Low-Density Parity-Check Codes Over Gaussian Channels With Erasures

Jeongseok Ha and Steven W. McLaughlin, *Senior Member, IEEE*

Abstract—We consider low-density parity-check code (LDPC) design for additive white Gaussian noise (AWGN) channels with erasures. This model, for example, represents a common situation in magnetic and optical recording where defects or thermal asperities in the system are detected and presented to the decoder as erasures. In this correspondence, we give thresholds of regular and irregular LDPCs and discuss practical code design over the mixed Gaussian/erasures channel. The analysis is an extension of the Gaussian approximation work of Chung *et al.* In the two limiting cases of no erasures and large signal-to-noise ratio (SNR), the analysis tends to the results of Chung *et al.* and Luby *et al.*, respectively, giving a general tool for a class of mixed channels. We derive a steady-state equation which gives a graphical interpretation of decoder convergence. This allows one to estimate the maximum erasure capability on the mixture channel, or conversely, to estimate the additional signal power required to compensate for the loss due to erasures. We see that a good (capacity-approaching) LDPC over an AWGN channel is also good over the mixed channel up to a moderate erasure probability. We also investigate practical issues such as the maximum number of iterations of message-passing decoders, the coded block length, and types of erasure patterns (random/block erasures). Finally, we design an optimized LDPC for the mixed channel, which shows better performance if the erasure probability is larger than a certain value (0.1 in our simulation) at the expense of performance degradation at unerased (AWGN channel) and lower erasure probability regions (less than 0.1 in our simulation).

Index Terms—Gaussian approximation, Gaussian noise, low-density parity-check codes (LDPCs), random erasures.

I. INTRODUCTION

Low-density parity-check codes (LDPCs) are due to Gallager [1], [2], and renewed interest began as a result of a great success of turbo codes [3] and subsequent rediscovery by MacKay [4], MacKay and Neal [5], and Sipser and Spielman [6]. Since the rediscoveries, much effort has been paid to designing good LDPCs over various channels such as binary erasure channels (BECs), binary symmetric channels (BSCs), and additive white Gaussian noise (AWGN) channels. Luby *et al.* [7] designed good LDPCs over BECs and introduced an irregular edge degree distribution. Richardson and Urbanke [8] investigated variations of message densities in message-passing decoders and devised an algorithm called *density evolution* for iteratively computing message densities and the signal-to-noise ratio (SNR) threshold of an LDPC. Although Chung [9] introduced a simplified density evolution called *discretized density evolution* (DDE), computing thresholds and designing good LDPCs over most channels, except BECs, require intensive computations. To address the computational difficulty, Chung [10] used the *Gaussian approximation* (GA) that models the message densities in message-passing decoders as Gaussian distributions for regular LDPCs and a mixture of Gaussian distributions for irregular

Manuscript received April 29, 2002; revised February 26, 2003. This work was supported in part by the National Science Foundation under grant CCR-0121565 and the State of Georgia Yamacraw Initiative. The material in this correspondence was presented in part at the IEEE International Symposium on Information Theory, Lausanne, Switzerland, June/July 2002.

The authors are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0250 USA (e-mail: jsha@ece.gatech.edu; swm@ece.gatech.edu).

Communicated by K. Abdel-Ghaffar, Associate Editor for Coding Theory.
Digital Object Identifier 10.1109/TIT.2003.813480

Fig. 1. AWGN with random erasures. (a) x is a message bit in $\{1, 0\}$, c is a coded symbol in $\{-1, +1\}$, n is white Gaussian noise ($n \sim \mathcal{N}(0, \sigma_n^2)$), e is in $\{1, 0\}$, $P(e = 0) = e^{(0)}$, $p(e = 1) = (1 - e^{(0)})$, and $\hat{x}$ is a decoded bit in $\{1, 0\}$. (b) r is a received signal and m is the mean of the received signal.

LDPCs. A prerequisite of the approximation is the symmetry condition [11]. At the expense of accuracy, the approximation gives a faster threshold computation, more insight into the convergence of message-passing decoders, and an analytic expression of the variations of the message densities during iterations. From the design point of view, the approximation makes the design problem a linear optimization that can be easily solved with linear programming.

In this correspondence, we extend Chung's work to include a channel model that is a mixture of Gaussian noise with random erasures, as depicted in Fig. 1(a) and (b). In this model, a binary symbol is corrupted with either added white Gaussian noise or is erased with a probability of $e^{(0)}$. This model, for example, represents a common channel in magnetic recordings where thermal asperities in the system are detected and represented at the decoder as erasures. A typical code must correct a burst of up to 48 bytes over a 512-byte sector. In an optical recording, defects or scratches cause long numbers of bytes (several hundred or thousand) to be erased. The probability density function of the log-likelihood ratio (LLR) of the channel output is

$$f(v) = \frac{1 - e^{(0)}}{\sqrt{4\pi m_{u0}}} e^{-\frac{(v \pm m_{u0})^2}{4m_{u0}}} + e^{(0)} \Delta_0(v) = g^{(0)}(v) + e^{(0)} \Delta_0(v) \quad (1)$$

where $v = \log_e [p(r|c=1)/p(r|c=-1)]$ is the LLR of r , r is the channel output shown in Fig. 1(b), $m_{u0} = E[v|v \neq 0]$, $\text{Var}(v|v \neq 0) = 2m_{u0}$, $e^{(0)}$ is the random erasure probability, and $\Delta_x(v) = \delta(v - x)$ is a shifted delta function. The probability density function satisfies the symmetry condition [11], $f(v) = f(-v)e^v$. We analyze the message density evolution over the channel model shown in Fig. 1(a) and (b) with some modifications to the Gaussian approximation.

The remainder of the correspondence is organized as follows. In Section II, for the reader's convenience, we briefly summarize Chung's work in [10]. In Section III, we consider regular LDPCs and define the basic terminology used in the subsequent sections. In Section IV, we extend the results of Section III to irregular LDPCs. In Section V, we simplify the results of Section IV using the fact that the erasure

probability during iterations does not depend on an SNR. The simplified equation, called a *steady-state equation*, gives us a graphical interpretation of decoder convergence that was originally introduced in [10]. In Section VI, we verify our analysis and the channel model by comparing theoretical E_b/N_0 thresholds of regular and irregular LDPCs with simulation results of actual LDPCs. We also investigate some practical issues such as maximum number of iterations of message-passing decoders, coded block length, and types of erasure patterns (random/block erasures). In Section VII, we design an LDPC for the mixed channel and compare the designed LDPC with an LDPC optimized for an AWGN channel in [10]. We conclude our work in Section VIII.

II. GAUSSIAN APPROXIMATION (GA)

Our work is based on the results of *Gaussian approximation* (GA) in [10]. In this section, we briefly summarize GA to help readers understand the following sections. This summary is the minimum needed to understand our work and describes GA for irregular LDPCs only, which includes regular LDPCs as a special case.

