to be updated, all nodes in that set are updated in parallel. In the work reported here, we employ sequential scheduling with updates performed in the following order $\mu_{S_{\lfloor \frac{i}{h} \rfloor} \rightarrow W_i}, \mu_{W_i \rightarrow X_i}, \mu_{X_i \rightarrow C_j}, \mu_{C_j \rightarrow X_i}, \mu_{X_i \rightarrow W_i}, \mu_{W_i \rightarrow S_{\lfloor \frac{i}{h} \rfloor}}, \mu_{S_{\lfloor \frac{i}{h} \rfloor} \rightarrow W_i}$ and so on.

III. PERFORMANCE OF LDPC CODES EVALUATED BY DENSITY EVOLUTION ON THE BLOCK INTERFERENCE CHANNEL

It is shown in [3] that when a binary-input memoryless channel and message passing algorithm satisfies a certain set of symmetry conditions then the codeword bit positions that are decoded in error for a fixed channel realization will not depend on which codeword is transmitted. This result can be extended to include the joint channel state estimation/decoding algorithm given by (8)-(14) when used over the two-state, noiseless/useless BSC, block interference channel [20]. Thus, when evaluating bit error rate performance, we can assume without loss of generality that the all zero codeword is transmitted.

It is shown in [3] that if we pick a code at random with equal probability from an LDPC code ensemble, then the fraction of codeword bits decoded in error with sum-product decoding on a memoryless channel will converge with probability 1 to the probability that a message error occurs along any specified directed edge leaving a codeword bit node and terminating on a check node. Assuming that the all zero codeword was transmitted, a message error occurs when the message (in LLR format) leaving a codeword bit node is negative. Thus the average bit-error-rate (BER) performance of the LDPC code ensemble, when used with sum-product decoding over a memoryless channel, can be found once the probability density of a message leaving a codeword bit node is known. This result, known as the concentration theorem, can be extended [20] to the class of block interference channels with finite block length h by using a slightly modified version of the edge exposure martingale argument given in [3] for memoryless channels.

If the diameter of the factor graph is sufficiently large then the messages entering any given node at message passing iteration step l will be independent random variables, and it can be shown that as the LDPC codeword length increases, almost all of the corresponding factor graphs will have sufficiently large diameter [3]. Furthermore these random variables will be identically distributed if the all zero-codeword was transmitted. Under such conditions the density evolution technique can be used to find the probability density of the outgoing message for the node [3]. In order to avoid numerical precision problems, we quantize each message and treat the messages as discrete random variables when performing the density evolution calculations as described in [5]. The density update rules at channel state nodes S_i 's and codeword bit nodes X_i 's involve the summation of independent random variables, and thus the resulting output message density is a convolution of a set of probability mass functions. The density update rules at a factor node are more involved. In order to compute the probability of a quantized outgoing message, we need to sum over the joint probability mass functions (pmf's) of all possible combinations of incoming messages multiplied by the local function that results in that particular output message. For the check nodes C_i 's, an efficient way to do the density update is given in [5]. We can then summarize the density update rules for (j, k) -regular LDPC codes on a

two-state block interference channel as follows

$$p_{U \rightarrow S}(\xi) = \delta \left(\xi - \log \frac{p(s_i = 0)}{p(s_i = 1)} \right) \quad (15)$$

$$p_{S \rightarrow W}(\xi|s) = p_{U \rightarrow S}(\xi) \otimes^{\otimes_{h-1}} p_{W \rightarrow S}(\xi|s) \quad (16)$$

$$p_{W \rightarrow X}(\xi|s) = \sum_{\substack{\xi', y \\ f_2(\xi', y) = \xi}} p_{S \rightarrow W}(\xi'|s) \cdot p(y|x = 0, s) \quad (17)$$

$$p_{W \rightarrow X}(\xi) = \sum_{s \in \{0,1\}} p_{W \rightarrow X}(\xi|s) p(s) \quad (18)$$

$$p_{X \rightarrow C}(\xi) = p_{W \rightarrow X}(\xi) \otimes^{\otimes_{j-1}} p_{C \rightarrow X}(\xi) \quad (19)$$

$$p_{C \rightarrow X}(\xi) = \mathcal{R}^{k-1}(p_{X \rightarrow C}(\xi)) \quad (20)$$

$$p_{X \rightarrow W}(\xi) = \otimes^{\otimes_j} p_{C \rightarrow X}(\xi) \quad (21)$$

$$p_{W \rightarrow S}(\xi|s) = \sum_{\substack{\xi', y \\ f_1(\xi', y) = \xi}} p_{X \rightarrow W}(\xi') \cdot p(y|x = 0, s) \quad (22)$$

where $\delta(x)$ is the dirac delta function, \otimes denotes the convolution operation, \otimes^i denotes i -fold self convolution, $\mathcal{R}^{k-1}(p_{X \rightarrow C}(\xi))$ represents the notation for the average message density obtained by evolving the density $p_{X \rightarrow C}(\xi)$ through a check node of degree k as described in [5]. The message update rules (12) and (13) are incorporated into (17) and (22) through the terms $f_1(\xi', y)$ and $f_2(\xi', y)$, respectively defined below

$$f_1(\xi', y) = \log \frac{\sum_{x \in \{0,1\}} p(y|x, s = 0) \cdot \frac{e^{\xi'(1-x)}}{1+e^{\xi'}}}{\sum_{x \in \{0,1\}} p(y|x, s = 1) \cdot \frac{e^{\xi'(1-x)}}{1+e^{\xi'}}} \quad (23)$$

$$f_2(\xi', y) = \log \frac{\sum_{s \in \{0,1\}} p(y|x = 0, s) \cdot \frac{e^{\xi'(1-s)}}{1+e^{\xi'}}}{\sum_{s \in \{0,1\}} p(y|x = 1, s) \cdot \frac{e^{\xi'(1-s)}}{1+e^{\xi'}}} \quad (24)$$