The LLR message (*message* for short) of a check node at the k th iteration ($u^{(k)}$ and $k \in \mathbb{N}$) during message-passing decoding is approximated as a Gaussian probability density function that is completely specified by its mean ($m_u^{(k)}$) and variance ($\text{Var}(u^{(k)})$). The symmetry condition, $f(x) = e^x f(-x)$, which is preserved during iterations, imposes a relation between the mean and variance, $\text{Var}(u^{(k)}) = 2m_u^{(k)}$. Thus, by tracing the changes of the mean (called *updated mean*) during iterations, we can watch the evolution of the density of the check node message, $f^{(k)}(u) = \mathcal{N}(m_u^{(k)}, 2m_u^{(k)})$, where $\mathcal{N}(\mu, \sigma^2)$ is the normal probability density function with a mean and variance of μ and σ^2 , respectively.

The probability density function (*density* for short) of a variable-node message at the k th iteration ($v^{(k)}$) is approximated as a Gaussian mixture for irregular LDPCs and a Gaussian for regular LDPCs

$$f^{(k)}(v) = \sum_{i=2}^{d_l} \lambda_i \mathcal{N}(m_{v,i}^{(k)}, 2m_{v,i}^{(k)})$$

where λ_i is the fraction of edges belonging to degree i variable nodes, $m_{v,i}^{(k)} = m_{u_0} + (i-1)m_u^{(k-1)}$, m_{u_0} is the mean of an LLR message out of channel, and $m_u^{(0)} = 0$.

Between a check node of degree d_c and the variable nodes, there is the well-known tanh rule [12]:

$$\tanh \frac{u^{(k)}}{2} = \prod_{j=1}^{d_c-1} \tanh \frac{v_j^{(k)}}{2} \quad (2)$$

where $v_j^{(k)}$, $j = 1, \dots, d_c-1$ are the incoming messages from d_c-1 neighbors and $u^{(k)}$ is the message sent to the remaining node. The expectation of each side of (2) can be simplified as

$$E \left[\tanh \frac{u^{(k)}}{2} \right] = E \left[\tanh \frac{v^{(k)}}{2} \right]^{d_c-1}. \quad (3)$$

Because the variable node messages are independent and identically distributed (i.i.d.), we can omit the index j in (2) and simplify the product. The expectation $E \left[\tanh \frac{u}{2} \right]$ can be expressed as

$$E \left[\tanh \frac{u}{2} \right] = \frac{1}{\sqrt{4\pi m_u}} \int_{\mathbb{R}} \tanh \frac{u}{2} e^{-\frac{(u-m_u)^2}{4m_u}} du \quad (4)$$

where u is a Gaussian, $m_u = E[u]$, and $\text{Var}(u) = 2m_u$. Thus, both sides of (3) become integral forms that are simplified by the following definition.

Definition 1 [10, Definition 1]:

$$\phi(x) = \begin{cases} 1 - \frac{1}{\sqrt{4\pi x}} \int_{\mathbb{R}} \tanh \frac{u}{2} e^{-\frac{(u-x)^2}{4x}} du, & \text{if } x > 0 \\ 1, & \text{if } x = 0. \end{cases}$$

$\phi(x)$ is continuous and monotonically decreasing on $[0, \infty)$, with $\phi(0) = 1$ and $\phi(\infty) = 0$.

By applying (4) and $\phi(x)$ to both sides of (3), we can calculate the updated mean of a degree j check node as

$$m_{u,j}^{(k)} = \phi^{-1} \left(1 - \left[1 - \sum_{i=2}^{d_l} \lambda_i \phi(m_{v,i}^{(k)}) \right]^{(j-1)} \right).$$

The check node updated mean becomes a recursive equation

$$\begin{aligned} m_u^{(k)} &= \sum_{j=2}^{d_r} \rho_j m_{u,j}^{(k)} \\ &= \sum_{j=2}^{d_r} \rho_j \phi^{-1} \left(1 - \left[1 - \sum_{i=2}^{d_l} \lambda_i \phi(m_{u_0} + (i-1)m_u^{(k-1)}) \right]^{(j-1)} \right). \end{aligned} \quad (5)$$

For error-free decoding, the recursive equation converges to ∞ as $k \rightarrow \infty$, which implies

$$\sum_{i=2}^{d_l} \lambda_i \phi(m_{u_0} + (i-1)m_u^{(k-1)}) \rightarrow 0, \quad \text{as } k \rightarrow \infty$$

because $\phi^{-1}(x) \rightarrow \infty$ as $x \rightarrow 0$.

For $0 \leq s < \infty$ and $0 < r \leq 1$, $h_i(s, r)$ and $h(s, r)$ are defined in [10] as

$$h_i(s, r) = \phi \left(s + (i-1) \sum_{j=2}^{d_r} \rho_j \phi^{-1} \left(1 - (1-r)^{j-1} \right) \right),$$

$$h(s, r) = \sum_{i=2}^{d_l} \lambda_i h_i(s, r)$$

and (5) is equivalently expressed as

$$r_k = h(s, r_{k-1}) \quad (6)$$

where $s = m_{u_0}$ and $r_0 = \phi(m_{u_0})$.

In the recursion, the degree distribution pair $(\lambda(x), \rho(x))$ and s determine the convergence to a fixed point (not necessarily zero). As shown in (5) and (6), $m_u^{(k)} \rightarrow \infty$ if and only if $r_k \rightarrow 0$. Thus, the convergence condition $r_k(s) \rightarrow 0$ is satisfied if and only if

$$h(s, r) < r, \quad \forall r \in (0, \phi(s)). \quad (7)$$

Definition 2 [10, Definition 2]: The threshold s^* is the infimum of all s in $\mathbb{R}^+$ such that $r_k(s)(m_u^{(k)}(s))$ converges to $0(\infty)$ as $k \rightarrow \infty$.

III. ANALYSIS OF REGULAR LDPCs

The message flows between a check and a variable node are depicted in Fig. 2(a) and (b), where the square and circle symbols represent check nodes and variable nodes, respectively. The function $h^{(k)}(u)$ ($g^{(k)}(v)$) is the continuous part of the probability density of a check (variable) node message. The term $\varepsilon^{(k)}$ ($e^{(k)}$) is the probability that a check (variable) node messages is equal to zero. d_c (d_v) is the number

Fig. 2. Message flows between a check and a variable node. (a) LLR message of a check node at the k th iteration. (b) LLR message of a variable node at the $(k + 1)$ th iteration.

of neighbors of a check (variable) node. $u_j^{(k)}$ ($v_j^{(k)}$) is the message of a check (variable) node emitting through the j th edge (hereafter, without loss of generality, we will drop the index j). In the sum-product decoding algorithm, if at least one of the variable nodes connected to a check node is erased, the check node has zero LLR as its message. Thus, the probability of $\varepsilon^{(k)}$ is expressed as

$$\varepsilon^{(k)} = 1 - \prod_{i=1}^{d_c-1} P(v_i^{(k)} \neq 0) = 1 - (1 - e^{(k)})^{d_c-1}. \quad (8)$$

If none of $d_c - 1$ variable messages is erased, the probabilistic characteristics of the check message can be approximated by a Gaussian density denoted by $h^{(k)}(u)$ in Fig. 2. The Gaussian density of a message is determined by one variable, the updated mean value $m_u^{(k)}$ at the k th iteration. The updated mean is determined from the following relation:

$$\begin{aligned} 1 - \phi(m_u^{(k)}) &= E \left[\tanh \left[u^{(k)} / 2 \right] \middle| u^{(k)} \neq 0 \right] \\ &= E \left[\tanh \left[v^{(k)} / 2 \right] \middle| v^{(k)} \neq 0 \right]^{d_c-1} \\ &= \left[\frac{1}{1 - e^{(k)}} \left\langle \tanh \left[v^{(k)} / 2 \right], g^{(k)}(v) \right\rangle \right]^{d_c-1} \end{aligned} \quad (9)$$

where $\langle f(v), g(v) \rangle = \int_{\mathbb{R}} f(v)g(v) dv$, and $g^{(k)}(v)$ is the probability density of a variable node message given $v^{(k)} \neq 0$. From (9), we can express the updated mean value as

$$m_u^{(k)} = \phi^{-1} \left(1 - \frac{1}{(1 - e^{(k)})^{d_c-1}} \left\langle \tanh \left[v^{(k)} / 2 \right], g^{(k)}(v) \right\rangle^{d_c-1} \right). \quad (10)$$

To make (10) a recursive equation of $m_u^{(k)}$, we need an expression for the probability density of a variable node message. In the log-probability domain, the variable node message is determined by the linear

sum of $d_v - 1$ incident check node messages as shown in Fig. 2. Thus, the probability that the message of a variable node is zero can be computed as

$$P(v^{(k)} = 0) = e^{(0)} \left(\varepsilon^{(k-1)} \right)^{d_v-1} \quad (11)$$

which is the probability that all incident check messages are zeros and the variable node is erased. The probability density of a variable message can be factored into three terms as shown in (12). The first term gives the probability in (11), the second term is made up of the incident check node messages given the variable node is erased, and the last term is the combination of the check node messages with the unerased received message. The last two terms comprise the continuous part $g^{(k)}(v)$ used in (10).

$$\begin{aligned} f^{(k)}(v) &= e^{(k)} \Delta_0(v) + g^{(k)}(v) \\ &= \underbrace{e^{(k)} \Delta_0(v)}_{\text{Unrecovered erased symbols}} \\ &\quad + \underbrace{e^{(0)} \left\{ \sum_{i=1}^{d_v-1} (d_v-1) \chi_i^{(k)} \mathcal{N} \left(im_u^{(k-1)}, 2im_u^{(k-1)} \right) \right\}}_{\text{Erased symbols}} \\ &\quad + \underbrace{\left(1 - e^{(0)} \right) \left\{ \sum_{i=0}^{d_v-1} (d_v-1) \chi_i^{(k)} \mathcal{N} \left(im_u^{(k-1)} + m_{u0}, 2im_u^{(k-1)} + 2m_{u0} \right) \right\}}_{\text{Unrecovered symbols}} \end{aligned} \quad (12)$$

where

$$\begin{aligned} \mathcal{N}(m, 2m) &= \frac{1}{\sqrt{4\pi m}} e^{-\frac{(v-m)^2}{4m}}, \\ n \chi_m^{(k)} &= {}_n C_m \left(\varepsilon^{(k-1)} \right)^{n-m} \left(1 - \varepsilon^{(k-1)} \right)^m \end{aligned}$$

and ${}_n C_m$ is the binomial coefficient. The continuous part of a variable-node message density derived in (12) expands (10) to

$$\begin{aligned} m_u^{(k)} &= \phi^{-1} \left(1 - \frac{1}{(1 - e^{(k)})^{d_c-1}} \right. \\ &\quad \times \left\{ 1 - e^{(0)} \sum_{i=0}^{d_v-1} (d_v-1) \chi_i^{(k)} \phi \left(im_u^{(k-1)} \right) \right. \\ &\quad \left. \left. - \left(1 - e^{(0)} \right) \sum_{i=0}^{d_v-1} (d_v-1) \chi_i^{(k)} \right. \right. \\ &\quad \left. \left. \times \phi \left(im_u^{(k-1)} + m_{u0} \right) \right\}^{d_c-1} \right) \end{aligned} \quad (13)$$

where $\langle \tanh(v/2), \mathcal{N}(x, 2x) \rangle = 1 - \phi(x)$. Now, the updated mean becomes a recursive equation for a given random erasure rate $e^{(0)}$.

IV. ANALYSIS OF IRREGULAR LDPCs

To describe irregular LDPCs, let

$$\lambda(x) = \sum_{i=2}^{d_l} \lambda_i x^{i-1} \quad \text{and} \quad \rho(x) = \sum_{i=2}^{d_r} \rho_i x^{i-1}$$

be a pair of generating functions (used in [12]) of the degree distributions for the variable and check edges, respectively. To include the degree distribution in calculating the probability of $\varepsilon^{(k)}$ and the updated

mean $m_u^{(k)}$ that describe the check node message density, we need to modify (8) and (9) to form (14) and (15), respectively

$$\begin{aligned}\varepsilon_{d_c=s}^{(k)} &= P(u^{(k)} = 0 \mid d_c = s) \\ &= 1 - \prod_{i=1}^{s-1} P(v_{i=1}^{(k)} \neq 0) = 1 - (1 - e^{(k)})^{s-1} \\ \varepsilon^{(k)} &= \sum_{s=2}^{d_r} P(d_c = s) \varepsilon_{d_c=s}^{(k)} \\ &= \sum_{s=2}^{d_r} \rho_s \left(1 - (1 - e^{(k)})^{s-1}\right) = 1 - \rho \left(1 - e^{(k)}\right) \quad (14) \\ 1 - \phi\left(m_{u \mid d_c=s}^{(k)}\right) &= E\left[\tanh\left[u^{(k)}/2\right] \mid u^{(k)} \neq 0, d_c = s\right] \\ &= E\left[\tanh\left[v^{(k)}/2\right] \mid v^{(k)} \neq 0\right]^{s-1} \quad (15)\end{aligned}$$

where $P(d_c = s)$ is the probability that an edge belongs to a degree s check node. Thus, by making use of (10), we compute the updated mean as

$$\begin{aligned}m_u^{(k)} &= \sum_{s=2}^{d_r} P(d_c = s) m_{u \mid d_c=s}^{(k)} \\ &= \sum_{s=2}^{d_r} \rho_s \phi^{-1} \left(1 - \frac{1}{(1 - e^{(k)})^{s-1}}\right) \\ &\quad \times \left\{ \left\langle \tanh\left[v^{(k)}/2\right], g^{(k)}(v) \right\rangle \right\}^{s-1} \quad (16)\end{aligned}$$

where

$$m_{u \mid d_c=s}^{(k)} = \phi^{-1} \left(1 - E\left[\tanh\left[v^{(k)}/2\right] \mid v^{(k)} \neq 0\right]^{s-1}\right)$$

and

$$\begin{aligned}E\left[\tanh\left[v^{(k)}/2\right] \mid v^{(k)} \neq 0\right]^{s-1} \\ = \frac{1}{(1 - e^{(k)})^{s-1}} \left\{ \left\langle \tanh\left[v^{(k)}/2\right], g^{(k)}(v) \right\rangle \right\}^{s-1}.\end{aligned}$$