Note that the density updates (15)-(22) are performed in the order given. This corresponds to the message update schedule described in the last sentence of section II. The probability $P_e^{(l)}$ of a bit-to-check node message error at message passing iteration step l is given by

$$P_e^{(l)} = \sum_{\xi \leq 0} p_{X \rightarrow C}^{(l)}(\xi) \quad (25)$$

By utilizing the density evolution technique the threshold ($\epsilon_b^{\text{thresh}}$), which is defined to be the maximum value of ϵ_b for which $P_e^{(l)}$ converges to zero as the number of iterations increases, can be numerically evaluated. Note that for regular LDPC codes, $\epsilon_b^{\text{thresh}}$ will be a function of column and row weight (j and k) of the parity check matrix. The complexity of this calculation is independent of the codeword length and thus this technique is more efficient than simulation for codes having large codeword lengths. Furthermore the concentration theorem implies that the bit error probability goes to zero for almost all regular LDPC codes of which parity check

matrices have column weight j and row weight k and sufficient codeword length as the number of message passing iterations increases provided $\epsilon_b < \epsilon_b^{\text{thresh}}(j, k)$.

Next, we present some simulation results along with the corresponding threshold values computed by density evolution. The channel considered is a two-state block interference channel where the first channel state corresponds to the noiseless BSC with crossover probability 0 and the second state corresponds to the useless BSC with crossover probability 1/2. When the receiver has perfect CSI, the two-state noiseless/useless BSC block interference channel reduces to a 2^h -ary erasure channel, and its capacity is given by [19]

$$C_{\text{CSI}} = 1 - \epsilon_b \quad (26)$$

where ϵ_b equals the probability that the channel is in the useless state.

When channel state is unknown to the receiver, the capacity is given by [19]

$$C_{\text{NCSI}} = (1 - \epsilon_b + \epsilon_b 2^{-h}) - \frac{1}{h} [H_2(\epsilon_b - \epsilon_b 2^{-h}) + (\epsilon_b - \epsilon_b 2^{-h}) \log_2(1 - 2^{-h})] \quad (27)$$

where

$$H_2(\epsilon) = -\epsilon \log_2(\epsilon) - (1 - \epsilon) \log_2(1 - \epsilon) \quad (28)$$

As h approaches infinity the channel state can be determined exactly by the receiver, and thus $C_{\text{NCSI}} \rightarrow C_{\text{CSI}}$ as $h \rightarrow \infty$

Many practical systems combat channel memory by interleaving the transmitted symbols so that after deinterleaving, the channel state appears to be independent from symbol to symbol, i.e., $h = 1$. The capacity of the two-state noiseless/useless BSC channel with $h = 1$ is given by

$$C_{\text{interleave}} = 1 - H_2\left(\frac{\epsilon_b}{2}\right) \quad (29)$$

In Fig. 2, $C_{\text{interleave}}$, C_{NCSI} ($h = 2, 10, 20$) and C_{CSI} of the two-state noiseless/useless BSC channel are shown for comparison. The gap between C_{NCSI} and $C_{\text{interleave}}$ indicates the potential gain in performance that can be realized by estimating the channel state.

We have computed the threshold ϵ_b values for this two-state noiseless/useless BSC channel operating at a rate of 1/2 bit per channel use using density evolution under the following three different conditions: (1) joint channel state estimation/decoding, (2) no channel state estimation, and (3) perfect CSI. The results are presented in Table I for rate 1/2 codes together with the maximum achievable ϵ_b as promised by channel capacity, and the ϵ_b achievable based on simulation results ($P_b \approx 10^{-5}$) using a regular, rate 1/2, LDPC code of codeword length 4000. As can be seen from the table, the thresholds of the iterative channel state estimation/decoding schemes are substantially better than those obtained using a decoder that makes no attempt to estimate the channel state. As the block length h increases, received symbols are more correlated and this correlation can be efficiently exploited by the joint channel state estimation/decoding receiver. Note that threshold calculations indicate that the gap between C_{NCSI} and $C_{\text{interleave}}$ can be substantially narrowed by using joint channel

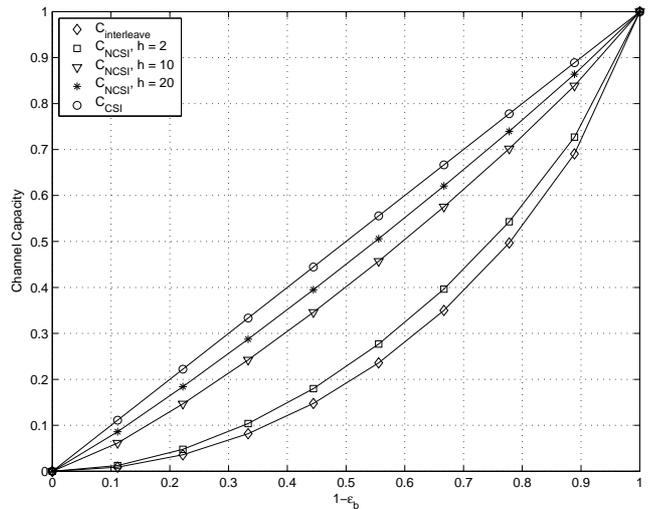


Fig. 2. $C_{\text{interleave}}$, C_{NCSI} ($h=2,10,20$) and C_{CSI} of the two-state noiseless/useless BSC block interference channel

TABLE I
THRESHOLDS ON ϵ_b OF LDPC WITH RATE $\frac{1}{2}$ ON THE TWO-STATE NOISELESS/USELESS BSC BLOCK INTERFERENCE CHANNEL