The probability density of a variable node message can be calculated by modifying (11) and (12). The expression of $e^{(k)}$ in (11) is changed to

$$\begin{aligned}e^{(k)} &= P(v^{(k)} = 0) \\ &= \sum_{j=2}^{d_l} P(d_v = j) e^{(0)} \left(\varepsilon^{(k-1)}\right)^{j-1} \\ &= e^{(0)} \sum_{j=2}^{d_l} \lambda_j \left(\varepsilon^{(k-1)}\right)^{j-1} \\ &= e^{(0)} \lambda \left(\varepsilon^{(k-1)}\right) \\ &= e^{(0)} \lambda \left(1 - \rho \left(1 - e^{(k-1)}\right)\right) \quad (17)\end{aligned}$$

where $P(d_v = j)$ is the probability that an edge belongs to a degree- j variable node. The recursive erasure probability of a variable node $e^{(k)}$ is exactly the same as the erasure probability derived by Luby *et al.* in [7]. There is a threshold $e_{\max}^{(0)}(\lambda, \rho)$ that is the largest real value in $(0, 1]$ satisfying $e_{\max}^{(0)}(\lambda, \rho) \lambda (1 - \rho(1 - e)) < e$ for any $e \in (0, e_{\max}^{(0)}(\lambda, \rho)]$. The threshold does not depend on the SNR but on the code structure specified by a degree distribution pair $(\lambda(x), \rho(x))$. Thus, as long as the erasure probability ($e^{(0)}$) is less than the threshold ($e_{\max}^{(0)}(\lambda, \rho)$), we can make the probability of an erased symbol as small as possible with the sum-product algorithm. If we increase the

SNR of the transmitted signal to infinity, which becomes a pure erasure channel, the probability density of a variable node message becomes

$$f^{(k)}(v) = e^{(0)} \lambda \left(1 - \rho \left(1 - e^{(k-1)}\right)\right) \Delta_0(v).$$

The continuous part of the probability density, $g^{(k)}(v)$, can be calculated with proper conditioning on λ_j , and the probability density of a variable node message can be factorized into three parts as we did in (12)

$$\begin{aligned}f^{(k)}(v) &= e^{(k)} \Delta_0(v) + g^{(k)}(v) \\ &= e^{(k)} \Delta_0(v) \\ &\quad + e^{(0)} \left\{ \sum_{j=2}^{d_l} \lambda_j \sum_{i=1}^{j-1} (j-1) \chi_i^{(k)} \mathcal{N}\left(im_u^{(k-1)}, 2im_u^{(k-1)}\right) \right\} \\ &\quad + \left(1 - e^{(0)}\right) \left\{ \sum_{j=2}^{d_l} \lambda_j \times \sum_{i=0}^{j-1} (j-1) \chi_i^{(k)} \right. \\ &\quad \left. \times \mathcal{N}\left(im_u^{(k-1)} + m_{u_0}, 2im_u^{(k-1)} + 2m_{u_0}\right) \right\}. \quad (18)\end{aligned}$$

Thus, the continuous part, $g^{(k)}(v)$ expands (16) to

$$\begin{aligned}m_u^{(k)} &= \sum_{s=2}^{d_r} \rho_s \phi^{-1} \left(1 - \frac{1}{(1 - e^{(k)})^{s-1}}\right) \\ &\quad \left[1 - \sum_{j=2}^{d_l} \lambda_j \left\{ e^{(0)} \sum_{i=0}^{j-1} (j-1) \chi_i^{(k)} \phi\left(im_u^{(k-1)}\right) \right. \right. \\ &\quad \left. \left. + \left(1 - e^{(0)}\right) \sum_{i=0}^{j-1} (j-1) \chi_i^{(k)} \phi\left(im_u^{(k-1)} + m_{u_0}\right) \right\} \right]^{s-1} \quad (19)\end{aligned}$$

which becomes a recursive equation of the updated mean $m_u^{(k)}$. Because we model the message densities as Gaussians, the bit-error probability can be computed as a weighted sum of Q functions

$$\begin{aligned}P_e^{(k)} &= e_0 \sum_{j=2}^{d_l} \lambda_j' \sum_{i=0}^j j \chi_i^{(k)} Q\left(\sqrt{\frac{im_u^{(k)}}{2}}\right) \\ &\quad + (1 - e_0) \sum_{j=2}^{d_l} \lambda_j' \sum_{i=0}^j j \chi_i^{(k)} Q\left(\sqrt{\frac{im_u^{(k)} + m_{u_0}}{2}}\right) \\ &= \underbrace{e_0 \frac{\sum_{j=2}^{d_l} \left(\lambda_j' \left(\varepsilon^{(k)}\right)^j\right)}{2}}_{\text{Unrecovered erasures}} \\ &\quad + \underbrace{e_0 \sum_{j=2}^{d_l} \lambda_j' \sum_{i=1}^j j \chi_i^{(k)} Q\left(\sqrt{\frac{im_u^{(k)}}{2}}\right)}_{\text{Erased symbols}} \\ &\quad + \underbrace{(1 - e_0) \sum_{j=2}^{d_l} \lambda_j' \sum_{i=0}^j j \chi_i^{(k)} Q\left(\sqrt{\frac{im_u^{(k)} + m_{u_0}}{2}}\right)}_{\text{Unrecovered symbols}} \quad (20)\end{aligned}$$

where

$$\lambda_j' = \frac{\lambda_j/j}{\sum_{i=2}^{d_l} \lambda_i/i}$$

and

$$e_0 \sum_{j=2}^{d_l} \lambda_j' \left(j \chi_0^{(k)} Q(0) \right) = e_0 \frac{\sum_{j=2}^{d_l} (\lambda_j'(\varepsilon^{(k)})^j)}{2}.$$

In the error probability, the first term is half of the unrecovered erasure probability at the k th iteration, the second term is the error probability due to the erased symbols, and the last term comes from the unerased symbols.