Code/Decode Method/Block Length	Capacity	Threshold	Simulation
(3,6)-reg., No State Est., $h = 2$	0.220 ⁽²⁾	0.168	0.090
(3,6)-reg., No State Est., $h = 10$	0.220 ⁽²⁾	0.168	0.060
(3,6)-reg., Soft Est./Dec., $h = 2$	0.252 ⁽¹⁾	0.203	0.160
(3,6)-reg., Soft Est./Dec., $h = 10$	0.403 ⁽¹⁾	0.315	0.223
(3,6)-reg., Perfect CSI, $h = 2$	0.500 ⁽³⁾	0.429	0.381
(3,6)-reg., Perfect CSI, $h = 10$	0.500 ⁽³⁾	0.429	0.334
(4,8)-reg., Soft Est./Dec., $h = 10$	0.403 ⁽¹⁾	0.254	0.170

⁽¹⁾ C_{NCSI} (27), ⁽²⁾ $C_{\text{interleave}}$ (29), ⁽³⁾ C_{CSI} (26)

state estimation/decoding with LDPC codes. For example, with $h = 10$ $C_{\text{NCSI}} = 0.5 \rightarrow \epsilon_b = 0.403$, $C_{\text{interleave}} = 0.5 \rightarrow \epsilon_b = 0.22$ and (3,6)-regular LDPC ($R = 0.5$) $\rightarrow \epsilon_b^{\text{thresh}} = 0.315$. This favorable result is obtained even though optimization of the node degree sequence of the LDPC code was not undertaken. By using an irregular LDPC code with an optimized node degree sequence we expect to see further improvements. The gaps between the simulation results and threshold values are due to the use of LDPC codes with relatively short codeword lengths. By using codes with sufficiently long codeword lengths, the simulation results will approach the threshold values. In Fig. 3, simulation results are shown for the (3,6)-regular LDPC code cases corresponding to the entries given in Table I. The (3,6)-regular LDPC code which we used was designed by David Mackay [21] and has a codeword length of 4000. It can be seen from the figure that joint channel state estimation/decoding performs closer to the system with perfect CSI as h increases. For the case of no state estimation and perfect CSI, channels with block lengths of 2 or 10 achieve the same thresholds (see Table I). The simulation performance for this case (see Fig. 3), however, depends on the block length, and unlike the case of joint channel state estimation/decoding, the performance degrades as the block length increases for a fixed codeword length. This trend is due to the fact that when the codeword

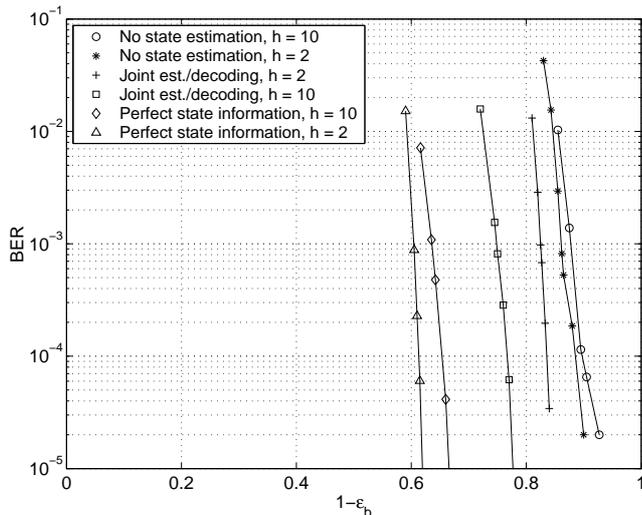


Fig. 3. Simulation performances of a (3,6)-regular LDPC, $n = 4000$ on the two-state noiseless/useless BSC block interference channel

length is fixed, the number of independent blocks decreases as the block length increases. A smaller number of independent blocks yields a larger probability that an atypically large number of bad channel states occurs. Thus, a higher decoding error rate occurs for schemes that do not estimate the channel state and for schemes with perfect CSI as the block length increases and the codeword length is held fixed. The situation is different for joint channel state estimation/decoding, since increased correlation occurs between received symbols as the block length increases for a fixed codeword length, and this correlation can be efficiently exploited by the receiver. Nevertheless, it is also possible under some situations that the increased correlation between symbols is insufficient to compensate for the decreasing number of independent blocks, and a degradation in performance will result. Note that these results are only physically meaningful when $h \ll n$. When h becomes an appreciable fraction of n , *outage* rates as opposed to bit-error-rates are the appropriate performance measure.

The most important use of density evolution is to optimize the code degree sequence as well as other decoding parameters. The computational complexity of the density evolution technique does not depend on the codeword length, and thus it provides a more efficient way to optimize the code degree sequence than using simulations. In Fig. 4, we compare the thresholds and simulation performances of (3,6)-regular and (4,8)-regular LDPC when employing joint channel state estimation/decoding over a channel with block length 10. The two codes have the same rate but the (3,6)-regular code outperforms the (4,8)-regular code. These results coincide with those predicted by threshold values obtained by density evolution and given in Table I. Also illustrated in the figure is the fact that code performance gets closer to its threshold value as the codeword length increases.

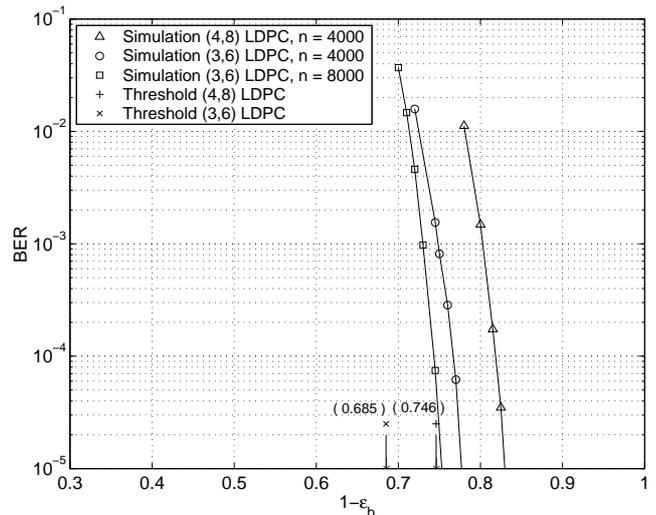


Fig. 4. Performance comparison between (3,6)-regular LDPCs, $n = 4000, 8000$ and a (4,8)-regular LDPC, $n = 4000$ on the two-state noiseless/useless BSC block interference channel with $h = 10$ when employing joint channel state estimation and decoding