V. STEADY-STATE RECURSIVE EQUATION

In this section, we provide a simplified *steady-state equation* which gives us a graphical interpretation of the convergence of LDPCCs. As long as $e^{(0)}$ is less than the threshold $e_{\max}^{(0)}(\lambda, \rho)$, $e^{(k)}$ gradually reduces to zero and the recursive equation (19) becomes

$$m_u^{(k)} = \sum_{s=2}^{d_r} \rho_s \phi^{-1} \left(1 - \left[1 - \sum_{j=2}^{d_l} \lambda_j \left\{ e^{(0)} \phi \left((j-1) m_u^{(k-1)} \right) + \left(1 - e^{(0)} \right) \phi \left((j-1) m_u^{(k-1)} + m_{u0} \right) \right\} \right]^{s-1} \right) \quad (21)$$

which is a steady-state equation. For the steady-state recursive equation to grow up to infinity, the following inequality must be satisfied

$$\begin{aligned} r &> \sum_{j=2}^{d_l} \lambda_j \left\{ e^{(0)} h_j(0, r) + \left(1 - e^{(0)} \right) h_j(m_{u0}, r) \right\} \\ &= e^{(0)} h(0, r) + \left(1 - e^{(0)} \right) h(m_{u0}, r) \\ &= H(m_{u0}, e^{(0)}, r), \quad \forall r \in (0, \phi(m_{u0})) \end{aligned} \quad (22)$$

where

$$h(s, r) = \sum_{j=2}^{d_l} \lambda_j h_j(s, r) = E_{\lambda_j} [h_j(s, r)]$$

and

$$h_j(s, r) = \phi \left(s + (j-1) \sum_{i=2}^{d_r} \rho_i \phi^{-1} \left(1 - (1-r)^{i-1} \right) \right)$$

as defined in Section II. The inequality is the same as (7) if $e^{(0)} = 0$, which corresponds to an AWGN channel. From (22), we can notice that the variable node degree distribution λ_j must be optimized to minimize m_{u0} , while satisfying the inequality for a given erasure probability $e^{(0)} < e_{\max}^{(0)}(\lambda, \rho)$, which becomes a simple linear optimization. The details of the design process will be discussed in Section VII.

We explain the steady-state equation with a graphical interpretation by evaluating a 1/2 rate irregular code in [9] whose degree distributions are

$$\begin{aligned} \lambda(x) &= 0.23403x + 0.21242x^2 + 0.14690x^5 + 0.10284x^6 \\ &\quad + 0.30381x^{19} \end{aligned}$$

and

$$\rho(x) = 0.71875x^7 + 0.28125x^8. \quad (23)$$

In Fig. 3, the dotted and the long-dashed gray lines represent $h_i(s, r) - r$ and $h_i(0, r) - r$ for $i = 2, \dots, 20$ from the top to the bottom, respectively. By averaging $h_i(s, r) - r$ and $h_i(0, r) - r$ with λ_j , we can compute $h(s, r) - r$ and $h(0, r) - r$ denoted by the black dotted and the long-dashed lines, respectively. Finally, $H(s, e^{(0)}, r) - r$ is calculated by averaging $h(s, r) - r$ and $h(0, r) - r$ by $(1 - e^{(0)})$ and $e^{(0)}$, respectively. To satisfy the inequality in (22), $H(s, e^{(0)}, r) - r$ must be less than zero, $\forall r \in (0, \phi(s)]$. Therefore, over the mixed channel, the threshold $m_{u0}^M(e^{(0)})$ is the minimum of m_{u0} such that for a given erasure probability $e^{(0)} < e_{\max}^{(0)}(\lambda, \rho)$, if $s \geq m_{u0}$, then $H(s, e^{(0)}, r) - r < 0, \forall r \in (0, \phi(s)]$. Over an AWGN channel,

Fig. 3. $\{h_i(s, r) - r\}$ for $i = 2, \dots, 20$ (top to bottom), and $\{h(s, r) - r\}$ for $s = 0$ and 6.320891 .

Fig. 4. A magnified view of $H(s, e^{(0)}, r) - r$ when $s = 6.320891$ and $e^{(0)} = 0.38$.

the threshold m_{u0}^G is the minimum of m_{u0} such that if $s \geq m_{u0}$ then $H(s, 0, r) - r = h(s, r) - r < 0, \forall r \in (0, \phi(s)]$, which is the same as Definition 2 in Section II. Fig. 4 shows a magnification of $H(s, e^{(0)}, r) - r$ when $e^{(0)} = 0.38$ and $s = 6.320891$. From the curve, we can see that $H(s, e^{(0)}, r) - r$ barely avoids touching zero. If it does reach zero, the point of intersection becomes a fixed point of the recursive equation.

VI. PERFORMANCE PREDICTION

In this section, we verify our analysis and the channel model by comparing theoretical E_b/N_0 thresholds of regular and irregular LDPCCs with simulation results of actual LDPCCs. We also investigate some practical issues concerning small maximum number of iterations of the message-passing decoders, types of erasure patterns, and short block lengths.

In magnetic storage, a code block length of 4096 and coding rates of 4/5, 8/9, and 16/17 are common. In Fig. 5, we compare simulation results of corresponding regular LDPCCs for a bit-error rate (BER) of 10^{-4} with the thresholds predicted by GA. In the simulation, we limit the maximum number of iterations of the message-passing decoder to 25. The computed theoretical thresholds fairly predict asymptotic performances of the LDPCCs, which confirms our analysis of regular LDPCCs over the mixed channel.

Fig. 5. Comparison of E_b/N_0 variations of regular LDPCs with erasures at the coding rates of 4/5, 8/9, and 16/17 which are made of three ($\lambda_3 = 1$) nonzero terms in each column and 15 ($\rho_{15} = 1$), 27 ($\rho_{27} = 1$), and 51 ($\rho_{51} = 1$) nonzero terms in each row of the parity-check matrices, respectively; dots represent simulation results for a code length of 4096 and a BER of 10^{-4} , and lines are corresponding thresholds predicted with GA.

Fig. 6. BERs of an irregular LDPC having a code block length of 131702 bits, a coding rate of 1/2, maximum iterations of 200, and edge degree distributions $\lambda(x) = 0.23403x + 0.21242x^2 + 0.14690x^5 + 0.10284x^6 + 0.30381x^{19}$ and $\rho(x) = 0.71875x^7 + 0.28125x^8$.