IV. MAXIMUM LIKELIHOOD PERFORMANCE OF LDPC OVER THE BLOCK INTERFERENCE CHANNEL

In this section, we will derive information theoretic bounds for the performance of an optimal maximum likelihood receiver on the two-state noiseless/useless BSC block interference channel. The bounds will be derived based on the assumption that the code is restricted to be in an ensemble of (j, k) -regular LDPC codes. The quantity of interest is the *error exponent* of the code family. The error exponent describes the rate at which the probability of codeword error of the best sequence of codes in the ensemble decreases as the codeword length increases. When the code ensemble is enlarged to include all possible codes (i.e., not just LDPC codes), this error exponent $E(R)$ will depend *only* on the channel and will be referred to as *channel error exponent*. For a (j, k) -regular LDPC ensemble, we will define the LDPC error exponent as

$$E(j, k) = \lim_{n \rightarrow \infty} \sup \left[-\frac{1}{n} \log P_{opt}(j, k, n) \right] \quad (30)$$

where $P_{opt}(j, k, n)$ is the codeword error probability of the best (j, k) -regular LDPC code (i.e., the (j, k) -regular LDPC code with the lowest probability of codeword error assuming maximum likelihood decoding) of codeword length n .

In [19], an argument has been made that the block interference channel can be treated as a memoryless channel where each block of h symbols from the original channel forms a single input of the corresponding super channel. Thus, the coding theorem and all the bounding techniques developed for memoryless channels can be applied directly to the block interference channel with and without perfect CSI. Evaluating these bounds, however, is usually quite involved due to the size of the input and output alphabets, which grow exponentially with the hop length h . The computation is not particularly burdensome for the case of the two-state noiseless/useless BSC block interference channel. Using standard techniques

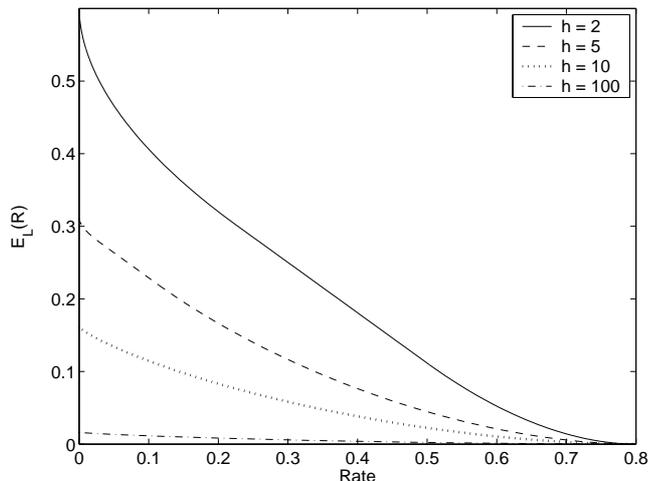


Fig. 5. Lower bound on the channel error exponent of the two-state noiseless/useless BSC block interference channel, $\epsilon_b = 0.2$ when channel state is known

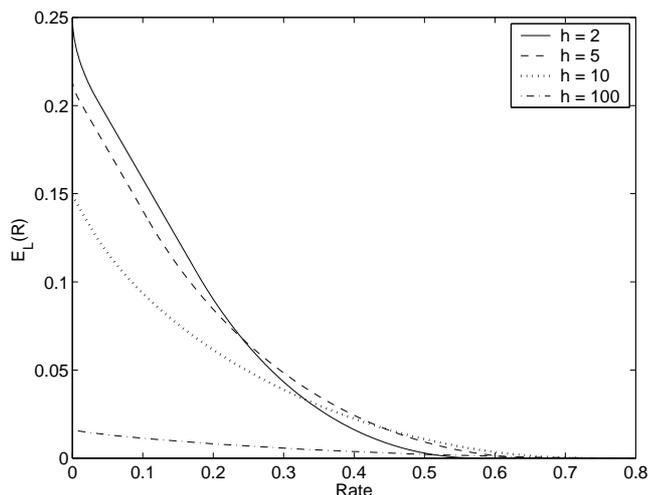


Fig. 6. Lower bound on the channel error exponent of the two-state noiseless/useless BSC block interference channel, $\epsilon_b = 0.2$ when channel state is unknown

[22][23], lower bounds (i.e., the random coding channel exponent $E_L(R)$) on the channel error exponent can be derived. These bounds are shown in Fig. 5 and 6 for the cases where the channel state is known and unknown to the receiver, respectively.

As illustrated in the figures, the lower bound on the channel error exponent at any rate is monotonically decreasing with the hop length when the channel state is known to the receiver. This result provides some theoretical justification for the increase in error probability observed in section III for the channel with known CSI as the hop length increases. The monotonicity described above is not present at all rates for the case when channel state is unknown to the receiver. In fact, the hop length at which the lower bound on the channel error exponent is maximum varies as a function of rate. This observation illustrates the tradeoff that exists between the number of independent hops and the ability of the receiver to estimate the channel state as previously described. A proof of

the monotonicity of the channel error exponent when channel state is known to the receiver together with a proof of existence of an optimal hop length when channel state is unknown to the receiver are given in [11] for a general class of block interference channels.

A general case analysis of the LDPC code error exponent for the block interference channel is usually quite involved, however, the derivation can be simplified for some special cases such as the two-state noiseless/useless BSC channel described in section II. The basic bounding technique that we apply here was developed by Gallager [1] and Fano [24]. We have made some modifications to extend the technique to the block interference channel.