Although regular LDPCs may be a more practical choice, irregular LDPCs give us more freedom in searching for capacity-approaching LDPCs. In Fig. 6 we evaluate BER performances of an irregular LDPC. The code is of length 131072 and is implemented with the degree distribution pair in (23). The analysis with GA fairly predicts the thresholds of the LDPC over the mixed channel. In Fig. 7 we depict thresholds predicted by GA and DDE [9], which are compared with the required E_b/N_0 's for a BER of 10^{-4} shown in Fig. 6. To see the effect of the maximum number of iterations and erasure patterns, we did the same simulation with two different values for the maximum number of iterations (25 and 200) and types of erasure patterns (random and block erasures). The thresholds predicted by GA and DDE have a 0.2-dB discrepancy, which was observed for the AWGN channel in [10]. In Fig. 8, we depict a magnified view of the gaps between the thresholds and the capacity of the mixed channel $C_{BPSK}(1 - e^{(0)})$. The threshold from DDE is less than 0.2 dB from the capacity as long as the erasure probability is less than 0.2. Thus,

Fig. 7. E_b/N_0 variations regarding erasure probabilities; filled and unfilled circles and triangles represent the results of an irregular LDPC having code block length of 131702 bits, a coding rate of 1/2, and maximum iterations of 25 and 200, respectively. The edge degree distributions are $\lambda(x) = 0.23403x + 0.21242x^2 + 0.14690x^5 + 0.10284x^6 + 0.30381x^{19}$ and $\rho(x) = 0.71875x^7 + 0.28125x^8$.

Fig. 8. E_b/N_0 gaps from the capacity; filled and unfilled circles and triangles represent the results of an irregular LDPC having code block length of 131702 bits, a coding rate of 1/2, and maximum iterations of 25 and 200, respectively. Edge degree distributions are $\lambda(x) = 0.23403x + 0.21242x^2 + 0.14690x^5 + 0.10284x^6 + 0.30381x^{19}$ and $\rho(x) = 0.71875x^7 + 0.28125x^8$.

good (capacity-approaching) LDPCs over AWGN channels are still good over the mixed channels up to a moderate erasure probability. However, at a high erasure probability there is a more distinct gap from the capacity. We can design an LDPC optimized for a high erasure probability or for a small average gap from the capacity over several erasure probabilities. Although the optimized LDPC has better thresholds at high erasure probabilities than those of an LDPC for an AWGN channel, the LDPC will suffer some loss in performance at low erasure probabilities.

Figs. 7 and 8 also show the effects of the maximum iterations and the types of erasure patterns (random and block erasures). In both erasure cases, we erase a fixed number of symbols. In the random erasure case, the positions of erased symbols are chosen randomly. In the block erasure case, we randomly choose the beginning of an erasure block from which we erase consecutive symbols. The block and random erasure cases (unfilled circles and triangles) show

Fig. 9. E_b/N_0 variations of LDPCCs for a BER of 10^{-4} with the erasure probabilities between 0 and 0.4 by a 0.1 step, a maximum number of iterations of 200, and the random and block erasure patterns.

Fig. 10. E_b/N_0 variations of LDPCCs for a BER of 10^{-4} with different maximum iterations (25, 50, 100, 150, and 200), erasure probability of 0.1, and random and block erasure patterns.

almost the same performance with a large number of iterations (200 iterations). We see performance degradation (filled circles and triangles) in both cases due to a smaller number of iterations (25 iterations). However, the degradation is more severe in the block erasure case. In Fig. 8, the additional E_b/N_0 's (the additional gaps from the capacity) in the random erasure case do not depend on the erasure probabilities. That is, at each erasure probability, the additional E_b/N_0 is almost the same. In the block erasure case, more additional E_b/N_0 is required for a higher erasure probability. Because block erasures can be efficiently converted to random erasures with a proper interleaver, the results in Sections III–V are still useful in the block erasure case.

We evaluate E_b/N_0 performances of the LDPCC used in Figs. 7 and 8 with several different block lengths (131072, 16384, and 4096), erasure patterns (random and block erasures), and maximum iterations. In Fig. 9, we depict the required E_b/N_0 's for a BER of 10^{-4} with a fixed maximum number of iterations, 200. At each erasure probability, the block and random erasure patterns have the same performances. Fig. 10 shows the variations of required E_b/N_0 's with different maximum iterations at an erasure probability of 0.1. The results show that

the performance variation decreases as the number of maximum iterations increases. Block and random erasure patterns have the same required E_b/N_0 as the maximum number of iterations becomes large enough (200 in our simulations). Figs. 9 and 10 have a similar trend to Figs. 7 and 8. Thus, either a large number of iterations or a proper interleaver can remove the performance variations due to the erasure patterns. Yang and Ryan [14] designed LDPCCs based on our results and showed that better designed LDPCCs can also efficiently avoid the performance variation between random and block erasure patterns.

VII. DESIGN OF AN LDPCC FOR THE MIXED CHANNEL

In this section, we discuss code optimization with respect to left and right degree distributions to minimize the following metric:

$$\min_{\arg \{\lambda(x), \rho(x)\}} M = \sum_{j=0}^L w_j \{E(\lambda(x), \rho(x), e_j) - K(e_j, r)\}$$

$$\text{given } r = 1 - \frac{\sum_{j=2}^{d_r} \rho_j / j}{\sum_{j=2}^{d_l} \lambda_j / j} \quad (24)$$

where $E(\cdot)$ is the E_b/N_0 threshold computed with GA given the degree distribution pair $(\lambda(x), \rho(x))$, $K(\cdot)$ is E_b/N_0 of the capacity for a coding rate of r , e_j 's are the erasure probabilities at which $E(\cdot)$ and $K(\cdot)$ are evaluated, and w_j 's are positive real weighting factors.

For an LDPCC on an AWGN channel, the weight factors become $w_0 = 1$ and $w_j = 0$ for all $j \neq 0$, and $e_0 = 0$. The optimization on the left degree distribution can be done through a linear optimization as described in Section V for a given right edge degree distribution that is jointly optimized through the differential evolution technique in [13]. It is also possible to design an LDPCC with the differential evolution technique on the left and right degree distributions with the recursive equation (20). We state the latter design procedure as follows.

- 1) [Initialization] Set $0 < \min E_b/N_0 < \max E_b/N_0$, $1 \ll N_{\max}$, $0 < \min P_e \ll 1$, $0 < \min \delta P_e \ll 1$, and $n = 0$.
- 2) $n = n + 1$.
- 3) [Differential Evolution] Get a trial vector composed of a pair of distributions $(\lambda_n(x), \rho_n(x))$ from the differential evolution whose coding rate is fixed.
- 4) [Compute Thresholds] Compute the threshold

$$E(\lambda_n(x), \rho_n(x), e_j)$$

at each e_j

- a) set $1 \ll K_{\max}$, $0 < \delta E \ll 1$, $E_{\max} = \max E_b/N_0$, and $E_{\min} = \min E_b/N_0$;

- b) $k = 0$, and

$$P_e^{(0)} = e_0/2 + (1 - e_0)Q\left(\sqrt{m_{u0}/2}\right);$$

- c) $E(\lambda_n(x), \rho_n(x), e_j) = (E_{\max} + E_{\min})/2$;

- d) if $(E_{\max} - E_{\min}) < \delta E$ go to 5)

- e) $k = k + 1$;

- f) compute $\varepsilon^{(k)}$ in (14) and store $\varepsilon^{(k)}$ for computing $\varepsilon^{(k+1)}$ at the next iteration;

- g) compute $m_u^{(k)}$ in (19) and store $m_u^{(k)}$ for computing $m_u^{(k+1)}$ at the next iteration;

- h) compute $P_e^{(k)}$ defined in (20) and

$$\delta P_e^{(k)} = (P_e^{(k-1)} - P_e^{(k)})/P_e^{(k-1)}$$

with $\varepsilon^{(k)}$ and $m_u^{(k)}$;

- i) If $P_e^{(k)} < \min P_e$ then

$$E_{\max} = E(\lambda_n(x), \rho_n(x), e_j)$$

and go to b);

j) If $K_{\max} < k$ or $\delta P_e^{(k)} < \min \delta P_e$ then

$$E_{\min} = E(\lambda_n(x), \rho_n(x), e_j)$$

and go to b);

k) Go to e).