In [19], it was argued that the block interference channel can be treated as a memoryless channel where each block of h symbols from the original channel forms a single input of the corresponding super channel. Let $\mathbf{x}_0 = x_{00}, x_{10}, \dots, x_{(n'-1)0}$ be the codeword of length n transmitted over the super channel, and let $\mathbf{y} = y_0, y_2, \dots, y_{n'-1}$ be the corresponding received sequence, where $n' = n/h$ is the number of times the channel is used independently during transmission and we assume h divides n . The input symbol x_{i0} and the output symbol y_i are letters of the input and output alphabet of the memoryless super channel that represents the block interference channel. Let the other codewords be $\mathbf{x}_j = x_{0j}, x_{1j}, \dots, x_{(n'-1)j}$ where $j = 1, 2, \dots, M-1$ and M is the number of codewords. Using maximum-likelihood decoding, a decoding error occurs iff (we consider a ‘‘tie’’ an error)

$$\sum_{i=0}^{n'-1} \log P(y_i | x_{ij}) \geq \sum_{i=0}^{n'-1} \log P(y_i | x_{i0}) \quad (31)$$

for some $j \neq 0$. We define the symbol discrepancy between input symbol x_{ij} and output symbol y_i as

$$d(x_{ij}, y_i) = \log \frac{1}{P(y_i | x_{ij})} \quad (32)$$

and the sequence discrepancy between \mathbf{x} and \mathbf{y} as

$$D(\mathbf{x}_j, \mathbf{y}) = \sum_{i=0}^{n'-1} d(x_{ij}, y_i) \quad (33)$$

By applying the union bound to (31), the codeword error probability can be bounded by

$$\begin{aligned} P_e &\leq \Pr \left\{ \bigcup_{j=1}^{M-1} [D(\mathbf{x}_j, \mathbf{y}) \leq D(\mathbf{x}_0, \mathbf{y})] \right\} \\ &\leq \Pr[D(\mathbf{x}_0, \mathbf{y}) > n\delta] \\ &\quad + \sum_{j=1}^{M-1} \Pr[D(\mathbf{x}_0, \mathbf{y}) \leq n\delta; D(\mathbf{x}_j, \mathbf{y}) \leq D(\mathbf{x}_0, \mathbf{y})] \end{aligned} \quad (34)$$

Invoking the Chernoff bound, we get, for any $z \geq 0$

$$\Pr[D(\mathbf{x}_0, \mathbf{y}) > n\delta] \leq \left(\prod_{i=0}^{n'-1} \phi_i(z) \right) \exp(-nz\delta) \quad (36)$$

where $\phi_i(z)$ is the moment generating function defined by

$$\phi_i(z) = \sum_{y_i} P(y_i|x_{i0})^{1-z} \quad (37)$$

By symmetry, it can be verified that $\phi_i(z)$ does not depend on the value of x_{i0} . Thus the value of the bound is the same for any transmitted codeword \mathbf{x}_0 and

$$\prod_{i=0}^{n'-1} \phi_i(z) = [g(z)]^{n'} \quad (38)$$

where

$$\phi_i(z) = g(z) \equiv \sum_{y_i} p(y_i|x_{i0})^{1-z} \quad (39)$$

for any value of x_{i0}

In order to bound the second term on the right hand side of (35), we again invoke the Chernoff bound. Let v_i and w_i be n' pairs of random variables with joint probability mass function $P_i(v_i, w_i)$ and joint moment generating function

$$h_i(r, \xi) = \sum_{v_i} \sum_{w_i} \exp(rv_i + \xi w_i) P_i(v_i, w_i) \quad (40)$$

Let V and W each be sums of independent random variables as defined below

$$V = \sum_{i=0}^{n'-1} v_i, \quad W = \sum_{i=0}^{n'-1} w_i \quad (41)$$

we have

$$\Pr(V \leq n\tilde{v}; W \leq n\tilde{w}) \leq \left(\prod_{i=0}^{n'-1} h_i(r, \xi) \right) \exp[-n(r\tilde{v} + \xi\tilde{w})] \quad (42)$$

for any $r \leq 0$, $\xi \leq 0$. It follows from (40) with

$$v_i = d(x_{i0}, y_i), \quad w_i = d(x_{ij}, y_i) - d(x_{i0}, y_i) \quad (43)$$

that

$$h_i(r, \xi) = \sum_{y_i} \left\{ P^{1-r}(y_i|x_{i0}) \left[\frac{P(y_i|x_{i0})}{P(y_i|x_{ij})} \right]^\xi \right\} \quad (44)$$

If $x_{ij} = x_{i0}$, then (44) becomes

$$h_i(r, \xi) = g(r) = \sum_{y_i} \exp[rd(x_{i0}, y_i)] P(y_i|x_{i0}) \quad (45)$$

where $g(r)$ is as defined in (39). By symmetry, one can verify that $h_i(r, \xi)$ does not depend on the particular x_{i0} or x_{ij} . Thus without loss of generality we can assume that $x_{i0} = [0 \dots 0]$ and $x_{ij} = [1 \dots 1]$. With r held fixed, $h_i(r, \xi)$ given by (44) achieves its minimum value when $\xi = \frac{r-1}{2}$, and thus for any $x_{ij} \neq x_{i0}$,

$$h(r) \equiv h\left(r, \frac{r-1}{2}\right) \quad (46)$$

$$= \sum_{y_i} [P(y_i|x = [0 \dots 0])P(y_i|x = [1 \dots 1])]^{\frac{1-r}{2}} \quad (47)$$

Hence, if \mathbf{x}_0 and \mathbf{x}_j differ in m symbols and agree in $n' - m$ symbols, we will have

$$\Pr[D(\mathbf{x}_j, \mathbf{y}) \leq n\delta; D(\mathbf{x}_j, \mathbf{y}) \leq D(\mathbf{x}_0, \mathbf{y})] \leq [h(r)]^m [g(r)]^{n'-m} e^{-nr\delta} \quad (48)$$

for any $r \leq 0$. We will assume without loss of generality that the all 0 codeword is transmitted as \mathbf{x}_0 , and that \mathbf{x}_j is a codeword of some weight $l \neq 0$. We will evaluate the error probability bound (48) by using the weight spectrum of the code. In order to do so, we need to establish the relation between the binary weight l of the codeword and the number of super symbols m where \mathbf{x}_0 and \mathbf{x}_j differ. Let $P_{\text{box}}(l, m, n)$ denote the probability that any randomly chosen codeword of weight l and length n will differ from the all zero codeword in m super symbol positions. The symbol pairs x_{ij} and x_{i0} are identical when they agree in every one of h binary positions in the super symbol. The LDPC code ensemble to be considered includes all possible permutations of the codeword bits. This implies that, given a codeword of weight l in some randomly chosen code from the LDPC code ensemble, the ones are equally likely to be in any of the n positions. Thus computing $P_{\text{box}}(l, m, n)$ is equivalent to placing l indistinguishable balls at random into $n' = n/h$ distinguishable boxes, each box having a capacity to hold h balls, and asking for the probability $P_{\text{box}}(l, m, n)$ that m of the boxes contain at least one ball. For l, m and h fixed, an exponentially tight bound in n on the probability can be given by

$$\frac{\binom{n'}{m} \left[\min \left\{ \binom{h}{\lfloor \frac{l}{m} \rfloor}, \binom{h}{\lceil \frac{l}{m} \rceil} \right\} \right]^m}{\binom{n}{l}} \leq P_{\text{box}}(l, m, n) \leq \frac{\binom{n'}{m} \binom{mh}{l}}{\binom{n}{l}} \quad (49)$$

for $m \geq \lceil \frac{l}{h} \rceil$ and 0 otherwise. The upper bound in (49) follows from the fact that the number of combinations of m nonempty boxes equals the number of ways to choose m boxes out of n' boxes times the number of ways to place l balls in the mh spaces of those m boxes with at least one ball per box. The former quantity is given by $\binom{n'}{m}$ while the latter is upper bounded by $\binom{mh}{l}$. The lower bound in (49) follows from the fact that one way to fill the m nonempty boxes is to fill each one of them with an equal (or approximately equal) number of balls.

Invoking the well-known bound on the binomial coefficients given in [1]

$$\frac{1}{\sqrt{2\pi n\epsilon(1-\epsilon)}} \exp \left[nH_e(\epsilon) - \frac{1}{12n\epsilon(1-\epsilon)} \right] < \binom{n}{n\epsilon} < \frac{1}{\sqrt{2\pi n\epsilon(1-\epsilon)}} \exp [nH_e(\epsilon)] \quad (50)$$

where $n\epsilon$ is an integer less than or equal to n and

$$H_e(\epsilon) = -\epsilon \log \epsilon - (1-\epsilon) \log(1-\epsilon) \quad (51)$$

is the entropy function. We have

$$P_{\text{box}}(l, m, n) \leq C_b(l, m, n) e^{n\hat{B}(l, m, n)} \quad (52)$$

where

$$\hat{B}(l, m, n) = \frac{1}{h} H_e \left(\frac{m}{n'} \right) + \frac{mh}{n} H_e \left(\frac{l}{mh} \right) - H_e \left(\frac{l}{n} \right) \quad (53)$$

$$C_b(l, m, n) = \sqrt{\frac{1 - \frac{l}{n}}{2\pi m \left(1 - \frac{m}{n'}\right) \left(1 - \frac{l}{mh}\right)}} \exp \left[\frac{1}{12l \left(1 - \frac{l}{n}\right)} \right] \quad (54)$$

Let $N(l)$ be the number of codewords at a distance l from \mathbf{x}_0 . Thus from (48), we have

$$\sum_{j=1}^{M-1} \Pr[D(\mathbf{x}_0, \mathbf{y}) \leq n\delta; D(\mathbf{x}_j, \mathbf{y}) \leq D(\mathbf{x}_0, \mathbf{y})] \leq \sum_{l=0}^n N(l) e^{-nr\delta} \sum_{m=\lceil l/h \rceil}^{\min(n', l)} [h(r)]^m [g(r)]^{n'-m} P_{box}(l, m, n) \quad (55)$$

Let $\overline{N(l)}$ be the average number of codewords of weight l in some ensemble of linear codes. The ensemble average probability of decoding error is then given by (35),(36),(39) and (55) as

$$\overline{P_e} \leq g(z)^{n'} e^{-nz\delta} + \sum_{l=0}^n \overline{N(l)} e^{-nr\delta} \sum_{m=\lceil l/h \rceil}^{\min(n', l)} [h(r)]^m [g(r)]^{n'-m} P_{box}(l, m, n) \quad (56)$$

for any $z \geq 0, r \leq 0$

In [1][25], $\overline{N(l)}$ of a (j, k) -regular LDPC ensemble has been shown to be bounded by

$$\overline{N(l)} \leq C(\lambda, n) e^{nB(\lambda)} \quad (57)$$

where $\lambda = l/n$ and

$$B(\lambda) = (1-j)H_e(\lambda) + \frac{j}{k} [\mu(\gamma) + (k-1)\log(2)] - j\gamma\lambda \quad (58)$$

$$C(\lambda, n) = [2\pi n \lambda (1-\lambda)]^{\frac{j-1}{2}} \exp \frac{j-1}{12n\lambda(1-\lambda)} \quad (59)$$

$$\mu(\gamma) = \log \left\{ 2^{-k} [(1+e^\gamma)^k + (1-e^\gamma)^k] \right\} \quad (60)$$

where γ , the parameter to optimize the bound, is selected such that $\lambda = \frac{\mu'(\gamma)}{k}$.