- 5) [Compute Metric] Compute the metric in (24) with $E(\lambda_n(x), \rho_n(x), e_j)$'s for $0 \leq j \leq L$ and return the metric to the differential evolution routine that updates the best metric and the degree distribution pair.
- 6) If $n < N_{\max}$ then go to 2).
- 7) [Termination] Get the best degree distribution, $(\lambda^*(x), \rho^*(x))$ from the differential evolution routine and stop.

The design procedure is a combination of the differential evolution and the binary search algorithm. In the design procedure, we search for the best degree distribution pair $(\lambda^*(x), \rho^*(x))$ that minimizes the metric defined in (24). The equality in each step is an assignment operator instead of a Boolean operator. That is, the left variable will be the same as the right variable. In step 1) we define the initial values, where $\max E_b/N_0$ and $\min E_b/N_0$ are the upper and lower limits of the E_b/N_0 threshold at each erasure probability (e_j), $N_{\max}$ is the maximum number of iterations of this design procedure, and $\min P_e$ and $\min \delta P_e$ are the minimum bit-error probability and the minimum bit-error probability decrease, respectively, which are explained in step 4). In step 3), the differential evolution routine returns a randomly generated edge degree distribution pair $(\lambda_n(x), \rho_n(x))$. In step 4), the threshold at each erasure probability (e_j) is evaluated with the binary search algorithm. In the binary search, the threshold of an LDPCC is presumed to be between $E_{\max}$ and $E_{\min}$. In a), $E_{\max}$ and $E_{\min}$ are set to $\max E_b/N_0$ and $\min E_b/N_0$, respectively. $E(\lambda_n(x), \rho_n(x), e_j)$ will be always the middle of $E_{\max}$ and $E_{\min}$. In d), if the difference between $E_{\max}$ and $E_{\min}$ is less than $\delta E \ll 1$, which means the threshold is within $\delta E/2$ from the true threshold, the binary search terminates. From e) to h), the bit-error probability at the k th iteration, $P_e^{(k)}$, and the decrease of the bit-error probability $\delta P_e^{(k)}$ are evaluated. In i), if the bit-error probability is less than $\min P_e$, that is regarded as zero probability, the threshold is less than or equal to the current threshold. Thus, $E_{\max} = E(\lambda_n(x), \rho_n(x), e_j)$ and a search for the thresholds between $E_{\max}$ and $E_{\min}$ is performed again. In j), if $K_{\max} < k$ (too many iterations) or $\delta P_e < \min \delta P_e$, which means the iterative bit-error probability falls into a fixed point, the threshold must be larger than the current threshold $E(\lambda_n(x), \rho_n(x), e_j)$. In step 5), with the thresholds found through the binary search, the metric defined in (24) is passed to the differential evolution routine that makes a new random vector in step 3). The steps from 3) to 5) will be iterated $N_{\max}$ times, and we get the best results $(\lambda^*(x), \rho^*(x))$.

One can design a code that has been optimized for a specific erasure probability. We are more interested in a single code that is optimized across a range of erasure probabilities. Again, this is common in recording when the code is designed to handle a maximum number of erasures. We consider an LDPCC which has a coding rate of 1/2 and is optimized with uniform weighting factors. The metric in (24) is evaluated at every erasure probability between 0 and 0.35 with a 0.05 step. In Fig. 11, we compare between a well-designed LDPCC with GA for an AWGN channel in [10] and an LDPCC designed for the mixed channel, where the thresholds are computed with GA and DDE. The edge degree distribution pairs of the LDPCC in [10] and the designed one are listed in Table I. The designed LDPCC shows better thresholds if the erasure probability is greater than 0.1 in both the GA and DDE cases. The thresholds from DDE with erasure probability less than 0.1 are poorer than the LDPCC designed for the AWGN channel, which can be either mitigated or avoided with proper weighting factors.

Fig. 11. E_b/N_0 gap from the capacity; LDPCC designed for AWGN channel having a degree distribution of $\lambda(x) = 0.30780x + 0.27287x^2 + 0.41933x^6$ and $\rho(x) = 0.4x^5 + 0.6x^6$ and LDPCC designed over the mixed channel having degree distribution $\lambda(x) = 0.33175x + 0.24120x^2 + 0.42705x^6$ and $\rho(x) = 0.45290x^5 + 0.54710x^6$.

TABLE I
EDGE DEGREE DISTRIBUTION PAIRS OF AN LDPCC DESIGNED FOR AN AWGN CHANNEL ($\lambda(x), \rho(x)$) AND ONE DESIGNED FOR THE MIXED CHANNEL ($\lambda^*(x), \rho^*(x)$)

Degree	$\lambda(x)$	$\lambda^*(x)$
2	0.30780	0.33175
3	0.27287	0.24120
7	0.41933	0.42705
Degree	$\rho(x)$	$\rho^*(x)$
6	0.40000	0.45290
7	0.60000	0.54710

VIII. CONCLUSION

The Gaussian approximation is extended to include the mixture of white Gaussian noise with erasures, and the analytic results not only provide a way to predict asymptotic performance of LDPCCs but also help us design good LDPCCs over the mixed channel. The analysis tells us that the variation of the erasure probability during iterations does not depend on the received signal power but only on the structure of LDPCCs, which can be defined with edge degree distributions. Thus, as long as the erasure probability is less than a threshold of an LDPCC over BECs, we can assume that the erasure probability monotonically decreases to zero, which gives us a simple recursive equation describing message densities, called a steady-state equation. This simplified equation leads us to a design rule of good LDPCCs over the mixed channel and a graphical interpretation of the convergence of LDPCCs over the mixed channel.

To show the validity of the analytic results and the channel model, we simulated an LDPCC having a code block length of 131072 and a coding rate of 1/2. We also computed the E_b/N_0 thresholds of the LDPCC using the analytic results. The comparison between the simulation results and the thresholds showed that the prediction from the theoretical results was accurate enough.

We have found that well-designed LDPCCs for AWGN channels are still good over the mixed channel. In the simulation, we had less than 0.1-dB E_b/N_0 loss up to an erasure probability of 0.2. We also investigated the performance variations due to types of erasure patterns with different block lengths and different values of maximum iterations. The simulation results allowed us to conclude that to mitigate the performance variations, we need a large number of iterations or a proper interleaver before the decoder.