Letting

$$\tilde{C}_n \equiv \max_{\lambda} C(\lambda, n) \quad , \quad \hat{C}_n \equiv \max_{l, m} C_b(l, m, n) \quad (61)$$

rearranging (56), and bounding the summation by the number of terms in the sum times the value of the maximum term, we get

$$\overline{P_e} \leq \exp \left\{ n \left[\frac{\log g(z)}{h} - z\delta \right] \right\} + n'n\tilde{C}_n\hat{C}_n \max_{\lambda} \exp \left\{ n \left[B(\lambda) - r\delta + \max_m \left(\frac{m}{n} \log[h(r)] + \frac{n'-m}{n} \log[g(r)] + \hat{B}(\lambda n, m, n) \right) \right] \right\} \quad (62)$$

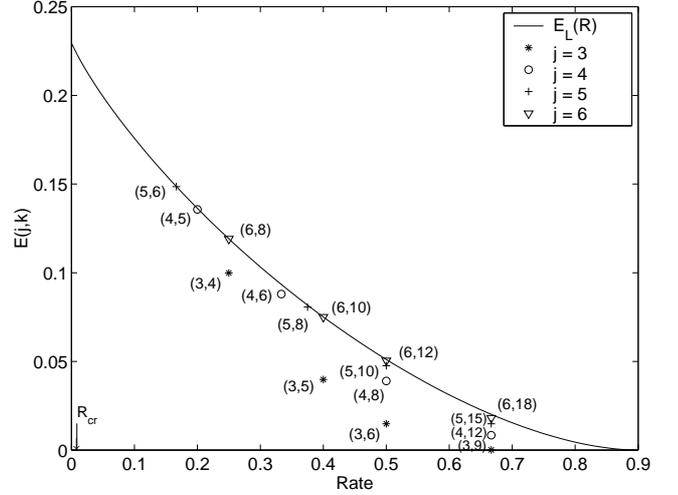


Fig. 7. Lower bound on the error exponent of (j, k) -regular LDPC over the two-state noiseless/useless BSC block interference channel with $h = 10$ when channel state is known and $\epsilon_b = 0.1$

For a given z, r and λ , we select the value δ that makes the two exponents in (62) equal. With this choice of δ , (62) becomes

$$\overline{P_e} \leq (1 + n'n\tilde{C}_n\hat{C}_n) \exp \left\{ -n \left[\min_{\lambda} E(z, r, \lambda) \right] \right\} \quad (63)$$

where

$$E(z, r, \lambda) = \frac{r}{(z-r)h} \log[g(z)] - \frac{z}{z-r} \left[B(\lambda) + \max_m \left\{ \frac{m}{n} \log[h(r)] + \frac{n'-m}{n} \log[g(r)] + \hat{B}(\lambda n, m, n) \right\} \right] \quad (64)$$

By (30), we have

$$E(j, k) \geq \max_{z \geq 0, r \leq 0} \min_{\lambda} E(z, r, \lambda) \quad (65)$$

Optimizing (65) with respect to r, z and λ yields a lower bound on the error exponent of a random ensemble of (j, k) -regular LDPC codes used over the two-state noiseless/useless BSC block interference channel. We evaluated this bound for some regular LDPC code ensembles with variable node degrees $j = 3, 4, 5$ and 6 . The LDPC code ensemble weight distribution exponent $B(\lambda)$ used in (64) is given by (58), however, we expurgated from the ensemble all codes with minimum fractional weight λ satisfying $B(\lambda) < 0$. This expurgation will have negligible effect since, asymptotically, almost all codes in the ensemble belong to this expurgated ensemble. The results along with the lower bounds on the channel error exponent $E_L(R)$ [22][23] are shown in Figs. 7 and 8 for the cases when the channel state is known and unknown, respectively. The critical rate [22], R_{cr} , of the channels has also been computed and is indicated on these figures. The bound $E_L(R)$ is known to be tight, i.e., $E_L(R) = E(R)$ for $R \geq R_{cr}$ [22].

As can be seen, the lower bound on the error exponent of the regular LDPC code ensemble approaches the random coding channel exponent bound, $E_L(R)$, as the node degree j increases for the case where $h = 10$ and $\epsilon_b = 0.1$. Similar

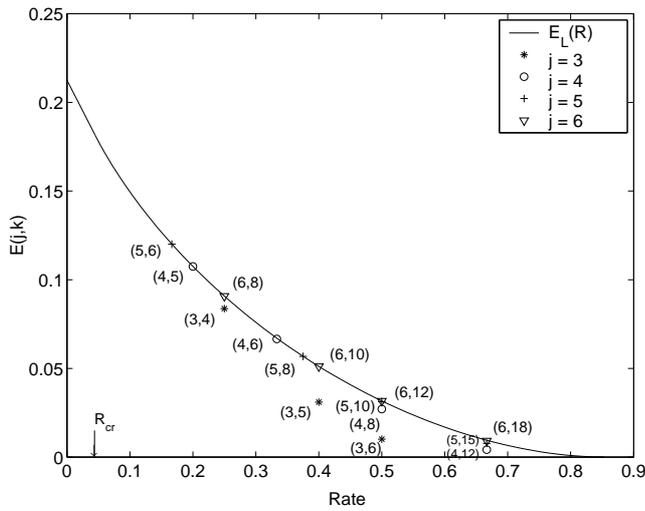


Fig. 8. Lower bound on the error exponent of (j, k) -regular LDPC over the two-state noiseless/useless BSC block interference channel with $h = 10$ when channel state is unknown and $\epsilon_b = 0.1$

results are obtained for the case when $h = 2, 10, 20$ and $\epsilon_b = 0.1, 0.3, 0.5, 0.7, 0.9$. Also illustrated in the figure is the fact that the difference is larger at higher rates. This increase, however, may depend more on the tightness of the bound at high rates rather than on the code itself.

V. CONCLUSION

In this paper we consider the two-state noiseless/useless BSC block interference channel. When the channel is in the noiseless state the BSC crossover probability is zero while it is 1/2 in the useless state. Each block of h consecutive channel input bits shares the same channel state, which is selected independently and at random from block to block according to a known prior distribution. Communication performance over this channel is analyzed when regular LDPC codes are used and the receiver implements iterative sum-product decoding either with or without joint channel state estimation. The threshold performance of the receiver is evaluated using density evolution and the results are compared with the channel capacity and bit-error-rate simulations obtained using LDPC codes of codeword length 4000. The threshold and simulation results indicate that substantial improvements in performance can be realized by using joint channel state estimation. Finally lower bounds are derived for the error exponent when LDPC codes are used. These lower bounds are shown to approach the random coding channel error exponent bound, in a number of cases, as the degree of the LDPC bit nodes increases.

ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their helpful comments and suggestions that materially improved the quality of our paper.

REFERENCES

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

[2] M. Luby, M. Mitzenmacher, and D. A. Spielman, "Analysis of low density codes and improved designs using irregular graphs," in *Proc. of the 30th Annual ACM Symposium on Theory of Computing*, 1998, pp. 249–258.

[3] T. J. Richardson and R. L. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 599–618, Feb. 2001.

[4] T. J. Richardson, A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 619–637, Feb. 2001.

[5] S. Y. Chung, G. D. Forney, T. J. Richardson, and R. Urbanke, "On the design of low-density parity-check codes within 0.0045 db of the Shannon limit," *IEEE Commun. Lett.*, vol. 5, pp. 58–60, Feb. 2001.

[6] X. Jin, A. W. Eckford, and T. E. Fuja, "Analysis of LDPC decoding for correlated and uncorrelated block fading channels," in *Proceedings of the 2004 International Symposium on Information Theory*, Chicago, IL, June 2004.

[7] N. Wiberg, *Codes and decoding on general graphs*, Ph.D. thesis, Dept. Elec. Eng., Linköping Univ., Linköping, Sweden, 1996.

[8] A. P. Worthen and W. E. Stark, "Low-density parity check codes for fading channels with memory," in *Proceedings of the 36th Allerton Conference on Communication, Control and Computing*, Monticello, IL, Sep. 23-25, 1998, pp. 117–125.

[9] A. P. Worthen and W. E. Stark, "On iterative receivers for non-coherent channels," in *Proceedings of International Symposium on Information Theory and Its Applications*, Honolulu, Hawaii, Nov. 5-8, 2000.

[10] A. P. Worthen and W. E. Stark, "Unified design of iterative receivers using factor graphs," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 843–849, Feb. 2001.

[11] A. P. Worthen, *Codes and Iterative Receivers for Wireless Communication Systems*, Ph.D. thesis, University of Michigan, Ann Arbor, MI, Sep. 2001.

[12] J. Garcia-Frías, "Decoding of low-density parity check codes over finite-state binary Markov channels," in *Proceedings of International Symposium on Information Theory*, Washington, DC, June 24-29, 2001, p. 72.

[13] E. A. Ratzler, "Low-density parity-check codes on Markov channels," in *Proceedings of 2nd IMA Conference on Mathematics in Communications*, Lancaster, UK, Dec. 16-18, 2002.

[14] A. W. Eckford, F. R. Kschischang, and S. Pasupathy, "Analysis of LDPC codes in channels with memory," in *Proceedings of 21st Biennial Symposium on Communications*, Kingston, Ontario, Canada, June 2-5, 2002.

[15] F. R. Kschischang and A. W. Eckford, "Low-density parity-check codes for the Gilbert-Elliott channel," in *Proceedings of 41st Allerton Conference on Communication, Control and Computing*, Monticello, IL, Oct. 1-3, 2003.

[16] A. W. Eckford, F. R. Kschischang, and S. Pasupathy, "Designing very good low-density parity-check codes for the Gilbert-Elliott channel," in *Proceedings of 8th Canadian Workshop on Information theory*, Waterloo, Canada, May 18-21, 2003.

[17] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[18] S. M. Aji and R. J. McEliece, "The generalized distributive law," *IEEE Trans. Inform. Theory*, vol. IT-46, no. 2, pp. 325–343, March 2000.

[19] R. J. McEliece and W. E. Stark, "Channels with block interference," *IEEE Trans. Inform. Theory*, vol. IT-30, no. 1, pp. 44–53, Jan. 1984.

[20] W. Vijacksungsithi, *Low-Density Parity-Check Codes and Iterative Decoding Algorithms for Input-Constrained Channels and Channels with Memory*, Ph.D. thesis, University of Michigan, Ann Arbor, MI, Jan. 2003.

[21] D. J. C. Mackay, "Encyclopedia of sparse graph codes," [Online]. Available: <http://www.inference.phy.cam.ac.uk/mackay/codes/data.html>.

[22] A. J. Viterbi, *Principles of Digital Communication and Coding*, McGraw-Hill, New York, 1979.

[23] R. G. Gallager, *Information Theory and Reliable Communication*, Wiley, New York, 1968.

[24] R. M. Fano, *Transmission of Information*, M.I.T. Press, 1961.

[25] S. Litsyn and V. Shevelev, "On ensembles of low-density parity-check codes: Asymptotic distance distribution," *IEEE Trans. Inform. Theory*, vol. 48, no. 4, pp. 887–908, April 2002.



PLACE
PHOTO
HERE

Wongkot Vijacksungsithi was born in Bangkok, Thailand in 1978. He received his B.S. degree in electrical engineering in 1997 from Chulalongkorn University, Bangkok, Thailand, and the M.S. and Ph.D. degrees in electrical engineering from the University of Michigan in 1999 and 2003, respectively. Since 2003, he has been with CAT Telecom, Bangkok, Thailand.



PLACE
PHOTO
HERE

Kim A. Winick (S'77-M'80-SM'98) was born in New York City, New York in 1954. He received his B.S. degree in electrical engineering in 1976 from the Pennsylvania State University and the M.S. and Ph.D. degrees in electrical engineering from the University of Michigan in 1977 and 1981, respectively. From 1981 to 1988, he was a member of the technical staff at the Massachusetts Institute of Technology Lincoln Laboratory. In 1988, he joined the Department of Electrical Engineering and Computer Science at the University of Michigan as an Assistant Professor, and in 1994 he was promoted to an Associate Professor with tenure. Professor Winick's principal research interests lie in the areas of quantum electronics, integrated optics, and communications. He is a member of the Optical Society of America and currently serves as a topical editor of Optics Letters.