Although good LDPCs are also good over the mixed channel, we can, in the average sense, improve LDPCs over several erasure probabilities. We optimized an LDPC for the mixed channel with a design rule from the analytic results to achieve a better average metric. The designed code shows better thresholds at erasure probabilities larger than 0.1 by 0.1 dB in GA and DDE. From a practical point of view, we believe our work would be helpful to predict thresholds of LDPCs over the mixed channel, especially for magnetic and optical storage applications. From a theoretical point of view, it also paves the way for a new study of LDPCs over the mixed channel.

ACKNOWLEDGMENT

The authors would like to acknowledge M. Yang and W. Ryan of the University of Arizona for providing a preprint of [14].

REFERENCES

- [1] R. G. Gallager, "Low-density parity check codes," *IRE Trans. Inform. Theory*, vol. IT-8, pp. 21–22, Jan. 1962.
- [2] —, *Low-Density Parity Check Codes*. Cambridge, MA: MIT Press, 1963.
- [3] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding," in *Proc. IEEE Int. Conf. Communications*, Geneva, Switzerland, May 1993, pp. 1064–1070.
- [4] D. J. C. MacKay and R. M. Neal, "Near Shannon limit performance of low-density parity-check codes," *Electron. Lett.*, vol. 32, pp. 1645–1646, Aug. 1996.
- [5] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Trans. Inform. Theory*, vol. 45, pp. 399–431, Mar. 1999.
- [6] M. Sipser and D. Spielman, "Expander codes," *IEEE Trans. Inform. Theory*, vol. 42, pp. 1710–1722, Nov. 1996.
- [7] M. Luby, M. Mitzenmacher, A. Shokrollahi, D. Spielman, and V. Stemann, "Practical loss-resilient codes," in *Proc. 29th Annu. ACM Symp. Theory of Computing*, 1997, pp. 150–159.
- [8] T. Richardson and R. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Trans. Inform. Theory*, vol. 47, pp. 599–618, Feb. 2001.
- [9] S.-Y. Chung, "On the construction of some capacity-approaching coding schemes," Ph.D. dissertation, MIT, Cambridge, MA, 2000.
- [10] S.-Y. Chung, T. Richardson, and R. Urbanke, "Analysis of sum-product decoding of low-density parity-check codes using a Gaussian approximation," *IEEE Trans. Inform. Theory*, vol. 47, pp. 657–670, Feb. 2001.
- [11] T. Richardson, A. Shokrollahi, and R. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 47, pp. 619–637, Feb. 2001.
- [12] M. Luby, M. Mitzenmacher, A. Shokrollahi, and D. Spielman, "Analysis of low density codes and improved designs using irregular graphs," in *Proc. 30th Annu. ACM Symp. Theory of Computing*, 1998, pp. 249–258.
- [13] K. Price and R. Storn, "Differential evolution-A simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optimiz.*, vol. 11, pp. 341–359, 1997.
- [14] M. Yang and W. Ryan, "Design of LDPC codes for the burst-erasure channel with AWGN," paper, to be published.

New Covering Codes From an ADS-Like Construction

Markku K. Kaikkonen and Petri Rosendahl

Abstract—A covering code construction is presented. Using this construction it is shown that $t[52, 39] = 3$, $t[36, 21] = 4$, $t[58, 32] = 7$, $K(32, 2) \leq 62 \cdot 2^{18}$, and $K(62, 5) \leq 31 \cdot 2^{37}$, where $t[n, k]$ is the minimum covering radius among all binary $[n, k]$ codes and $K(n, R)$ is the minimum cardinality of a binary code of length n and covering radius R . Four new linear codes found by computer search are also given. These include a $[23, 9]_5$ code, a $[32, 8]_{10}$ code, a $[51, 41]_2$ code, and a $[45, 20]_8$ code.

Index Terms—Amalgamated direct sum (ADS), covering code, normality.

I. INTRODUCTION

Let $\mathbb{F} = \{0, 1\}$ be the binary field. An (n, K) code is a K -element subset of the space $\mathbb{F}^n$, and an $[n, k]$ linear code is a k -dimensional subspace of $\mathbb{F}^n$. The covering radius $R(C)$ of a code C is by definition the maximum (Hamming) distance from vectors in $\mathbb{F}^n$ to the code C , i.e.,

$$R(C) = \max_{x \in \mathbb{F}^n} \min_{c \in C} d(x, c).$$

An $(n, K)R$ (or an $[n, k]R$) code is a code with covering radius at most R . The covering radius of a linear code can be calculated by means of its parity-check matrix H . Namely, the covering radius of an $[n, k]$ code is the smallest integer R such that every binary column vector of length $n - k$ is a sum of at most R columns of H (for a proof, see [2, Ch. 2, p. 21]).

Let A be a $k \times n$ matrix over $\mathbb{F}$. We shall always assume that A has full rank. A set $S \subseteq \mathbb{F}^k$ is said to R -cover $\mathbb{F}^k$ using A if for every $x \in \mathbb{F}^k$ there is a vector $y \in \mathbb{F}^n$ of weight at most R and a vector $s \in S$ such that $x = Ay + s$. The following theorem is from [1] and [5] (see also [2, Ch. 3, p. 70]); it is known as the *matrix construction*.

Theorem 1: If the set S R -covers $\mathbb{F}^k$ using A , then the code

$$C = \{c \in \mathbb{F}^n : Ac \in S\}$$

has $|S|2^{n-k}$ codewords and covering radius at most R .

The code in Theorem 1 is a union of some cosets of a linear code. In particular, if $S = \{0\}$, or more generally, if S is linear, then the code is linear.

Let C be a binary (n, K) code with covering radius R . Let $C_0^{(i)}$ (resp., $C_1^{(i)}$) denote the set of codewords in which the i th coordinate is 0 (resp., 1). The code C is said to be *normal* if there is a coordinate i such that for all $x \in \mathbb{F}^n$

$$d(x, C_0^{(i)}) + d(x, C_1^{(i)}) \leq 2R + 1.$$

In this case, the coordinate i is said to be acceptable.

The columns of a $k \times n$ matrix H are denoted by $h_1, h_2, \dots, h_n$.

The following lemma is an immediate generalization of a theorem in [4].

Manuscript received May 2, 2001; revised February 10, 2003. The work of P. Rosendahl was supported by the Academy of Finland under Grant 44002.

M. K. Kaikkonen is at Halkosuntie 65 C 5, 00660 Helsinki, Finland.

P. Rosendahl is with the Department of Mathematics and the Turku Centre for Computer Science TUCS, University of Turku, 20014 Turku, Finland.

Communicated by S. Litsyn, Associate Editor for Coding Theory.

Digital Object Identifier 10.1109/TIT.2003.813508